IBM Systems - iSeries

Database
SQL messages and codes

Version 5 Release 4

IBM Systems - iSeries

Database
SQL messages and codes

Version 5 Release 4

Note

Before using this information and the product it supports, read the information in "Notices," on page 189.

Fifth Edition (February 2006)

This edition applies to version 5, release 4, modification 0 of IBM i5/OS (product number 5722–SS1) and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CISC models.

Contents

SQL messages and codes

Use this topic to find descriptions of codes and messages returned when using SQL with DB2 Universal Database™ for iSeries™ (DB2® UDB for iSeries). This topic contains listings of SQLCODEs, SQLSTATEs, class codes, and SQL messages.

Note: By using the code examples, you agree to the terms of the "Code license and disclaimer information" on page 187.

What's new for V5R4

This topic highlights the changes made to this topic collection for V5R4.

New messages and codes added

Several additions were made to the "Listing of SQLSTATE class codes" on page 3, "Listing of SQLSTATE values" on page 4, and the Listing of SQL messages.

How to see what's new or changed

To help you see where technical changes have been made, this information uses:

- The » image to mark where new or changed information begins.
- The « image to mark where new or changed information ends.

To find other information about what's new or changed this release, see the Memo to users.

Printable PDF

Use this to view and print a PDF of this information.

To view or download the PDF version of this document, select SQL Messages and Codes (about 2824KB).

Saving PDF files

To save a PDF on your workstation for viewing or printing:

1. Right-click the PDF in your browser (right-click the link above).
2. Click the option that saves the PDF locally.
3. Navigate to the directory in which you want to save the PDF.
4. Click **Save**.

Downloading Adobe Reader

You need Adobe Reader installed on your system to view or print these PDFs. You can download a free copy from the Adobe Web site (www.adobe.com/products/acrobat/readstep.html).

SQLCODE and SQLSTATE concepts

This reference provides a list of SQLCODEs and their associated SQLSTATEs. In this reference, you can find instructions for finding a SQLCODE in the message file along with the text for these messages.

SQLCODEs and SQLSTATEs are returned in the SQLCA structure or through the GET DIAGNOSTICS statement. SQLSTATE is the preferred standard return code. It provides application programs with return codes for common error conditions found among the DB2 Universal Database products. SQLSTATEs are particularly useful when handling errors in distributed SQL applications.

SQLCODE

An SQLCODE is a return code. The return code is sent by the database manager after completion of each SQL statement.

Each SQLCODE that is recognized by a DB2 UDB for iSeries server has a corresponding message in the message file QSQLMSG. The message identifier for any SQLCODE is constructed by appending the absolute value (5 digits) of the SQLCODE to SQ and changing the third character to L if the first character of the SQLCODE is 0. For example, if the SQLCODE is 30070, the message identifier is SQ30070. If the SQLCODE is -0204, the message identifier is SQL0204. Lastly, if the SQLCODE is a 3-digit positive number, a zero is added before the first digit. For example, if the SQLCODE is 551, the message identifier is SQL0551.

SQLSTATE

SQLSTATE provides application programs with common return codes for success, warning, and error conditions found among the DB2 Universal Database products. SQLSTATE values are particularly useful when handling errors in distributed SQL applications. SQLSTATE values are consistent with the SQLSTATE specifications contained in the SQL 1999 standard.

An SQLSTATE value is a return code that indicates the outcome of the most recently executed SQL statement. The mechanism used to access SQLSTATE values depends on where the SQL statement is executed. In Java™, SQLSTATE values are returned by using getSQLState() method. In SQL functions, SQL procedures, SQL triggers, and embedded applications other than Java, SQLSTATE values are returned in the following:

- The last five bytes of the SQLCA
- A stand-alone SQLSTATE variable
- The GET DIAGNOSTICS statement

SQLSTATE values are designed so that application programs can test for specific conditions or classes of conditions.

SQLSTATE values are comprised of a two-character class code value, followed by a three-character subclass code value. Class code values represent classes of successful and unsuccessful completion conditions. If you want to use SQLSTATE as the basis of your application's return codes, you can define your own SQLSTATE classes or subclasses using the following guidelines:

- SQLSTATE classes that begin with the characters 7 through 9 or I through Z can be defined. Within these classes, any subclass can be defined.
- SQLSTATE classes that begin with the characters 0 through 6 or A through H are reserved for the database manager. Within these classes, subclasses that begin with the characters 0 through H are reserved for the database manager. Subclasses that begin with the characters I through Z can be defined.

The class code of an SQLSTATE value indicates whether the SQL statement was executed successfully (class codes 00 and 01) or unsuccessfully (all other class codes).

SQLSTATE is related to SQLCODE. Every SQLSTATE has one or more SQLCODEs associated with it. An SQLSTATE can refer to more than one SQLCODE.

SQLSTATEs returned by non-DB2 UDB for iSeries application servers

When an SQLSTATE other than 00000 is returned from an application server that is not DB2 UDB for iSeries, DB2 UDB for iSeries attempts to map the SQLSTATE to a DB2 UDB for iSeries SQLCODE and message with the following results:

- If the SQLSTATE is not recognized by DB2 UDB for iSeries, the common message for the class is issued.
- If the SQLSTATE and SQLCODE correspond to a single DB2 UDB for iSeries SQLCODE, DB2 UDB attempts to convert the returned tokens into the replacement data that the SQL message expects. If an error occurs while converting the tokens:
  - The tokens are returned without conversion in the SQLERRM field of the SQLCA or in the SQL diagnostics area.
  - A common message for the class code of the SQLSTATE is issued.

Related reference

"Listing of SQLSTATE class codes"
The table below provides a list of general SQLSTATE classes. Each class links to a list of the more specific SQLSTATE codes that comprise that class.

"Listing of SQLSTATE values" on page 4
The tables in this topic provide descriptions of SQLSTATE codes that can be returned to applications by DB2 UDB for iSeries. The tables include SQLSTATE values, their meanings, and their corresponding SQLCODE values.

Related information

SQL message finder

Listing of SQLSTATE class codes

The table below provides a list of general SQLSTATE classes. Each class links to a list of the more specific SQLSTATE codes that comprise that class.

Class code	Meaning	Subcodes
00	Unqualified Successful Completion	Table 1 on page 4
01	Warning	Table 2 on page 4
02	No Data	Table 3 on page 6
07	Dynamic SQL Error	Table 4 on page 7
08	Connection Exception	Table 5 on page 7
09	Triggered Action Exception	Table 6 on page 7
0A	Feature Not Supported	Table 7 on page 7
0E	Invalid Schema Name List Specification	Table 8 on page 7
0F	Invalid Token	Table 9 on page 8
0K	Resignal When Handler Not Active	Table 10 on page 8
0W	Prohibited Statement Encountered During Trigger	Table 11 on page 8
0Z	Diagnostics Exception	Table 12 on page 8
20	Case Not Found for Case Statement	Table 13 on page 8
21	Cardinality Violation	Table 14 on page 8
22	Data Exception	Table 15 on page 8
23	Constraint Violation	Table 16 on page 9
24	Invalid Cursor State	Table 17 on page 10

Class code	Meaning	Subcodes
25	Invalid Transaction State	Table 18 on page 10
26	Invalid SQL Statement Identifier	Table 19 on page 10
27	Triggered Data Change Violation	Table 20 on page 10
28	Invalid Authorization Specification	Table 21 on page 11
2D	Invalid Transaction Termination	Table 22 on page 11
2E	Invalid Connection Name	Table 23 on page 11
2F	SQL Function Exception	Table 24 on page 11
33	Invalid SQL Descriptor Name	Table 25 on page 11
34	Invalid Cursor Name	Table 26 on page 11
35	Invalid Condition Number	Table 27 on page 11
36	Cursor Sensitivity Exception	Table 28 on page 12
38	External Function Exception	Table 29 on page 12
39	External Function Call Exception	Table 30 on page 12
3B	Invalid SQL Descriptor Name	Table 31 on page 12
3C	Ambiguous Cursor Name	Table 32 on page 12
3F	Invalid Schema (Collection) Name	Table 33 on page 12
42	Syntax Error or Access Rule Violation	Table 34 on page 13
44	WITH CHECK OPTION Violation	Table 35 on page 20
46	JavaTM Errors	Table 36 on page 20
51	Invalid Application State	Table 37 on page 20
53	Invalid Operand or Inconsistent Specification	Table 38 on page 21
54	SQL or Product Limit Exceeded	Table 39 on page 21
55	Object Not in Prerequisite State	Table 40 on page 22
56	Miscellaneous SQL or Product Error	Table 41 on page 22
57	Resource Not Available or Operator Intervention	Table 42 on page 22
58	System Error	Table 43 on page 23

Listing of SQLSTATE values

The tables in this topic provide descriptions of SQLSTATE codes that can be returned to applications by DB2 UDB for iSeries. The tables include SQLSTATE values, their meanings, and their corresponding SQLCODE values.

Table 1. **Class code 00: Unqualified Successful Completion**

SQLSTATE Value	Meaning	SQLCODE Values
00000	Completion of the operation was successful and did not result in any type of warning or exception condition.	+000

Table 2. **Class code 01: Warning**

SQLSTATE Value	Meaning	SQLCODE Values
01002	A DISCONNECT error occurred.	+596

Table 2. **Class code 01: Warning** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
01003	Null values were eliminated from the argument of a column function.	+000
01004	The value of a string was truncated when assigned to another string data type with a shorter length.	+000, +445, +802, +20141
01005	Insufficient number of entries in an SQLDA.	+239
01006	A privilege was not revoked.	+569
01007	A privilege was not granted.	+570
01009	The search condition is too long for the information schema.	+177
0100A	The query expression of the view is too long for the information schema.	+178
0100C	One or more ad hoc result sets were returned from the procedure.	+466
0100D	The cursor that was closed has been reopened on the next result set within the chain.	+467
0100E	The procedure returned too many result sets.	+464, +20206
01503	The number of result columns is larger than the number of host variables provided.	+000, +030
01504	The UPDATE or DELETE statement does not include a WHERE clause.	+000, +088
01505	The statement was not executed because it is unacceptable in this environment.	+084
01506	An adjustment was made to a DATE or TIMESTAMP value to correct an invalid date resulting from an arithmetic operation.	+000
01515	The null value has been assigned to a host variable, because the non-null value of the column is not within the range of the host variable.	+304
01517	A character that cannot be converted was replaced with a substitute character.	+335
01519	The null value has been assigned to a host variable, because a numeric value is out of range.	+802
01520	The null value has been assigned to a host variable, because the characters cannot be converted.	+331
01522	The local table or view name used in the CREATE ALIAS statement is undefined.	+403
01526	Isolation level has been escalated.	+595
01527	A SET statement references a special register that does not exist at the AS.	+799
01528	WHERE NOT NULL is ignored, because the index key cannot contain null values.	+645
01532	An undefined object name was detected.	+204
01534	The string representation of a datetime value is invalid.	+180, +181
01535	An arithmetic operation on a date or timestamp has a result that is not within the valid range of dates.	+183
01536	During remote bind where existence checking is deferred, the server-name specified does not match the current server.	+114
01539	Connection is successful but only SBCS characters should be used.	+863
01542	Authorization ID does not have the privilege to perform the operation as specified.	+552
01544	The null value has been assigned to a host variable, because a substring error occurred; for example, an argument of SUBSTR is out of range.	+138

Table 2. **Class code 01: Warning** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
01545	An unqualified column name has been interpreted as a correlated reference.	+012
01547	A mixed data value is improperly formed.	+191, +304, +802
01548	The authorization ID does not have the privilege to perform the specified operation on the identified object.	+551
01557	Too many host variables have been specified on SELECT INTO or FETCH.	+326
01564	The null value has been assigned to a host variable, because division by zero occurred.	+802
01565	The null value has been assigned to a host variable, because a miscellaneous data exception occurred; for example, the character value for the CAST, DECIMAL, FLOAT, or INTEGER scalar function is invalid; a floating-point NAN (not a number) or invalid data in a packed decimal field was detected.	+304, +420, +802
01567	The table was created but not journaled.	+7905
01587	The unit of work was committed or rolled back, but the outcome is not fully known at all sites.	+990
01593	An ALTER TABLE might cause data truncation.	+460
01594	Insufficient number of entries in an SQLDA for ALL information (that is, not enough descriptors to return the distinct name).	+237
01623	The value of DEGREE is ignored.	+1530
01627	The DATALINK value might not be valid because the table is in reconcile pending or reconcile is not a possible state.	+360
01634	The distinct data type name is too long and cannot be returned in the SQLDA. The short name is returned instead.	+7036
01643	Assignment to SQLCODE or SQLSTATE variable does not signal a warning or error.	+385
01646	A result sets cannot be returned because the cursor was closed.	+7050
01647	A DB2SQL BEFORE trigger changed to DB2ROW.	+7051
01658	Binary data is invalid for DECRYPT_CHAR and DECYRYPT_DB.	+20224
01660	The routine was created but a restore will not update the catalog.	+7909
01662	Release record option ignored on CLOSE.	+30107
01Hxx	Valid warning SQLSTATEs returned by a user-defined function or external procedure CALL.	+462

Table 3. **Class code 02: No Data**

SQLSTATE Value	Meaning	SQLCODE Values
02000	One of the following exceptions occurred: • The result of the SELECT INTO statement or the subselect of the INSERT statement was an empty table. • The number of rows identified in the searched UPDATE or DELETE statement was zero. • The position of the cursor referenced in the FETCH statement was after the last row of the result table. • The fetch orientation is invalid.	+100
02001	No additional result sets returned.	+387

Table 3. **Class code 02: No Data** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
02505	The GET DESCRIPTOR VALUE is greater than COUNT.	+20298

Table 4. **Class code 07: Dynamic SQL Error**

SQLSTATE Value	Meaning	SQLCODE Values
07001	The number of host variables is not correct for the number of parameter markers.	-313
07002	The call parameter list or control block is invalid.	-804
07003	The statement identified in the EXECUTE statement is a select-statement, or is not in a prepared state.	-518
07004	The USING clause or INTO clause is required for dynamic parameters.	-313
07005	The statement name of the cursor identifies a prepared statement that cannot be associated with a cursor.	-517
07006	An input host variable, transition variable, or parameter marker cannot be used, because of its data type.	-301
07008	The descriptor count is invalid.	-074
07009	The descriptor index is invalid.	-075

Table 5. **Class code 08: Connection Exception**

SQLSTATE Value	Meaning	SQLCODE Values
08001	The application requester is unable to establish the connection.	-30080, -30082, -30089
08002	The connection already exists.	-842
08003	The connection does not exist.	-843, -900
08004	The application server rejected establishment of the connection.	-30060, -30061
08501	A DISCONNECT is not allowed when the connection uses an LU6.2 protected conversation.	-858

Table 6. **Class code 09: Triggered Action Exception**

SQLSTATE Value	Meaning	SQLCODE Values
09000	A triggered SQL statement failed.	-723

Table 7. **Class code 0A: Feature Not Supported**

SQLSTATE Value	Meaning	SQLCODE Values
0A001	The CONNECT statement is invalid, because the process is not in the connectable state.	-752

Table 8. **Class code 0E: Invalid Schema Name List Specification**

SQLSTATE Value	Meaning	SQLCODE Values
0E000	The schema name list in a SET PATH statement is not valid.	-329

Table 9. **Class code 0F: Invalid Token**

SQLSTATE Value	Meaning	SQLCODE Values
0F001	The locator value does not currently represent any value.	-423

Table 10. **Class code 0K: Resignal When Handler Not Active**

SQLSTATE Value	Meaning	SQLCODE Values
0K000	A RESIGNAL was issued but a handler is not active.	-787

Table 11. **Class code 0W: Prohibited Statement Encountered During Trigger**

SQLSTATE Value	Meaning	SQLCODE Values
0W000	The statement is not allowed in a trigger.	-751

Table 12. **Class code 0Z: Diagnostics Exception**

SQLSTATE Value	Meaning	SQLCODE Values
0Z001	Maximum number of stacked diagnostics areas exceeded.	-20226
0Z002	Stacked diagnostics accessed without an active handler.	-20228

Table 13. **Class code 20: Case Not Found for Case Statement**

SQLSTATE Value	Meaning	SQLCODE Values
20000	The case was not found for the CASE statement.	-773

Table 14. **Class code 21: Cardinality Violation**

SQLSTATE Value	Meaning	SQLCODE Values
21000	The result of a SELECT INTO, scalar fullselect, or subquery of a basic predicate is more than one value.	-811

Table 15. **Class code 22: Data Exception**

SQLSTATE Value	Meaning	SQLCODE Values
22001	Character data, right truncation occurred; for example, an update or insert value is a string that is too long for the column, or a datetime value cannot be assigned to a host variable, because it is too small.	-302, -303, -404, -433, -802
22002	A null value, or the absence of an indicator parameter was detected; for example, the null value cannot be assigned to a host variable, because no indicator variable is specified.	-305
22003	A numeric value is out of range.	-302, -304, -406, -446, -802
22004	A null value is not allowed.	-087, -305
22005	An error occurred on assignment.	-076
22006	The fetch orientation is invalid.	-231

Table 15. **Class code 22: Data Exception** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
22007	An invalid datetime format was detected; that is, an invalid string representation or value was specified.	-180, -181
22008	Datetime field overflow occurred; for example, an arithmetic operation on a date or timestamp has a result that is not within the valid range of dates.	-183
22011	A substring error occurred; for example, an argument of SUBSTR is out of range.	-138
22012	Division by zero is invalid.	-802
22018	The character value for the CAST, DECIMAL, FLOAT, or INTEGER scalar function is invalid.	-420
22019	The LIKE predicate has an invalid escape character.	-130
22021	A character is not in the coded character set or the conversion is not supported.	-330, -331
22023	A parameter or host variable value is invalid.	-302, -304, -406, -802
22024	A NUL-terminated input host variable or parameter did not contain a NUL.	-302
22025	The LIKE predicate string pattern contains an invalid occurrence of an escape character.	-130
22501	The length control field of a variable length string is negative or greater than the maximum.	-311
22503	The string representation of a name is invalid.	-188
22504	A mixed data value is invalid.	-191, -304, -406, -802
22511	ADT length exceeds maximum column length. The value for a ROWID or reference column is not valid.	-399
22522	A CCSID value is not valid at all, not valid for the data type or subtype, or not valid for the encoding scheme.	-189
22524	Character conversion resulted in truncation	-334
22525	Partitioning key value is not valid.	-327
22527	Invalid input data detected for a multiple row insert.	-30106

Table 16. **Class code 23: Constraint Violation**

SQLSTATE Value	Meaning	SQLCODE Values
23001	The update or delete of a parent key is prevented by a RESTRICT update or delete rule.	-531, -532
23502	An insert or update value is null, but the column cannot contain null values.	-407
23503	The insert or update value of a foreign key is invalid.	-530
23504	The update or delete of a parent key is prevented by a NO ACTION update or delete rule.	-531, -532
23505	A violation of the constraint imposed by a unique index or a unique constraint occurred.	-803
23511	A parent row cannot be deleted, because the check constraint restricts the deletion.	-543
23512	The check constraint cannot be added, because the table contains rows that do not satisfy the constraint definition.	-544

Table 16. **Class code 23: Constraint Violation** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
23513	The resulting row of the INSERT or UPDATE does not conform to the check constraint definition.	-545
23515	The unique index cannot be created or unique constraint added because the table contains duplicate values of the specified key.	-603
23520	The foreign key cannot be defined, because all of its values are not equal to a parent key of the parent table.	-667
23522	The range of values for the identity column or sequence is exhausted.	-359

Table 17. **Class code 24: Invalid Cursor State**

SQLSTATE Value	Meaning	SQLCODE Values
24501	The identified cursor is not open.	-501, -507
24502	The cursor identified in an OPEN statement is already open.	-502, -7055
24504	The cursor identified in the UPDATE, DELETE, SET, or GET statement is not positioned on a row.	-508
24506	The statement identified in the PREPARE is the statement of an open cursor.	-519
24507	FETCH CURRENT was specified, but the current row is deleted, or a value of an ORDER BY column of the current row has changed.	-226
24513	FETCH NEXT, PRIOR, CURRENT, or RELATIVE is not allowed, because the cursor position is not known.	-227
24514	A previous error has disabled this cursor.	-906

Table 18. **Class code 25: Invalid Transaction State**

SQLSTATE Value	Meaning	SQLCODE Values
25000	An update operation is invalid for the application execution environment.	-30090
25006	An update operation is not valid because the transaction is read-only.	-817
25501	The statement is only allowed as the first statement in a unit of work.	-428

Table 19. **Class code 26: Invalid SQL Statement Identifier**

SQLSTATE Value	Meaning	SQLCODE Values
26501	The statement identified does not exist.	-514, -516
26510	The statement name specified in a DECLARE CURSOR already has a cursor allocated to it.	-5023

Table 20. **Class code 27: Triggered Data Change Violation**

SQLSTATE Value	Meaning	SQLCODE Values
27000	An attempt was made to modify the target table of the MERGE statement by a constraint or trigger.	-907

Table 21. **Class code 28: Invalid Authorization Specification**

SQLSTATE Value	Meaning	SQLCODE Values
28000	Authorization name is invalid.	-113, -188, -567

Table 22. **Class code 2D: Invalid Transaction Termination**

SQLSTATE Value	Meaning	SQLCODE Values
2D522	COMMIT and ROLLBACK are not allowed in an ATOMIC Compound statement.	-774
2D528	Dynamic COMMIT or COMMIT ON RETURN procedure is invalid for the application execution environment	-426, -30090
2D529	Dynamic ROLLBACK is invalid for the application execution environment.	-427, -30090

Table 23. **Class code 2E: Invalid Connection Name**

SQLSTATE Value	Meaning	SQLCODE Values
2E000	Connection name is invalid.	-113, -188, -251

Table 24. **Class code 2F: SQL Function Exception**

SQLSTATE Value	Meaning	SQLCODE Values
2F002	The SQL function attempted to modify data, but the function was not defined as MODIFIES SQL DATA.	-577
2F003	The statement is not allowed in a function or procedure.	-751
2F004	The SQL function attempted to read data, but the function was not defined as READS SQL DATA.	-579
2F005	The function did not execute a RETURN statement.	-578

Table 25. **Class code 33: Invalid SQL Descriptor Name**

SQLSTATE Value	Meaning	SQLCODE Values
33000	SQL descriptor name is invalid.	-077

Table 26. **Class code 34: Invalid Cursor Name**

SQLSTATE Value	Meaning	SQLCODE Values
34000	Cursor name is invalid.	-504

Table 27. **Class code 35: Invalid Condition Number**

SQLSTATE Value	Meaning	SQLCODE Values
35000	Condition number is invalid.	-393

Table 28. **Class code 36: Cursor Sensitivity Exception**

SQLSTATE Value	Meaning	SQLCODE Values
36001	A SENSITIVE cursor cannot be defined for the specified select-statement.	-243

Table 29. **Class code 38: External Function Exception**

SQLSTATE Value	Meaning	SQLCODE Values
38xxx	Valid error SQLSTATEs returned by an external routine or trigger.	-443
38001	The external routine is not allowed to execute SQL statements.	-487
38002	The external routine attempted to modify data, but the routine was not defined as MODIFIES SQL DATA.	-577
38003	The statement is not allowed in a routine.	-751
38004	The external routine attempted to read data, but the routine was not defined as READS SQL DATA.	-579
38501	Error occurred while calling a user-defined function, external procedure, or trigger (using the SIMPLE CALL or SIMPLE CALL WITH NULLS calling convention).	-443, -4302

Table 30. **Class code 39: External Function Call Exception**

SQLSTATE Value	Meaning	SQLCODE Values
39004	A null value is not allowed for an IN or INOUT argument when using PARAMETER STYLE GENERAL or an argument that is a Java primitive type.	-470, -20205

Table 31. **Class code 3B: Savepoint Exception**

SQLSTATE Value	Meaning	SQLCODE Values
3B001	The savepoint is not valid.	-880
3B002	The maximum number of savepoints has been reached.	-20112
3B501	A duplicate savepoint name was detected.	-881
3B502	A RELEASE or ROLLBACK TO SAVEPOINT was specified, but a savepoint does not exist.	-882

Table 32. **Class code 3C: Ambiguous Cursor Name**

SQLSTATE Value	Meaning	SQLCODE Values
3C000	The cursor name is ambiguous.	-051

Table 33. **Class code 3F: Invalid Schema (Collection) Name**

SQLSTATE Value	Meaning	SQLCODE Values
3F000	The schema (collection) name is invalid.	-713

Table 34. **Class code 42: Syntax Error or Access Rule Violation**

SQLSTATE Value	Meaning	SQLCODE Values
42501	The authorization ID does not have the privilege to perform the specified operation on the identified object.	-551
42502	The authorization ID does not have the privilege to perform the operation as specified.	-552
42506	Owner authorization failure occurred.	-30053
42601	A character, token, or clause is invalid or missing.	-007, -011, -029, -097, -104, -109, -115, -128, -199, -441, -491
42602	A character that is invalid in a name has been detected.	-113, -251
42603	An unterminated string constant has been detected.	-010
42604	An invalid numeric or string constant has been detected.	-103, -105
42605	The number of arguments specified for a scalar function is invalid.	-170
42606	An invalid hexadecimal constant has been detected.	-110
42607	An operand of a column function or CONCAT operator is invalid.	-112
42609	All operands of an operator or predicate are parameter markers.	-417
42610	A parameter marker is not allowed.	-184, -418
42611	The column or argument definition is invalid.	-106, -604
42612	The statement string is an SQL statement that is not acceptable in the context in which it is presented.	-084, -142
42613	Clauses are mutually exclusive.	-628
42614	A duplicate keyword or clause is invalid.	-637
42615	An invalid alternative was detected.	-644
42616	Invalid options are specified.	-5047
42617	The statement string is blank or empty.	-198
42618	A host variable is not allowed.	-090, -312, -5012, -5024
42620	Read-only SCROLL was specified with the UPDATE clause.	-228
42621	The check constraint generated column expression is invalid.	-546
42622	A name or label is too long.	-107
42623	A DEFAULT clause cannot be specified.	-373
42625	A CASE expression is invalid.	-580
42629	Parameter names must be specified for SQL routines.	-078
42631	An expression must be specified on a RETURN statement in an SQL function.	-057
42633	An AS clause is required for an argument of XMLATTRIBUTES or XMLFOREST.	-20227
42634	The XML name is not valid.	-20275
42635	The XML namespace prefix is not valid.	-20276
42701	A duplicate column name in an INSERT or UPDATE operation or the SET transition-variable was detected.	-121
42702	A column reference is ambiguous, because of duplicate names.	-203

Table 34. **Class code 42: Syntax Error or Access Rule Violation** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
42703	An undefined column or parameter name was detected.	-205, -206, -213, -5001
42704	An undefined object or constraint name was detected.	-204
42705	An undefined server-name was detected.	-950
42707	A column name in ORDER BY does not identify a column of the result table.	-208
42709	A duplicate column name was specified in a key column list.	-537
42710	A duplicate object or constraint name was detected.	-456, -601
42711	A duplicate column name was detected in the object definition or ALTER TABLE statement.	-612
42712	A duplicate table designator was detected in the FROM clause or REFERENCING clause of a CREATE TRIGGER statement.	-212
42713	A duplicate object was detected in a list or is the same as an existing object.	-242
42718	The local server name is not defined.	-250
42723	A function with the same signature already exists in the schema.	-454
42724	Unable to access an external program used for a user-defined function or a procedure.	-444, -4300, -4303, -4304, -4306
42725	A routine or method was referenced directly (not by either signature or by specific instance name), but there is more than one specific instance of that routine or method.	-476
42726	Duplicate names for common table expressions were detected.	-340
42732	A duplicate schema name in the SET CURRENT PATH statement was detected.	-585
42733	A procedure with the specified name cannot be added to the schema because the procedure overloading is not allowed in this database and there is already a procedure with the same name in the schema.	-484
42734	A duplicate parameter-name, SQL variable name, label, or condition-name was detected.	-590
42736	The label specified on the GOTO, ITERATE, or LEAVE statement is not found or not valid.	-779
42737	The condition specified is not defined.	-781
42738	A duplicate column name or unnamed column was specified in a DECLARE CURSOR statement of a FOR statement.	-783
42747	The same descriptor item was specified more than once in the same SET DESCRIPTOR statement.	-20299
42802	The number of insert or update values is not the same as the number of columns.	-117
42803	A column reference in the SELECT or HAVING clause is invalid, because it is not a grouping column; or a column reference in the GROUP BY clause is invalid.	-119, -122
42804	The result expressions in a CASE expression are not compatible.	-581
42805	An integer in the ORDER BY clause does not identify a column of the result table.	-125
42806	A value cannot be assigned to a host variable, because the data types are not compatible.	-303
42807	The INSERT, UPDATE, or DELETE is not permitted on this object.	-150, -155

Table 34. **Class code 42: Syntax Error or Access Rule Violation** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
42808	A column identified in the INSERT or UPDATE operation cannot be updated.	-151
42809	The identified object is not the type of object to which the statement applies.	-152, -156, -159
42810	A base table is not identified in a FOREIGN KEY clause.	-157
42811	The number of columns specified is not the same as the number of columns in the SELECT clause.	-158
42813	WITH CHECK OPTION cannot be used for the specified view.	-160
42814	The column cannot be dropped, because it is the only column in the table.	-195
42815	The data type, length, scale, value, or CCSID is invalid.	-060, -171, -451, -713, -846, -5005
42816	A datetime value or duration in an expression is invalid.	-182, -187
42817	The column cannot be dropped, because RESTRICT was specified and a view or constraint is dependent on the column or the column is part of a partitioning key.	-196
42818	The operands of an operator or function are not compatible.	-131, -401
42819	An operand of an arithmetic operation or an operand of a function that requires a number is not a number.	-402
42820	A numeric constant is too long, or it has a value that is not within the range of its data type.	-405, -410
42821	A data type for an assignment to a column or variable is not compatible with the data type.	-408
42822	An expression in the ORDER BY clause or GROUP BY clause is not valid.	-214
42823	Multiple columns are returned from a subquery that only allows one column.	-412
42824	An operand of LIKE is not a string, or the first operand is not a column.	-132, -414
42825	The rows of UNION, INTERSECT, EXCEPT, or VALUES do not have compatible columns.	-415
42826	The rows of UNION, INTERSECT, EXCEPT, or VALUES do not have the same number of columns.	-421
42827	The table identified in the UPDATE or DELETE is not the same table designated by the cursor.	-509
42828	The table designated by the cursor of the UPDATE or DELETE statement cannot be modified, or the cursor is read only.	-510, -520
42829	FOR UPDATE OF is invalid, because the result table designated by the cursor cannot be modified.	-511
42830	The foreign key does not conform to the description of the parent key.	-538
42832	The operation is not allowed on system objects.	-607
42833	The qualified object name is inconsistent with the naming option.	-5016
42834	SET NULL cannot be specified, because no column of the foreign key can be assigned the null value.	-629
42835	Cyclic references cannot be specified between named derived tables.	-341
42836	The specification of a recursive, named derived table is invalid.	-345, -346
42837	The column cannot be altered, because its attributes are not compatible with the current column attributes.	-190
42841	A parameter marker cannot be a user-defined type or reference type.	-432

Table 34. **Class code 42: Syntax Error or Access Rule Violation** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
42842	A column or parameter definition is invalid, because a specified option is inconsistent with the column description.	-683
42845	An invalid use of a NOT DETERMINISTIC or EXTERNAL ACTION function was detected.	-583
42846	Cast from source type to target type is not supported.	-461
42847	An OVRDBF command was issued for one of the referenced files, but one of the parameters is not valid for SQL.	-7002
42848	Isolation level CS WITH KEEP LOCKS is not allowed.	-194
42850	A logical file is invalid in CREATE VIEW.	-7010
42851	A referenced file is not a table, view, or physical file.	-7011
42852	The privileges specified in GRANT or REVOKE are invalid or inconsistent. (For example, GRANT ALTER on a view.)	-557
42854	A result column data type in the select list is not compatible with the defined type in a typed view or materialized query table definition.	-20055
42855	The assignment of the LOB to this host variable is not allowed. The target host variable for all fetches of this LOB value for This cursor must be a locator or LOB variable.	-392
42857	A referenced file has more than one format.	-7003
42858	Operation cannot be applied to the specified object.	-7001
42860	The constraint cannot be dropped because it is enforcing a primary key or ROWID.	-784
42862	An extended dynamic statement cannot be executed against a non-extended dynamic package.	-827
42863	An undefined host variable in REXX has been detected.	-306
42866	The data type in either the RETURNS clause or the CAST FROM clause in the CREATE FUNCTION statement is not appropriate for the data type returned from the sourced function or RETURN statement in the function body.	-475
42872	FETCH statement clauses are incompatible with the cursor definition.	-225
42873	An invalid number of rows was specified in a multiple-row FETCH or multiple-row INSERT.	-221
42874	ALWCPYDTA(*NO) was specified, but a copy is necessary to implement the select-statement.	-527
42875	The schema-name portion of a qualified name must be the same name as the schema name.	-5051
42876	Different CCSIDs for keys in CREATE INDEX are only allowed with a *HEX sort sequence.	-7024
42877	The column name cannot be qualified.	-197
42878	An invalid function or procedure name was used with the EXTERNAL keyword.	-449
42879	The data type of one or more input parameters in the CREATE FUNCTION statement is not appropriate for the corresponding data type in the source function.	-492
42880	The CAST TO and CAST FROM data types are incompatible, or would always result in truncation of a fixed string.	-453

Table 34. **Class code 42: Syntax Error or Access Rule Violation** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
42881	Invalid use of a function.	-391
42882	The specific instance name qualifier is not equal to the function name qualifier.	-455
42883	No function or method was found with a matching signature.	-458
42884	No routine was found with the specified name and compatible arguments.	-440
42885	The number of input parameters specified on a CREATE FUNCTION statement does not match the number provided by the function named in the SOURCE clause.	-483
42886	The IN, OUT, or INOUT parameter attributes do not match.	-469
42887	The function is not valid in the context where it occurs.	-390
42888	The table does not have a primary key.	-539
42889	The table already has a primary key.	-624
42890	A column list was specified in the references clause, but the identified parent table does not have a unique constraint with the specified column names.	-573
42891	A duplicate UNIQUE constraint already exists.	-541
42892	The referential constraint and trigger are not allowed, because the DELETE rule and trigger event are not compatible.	-675
42893	The object or constraint cannot be dropped or authorities cannot be revoked from the object, because other objects are dependent on it.	-478, -616
42894	The value of a column or sequence attribute is invalid.	-574
42895	For static SQL, an input host variable cannot be used, because its data type is not compatible with the parameter of a procedure or user-defined function.	-301
42896	The ASP number is invalid.	-7026
42898	An invalid correlated reference or transition table was detected in a trigger.	-696
42899	Correlated references and column names are not allowed for triggered actions with the FOR EACH STATEMENT clause.	-697
428A1	Unable to access a file referenced by a file reference variable.	-452
428B3	An invalid SQLSTATE was specified.	-435
428B7	A number specified in an SQL statement is out of the valid range.	-490
428B8	The name specified on a rename is not valid.	-7029
428BA	WITHOUT RETURN cursors must not be specified in SET RESULT SETS.	-20236
428C1	Only one ROWID or IDENTITY column can be specified for a table.	-372
428C4	The number of elements on each side of the predicate operator is not the same.	-216
428C7	A ROWID or reference column specification is not valid.	-771
428C9	A ROWID or IDENTITY column cannot be specified as the target column of an INSERT or UPDATE.	-798
428D1	Unable to access a file referenced by a DATALINK value.	-358
428D2	AS LOCATOR cannot be specified for a non-LOB parameter.	-398
428D4	A cursor specified in a FOR statement cannot be referenced in an OPEN, CLOSE, or FETCH statement.	-776
428D5	The ending label does not match the beginning label.	-778

Table 34. **Class code 42: Syntax Error or Access Rule Violation** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
428D6	UNDO is not allowed for NOT ATOMIC compound statements.	-780
428D7	The condition value is not allowed.	-782
428D8	The sqlcode or sqlstate variable declaration is not valid.	-785
428DE	The PAGESIZE value is not supported.	-1583
428EC	The fullselect specified for the materialized query table is not valid.	-20058
428EK	The qualifier for a declared global temporary table name or an index on a declared global temporary table must be SESSION.	-079
428EW	The table cannot be converted to or from a materialized query table.	-20093
428F1	An SQL TABLE function must return a table result.	-20120
428F2	An integer expression must be specified on a RETURN statement in an SQL procedure.	-058
428F9	A sequence expression cannot be specified in this context.	-348
428FA	The scale of the decimal number must be zero.	-336
428FC	The length of the encryption password is not valid.	-20144
428FD	The password used for decryption does not match the password used to encrypt the data.	-20145
428FE	The data is not a result of the ENCRYPT function.	-20146
428FI	The ORDER OF clause was specified, but the referenced table designator is not ordered.	-20214
428FP	Only one INSTEAD OF trigger is allowed for each kind of operation on a view.	-20178
428FQ	An INSTEAD OF trigger must not specify a view that is defined using WITH CHECK OPTION, a view that is defined on another view that is defined WITH CHECK OPTION, or a view that is nested in a view that is defined with the WITH ROW MOVEMENT clause.	-20179
428FR	A column cannot be altered as specified.	-20180
428FT	The table is not compatible with the specified data partitioning operation.	-20183
428FY	A column cannot be added, dropped, or altered in a materialized query table.	-20235
428G0	A logical file prevents the alter of the partition attributes.	-20246
428G2	The last data partition cannot be dropped from the table.	-20251
42903	Invalid use of an aggregate function or OLAP function.	-120
42904	The SQL procedure was not created because of a compilation error.	-7032
42906	A column function in a subquery of a HAVING clause includes an expression that applies an operator to a correlated reference.	-133
42907	The string is too long.	-134
42908	The statement does not include a required column list.	-153, -343
42910	The statement is not allowed in a Compound statement.	-775
42911	A decimal divide operation is invalid, because the result would have a negative scale.	-419
42912	A column cannot be updated, because it is not identified in the UPDATE clause of the select-statement of the cursor.	-503

Table 34. **Class code 42: Syntax Error or Access Rule Violation** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
42914	The DELETE is invalid, because a table referenced in a subquery can be affected by the operation.	-536
42917	The object cannot be explicitly dropped or altered.	-658
42918	A user-defined data type cannot be created with a system-defined data type name (for example, INTEGER).	-473
42919	Nested compound statements are not allowed.	-777
42922	DROP SCHEMA cannot be executed under commitment control.	-5003
42923	Program or package must be re-created to reference an alias-name.	-7033
42924	An alias resolved to another alias rather than a table or view at the remote location.	-513
42925	Recursive named derived tables cannot specify SELECT DISTINCT and must specify UNION ALL.	-342
42926	Locators are not allowed with COMMIT(*NONE).	-7034
42928	WITH EMPTY TABLE cannot be specified.	-1596
42929	FOR ALL PARTITIONS is not allowed for an encoded vector index.	-20243
42930	The same column was identified in FOR UPDATE OF and ORDER BY.	-5021
42932	The program preparation assumptions are incorrect.	-30052
42937	The parameter must not have a subtype of mixed.	-192
42939	The name cannot be used, because the specified identifier is reserved for system use.	-457, -707
42944	The authorization ID cannot be both an owner and primary group owner.	-7028
42961	The server name specified does not match the current server.	-114
42962	A long column, LOB column, structured type column or datalink column cannot be used in an index, a key, or a constraint.	-350
42969	The package was not created and the current unit of work was rolled back, because of internal limitations or an invalid section number.	-7020
42970	COMMIT HOLD or ROLLBACK HOLD is only allowed to a DB2 UDB for iSeries application server.	-7018
42971	SQL statements cannot be executed under commitment control, because commitment control is already active to another relational database.	-7017
42972	An expression in a join-condition or ON clause of a MERGE statement references columns in more than one of the operand tables.	-338
42977	The authorization ID cannot be changed when connecting to the local server.	-7022
42978	An indicator variable is not a small integer.	-080
42981	CREATE SCHEMA is not allowed if changes are pending in the unit of work.	-7941
42984	The privilege cannot be granted to the view, because *OBJOPR or *OBJMGT authority exists on a dependent view or table, and the grantee does not have *ALLOBJ or the specified privilege on the dependent table or view.	-7027
42985	The statement is not allowed in a routine.	-577, -579, -751
42987	The statement is not allowed in a trigger.	-751
42990	A unique index or unique constraint is not allowed because the key columns are not a superset of the partitioned key columns.	-270
42995	The requested function does not apply to global temporary tables.	-526

Table 34. **Class code 42: Syntax Error or Access Rule Violation** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
42996	The partition key cannot be a datetime or floating-point column.	-328
42998	A referential constraint is not allowed because the foreign key columns are not a superset of the partitioned key columns or the node group is not the same as the parent table.	-256
42999	The query is not allowed on a distributed table or view.	-255
429B6	Rows from a distributed table cannot be redistributed because the table contains a datalink column with FILE LINK CONTROL.	-7037
429B7	A referential constraint with a delete rule of CASCADE is not allowed on a table with a DataLink column with FILE LINK CONTROL.	-7038
429BD	RETURN must be the last SQL statement of the atomic compound statement within an SQL row or table function.	-20148
429BH	A partitioned table cannot contain an identity column.	-20247
429BM	The ICU sort sequence cannot be used in this context.	-20268

Table 35. **Class code 44: WITH CHECK OPTION Violation**

SQLSTATE Value	Meaning	SQLCODE Values
44000	The INSERT or UPDATE is not allowed, because a resulting row does not satisfy the view definition.	-161

Table 36. **Class code 46: Java Errors**

SQLSTATE Value	Meaning	SQLCODE Values
46001	The URL specified on an install or replace of a jar procedure did not identify a valid jar file.	-20200
46002	The jar name specified on the install, replace, or remove of a Java procedure is not valid.	-20201
46003	The jar file cannot be removed, a class is in use by a procedure.	-20202
46007	A Java function has a Java method with an invalid signature.	-20203
46008	A Java function cannot map to a single Java method.	-20204
46501	The install or remove jar procedure for ″<jar-id>″ specified the use of a deployment descriptor.	-20207

Table 37. **Class code 51: Invalid Application State**

SQLSTATE Value	Meaning	SQLCODE Values
51002	The package corresponding to an SQL statement execution request was not found.	-805
51003	Consistency tokens do not match.	-818
51004	An address in the SQLDA is invalid.	-822
51009	COMMIT or ROLLBACK is not allowed, because commitment control has not been started.	-7007
51015	An attempt was made to execute a section that was found to be in error at bind time.	-525

Table 37. **Class code 51: Invalid Application State** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
51021	SQL statements cannot be executed until the application process executes a rollback operation.	-918
51035	A PREVVAL expression cannot be used because a value has not been generated for the sequence yet in this session.	-845
51036	An implicit connect to a remote server is not allowed because a savepoint is outstanding.	-20110
51037	The operation is not allowed because a trigger has been marked inoperative.	-7048
51039	The ENCRYPTION PASSWORD value is not set.	-20143

Table 38. **Class code 53: Invalid Operand or Inconsistent Specification**

SQLSTATE Value	Meaning	SQLCODE Values
53038	The number of key limit values is zero or greater than the number of columns in the key.	-663
53039	The PART clause of the ALTER statement is omitted or invalid.	-665
53045	The data type of the key limit constant is not the same as the data type of the column.	-678

Table 39. **Class code 54: SQL or Product Limit Exceeded**

SQLSTATE Value	Meaning	SQLCODE Values
54001	The statement is too long or too complex.	-101
54002	A string constant is too long.	-102
54004	The statement has too many table names or too many items in a SELECT or INSERT list.	-129, -840
54005	The sort key is too long, or has too many columns.	-136
54006	The result of concatenation is too long.	-137
54008	The key is too long, a column of the key is too long, or the key has too many columns.	-602, -613, -614, -631
54009	Too many users were specified in GRANT or REVOKE.	-5017
54010	The record length of the table is too long.	-101
54011	Too many columns were specified for a table, view, or table function.	-101, -680
54018	The row is too long.	-809
54019	The maximum number of late descriptors has been exceeded, probably because too many different CCSIDs were used.	-871
54021	Too many constraints, or the size of the constraint is too large.	-642
54023	The limit for the number of parameters or arguments for a function or a procedure has been exceeded.	-442, -448
54028	The maximum number of concurrent LOB handles has been reached.	-429
54035	An internal object limit was exceeded.	-7049
54038	The maximum depth of nested routines or triggers was exceeded.	-724
54044	A multiple-byte (UCS-2) sort sequence table cannot be supported in DRDA® because it is too large.	-7031

Table 39. **Class code 54: SQL or Product Limit Exceeded** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
54054	The combination of the number of table space partitions and the corresponding length of the partitioning limit key it too large.	-4701
54057	An XML element name, attribute name, namespace prefix or URL is too long.	-20326

Table 40. **Class code 55: Object Not in Prerequisite State**

SQLSTATE Value	Meaning	SQLCODE Values
55005	Recursion is only supported to a DB2 UDB for iSeries application server.	-145
55006	The object cannot be dropped, because it is currently in use by the same application process.	-615, -950
55007	The object cannot be altered, because it is currently in use by the same application process.	-951
55018	The schema cannot be dropped, because it is in the library list.	-7006
55019	The object is in an invalid state for the operation.	-7008, -20054
55029	Local program attempted to connect to a remote database.	-862
55042	The alias is not allowed because it identifies a single member of a multiple member file.	-7030
55048	Encrypted data cannot be encrypted.	-20147
55050	An object cannot be created into a protected schema.	-7052
55058	The DEBUG MODE cannot be changed for a routine that was created with DISABLE DEBUG MODE.	-20313

Table 41. **Class code 56: Miscellaneous SQL or Product Error**

SQLSTATE Value	Meaning	SQLCODE Values
56016	The ranges specified for data partitions are not valid.	-636
56084	An unsupported SQLTYPE was encountered in a select list or input list.	-351, -352
56095	A bind option is invalid.	-30104
560BF	The encryption and decryption facility has not been installed.	-20223
560C4	The option is not valid for the ARD interface.	-5027, -5028

Table 42. **Class code 57: Resource Not Available or Operator Intervention**

SQLSTATE Value	Meaning	SQLCODE Values
57005	The statement cannot be executed, because a utility or a governor time limit was exceeded.	-666
57006	The object cannot be created, because a DROP or CREATE is pending.	-679
57007	The object cannot be used, because an operation is pending.	-910
57011	Virtual storage or database resource is not available.	-904, -971, -7053
57012	A non-database resource is not available. This will not affect the successful execution of subsequent statements.	-30040

Table 42. **Class code 57: Resource Not Available or Operator Intervention** *(continued)*

SQLSTATE Value	Meaning	SQLCODE Values
57013	A non-database resource is not available. This will affect the successful execution of subsequent statements.	-30041
57014	Processing was canceled as requested.	-952
57017	Character conversion is not defined.	-332
57033	Deadlock or timeout occurred without automatic rollback.	-913
57042	DDM recursion has occurred.	-30001
57043	A local SQL application program cannot be executed on an application server.	-7021
57050	The file server is not currently available.	-357

Table 43. **Class code 58: System Error**

SQLSTATE Value	Meaning	SQLCODE Values
58002	An exit has returned an error or invalid data.	-7054
58003	An invalid section number was detected.	-144
58004	A system error (that does not necessarily preclude the successful execution of subsequent SQL statements) occurred.	-901, -4301
58008	Execution failed due to a distribution protocol error that will not affect the successful execution of subsequent DDM commands or SQL statements.	-30000
58009	Execution failed due to a distribution protocol error that caused deallocation of the conversation.	-30020
58010	Execution failed due to a distribution protocol error that will affect the successful execution of subsequent DDM commands or SQL statements.	-30021
58011	The DDM command is invalid while the bind process is in progress.	-30050
58012	The bind process with the specified package name and consistency token is not active.	-30051
58014	The DDM command is not supported.	-30070
58015	The DDM object is not supported.	-30071
58016	The DDM parameter is not supported.	-30072
58017	The DDM parameter value is not supported.	-30073
58018	The DDM reply message is not supported.	-30074
58028	The commit operation failed, because a resource in the unit of work was not able to commit its resources.	-175
58033	An unexpected error occurred while attempting to access a client driver.	-969

SQL message concepts

SQL messages are displayed when a DB2 Universal Database for iSeries returns an error or warning code to the application that uses it.

To find a specific message, SQLCODE, or SQLSTATE, try the SQL message finder.

Use Display Message Description (DSPMSGD) to display a message description

Detailed descriptions of all DB2 UDB for iSeries messages, including SQLCODEs, are available in message file QSQLMSG. You can display and print them from the Display Message Description display. The CL command to display the message description is DSPMSGD. This command shows you both the first level and the second level text for the message. The first level is a short, single sentence version of the message. The second level generally describes the reason in more detail and provides suggested solutions.

To show the message that corresponds to SQLCODE 0204, type the following command:

```
DSPMSGD RANGE(SQL0204) MSGF(QSYS/QSQLMSG)
```

Related reference

"Listing of SQL messages"
These tables list SQL messages. Use these tables to find message text, cause text, recovery text, and corresponding SQLCODEs and SQLSTATEs.

Listing of SQL messages

These tables list SQL messages. Use these tables to find message text, cause text, recovery text, and corresponding SQLCODEs and SQLSTATEs.

You can use the following tables with both positive and negative SQLCODEs. Take the absolute value of the SQLCODE, then append it to the letters SQL (for SQLCODEs less than 10 000) or the letters SQ (for SQLCODEs greater than or equal to 10 000) to determine the message identifier. Each SQLCODE corresponds to one or more SQLSTATEs.

SQ20054	
Message Text:	File &1 in &2 has DataLinks in link pending mode.
Cause Text:	The accessed file &1 has DataLinks in link pending mode for reason code &3. The reason codes are as follows: 21 - The status of the DataLinks cannot be verified. A file cannot be used for INSERT and UPDATE statements while in link pending mode.
Recovery Text:	Use the WRKPFDL (Work with Physical File DataLinks) command to determine which files have DataLinks in link pending mode. Resolve the pending links and try the request again.
SQLCODE or SQLCODEs:	-20054
SQLSTATE or SQLSTATEs:	55019

SQ20055	
Message Text:	Result column not compatible with column &1 in materialized query table.
Cause Text:	The data type of a result column in the select list is not compatible with column &1 in the materialized query table.
Recovery Text:	Ensure that the data types of all SELECT result columns are compatible with the corresponding existing columns of the materialized query table.
SQLCODE or SQLCODEs:	-20055
SQLSTATE or SQLSTATEs:	42854

SQ20058	
Message Text:	Subselect for materialized query table &1 not valid for reason code &3.

SQ20058	
Cause Text:	&1 in &2 cannot be created or altered because the SELECT statement contains a reference or expression that is not valid. Reason codes and their meanings are: • 1 - Refers to another materialized query table or to a view that refers to a materialized query table. • 2 - Refers to a declared temporary table, a table in QTEMP, a program-described file, or a non-SQL logical file in the FROM clause. • 3 - Refers to a VIEW that contains an invalid item for a materialized query table. • 4 - Contains an expression with a DataLink or a distinct type based on a DataLink where the DataLink is FILE LINK CONTROL. • 6 - Refers to a function that has an external action or that is not deterministic. • 8 - Refers to a special register. • 9 - Refers to result column that is a not an SQL data type, such as binary with precision, DBCS-ONLY, or DBCS-EITHER.
Recovery Text:	Based on the reason code, correct the SELECT statement for the materialized query table. Try the request again.
SQLCODE or SQLCODEs:	-20058
SQLSTATE or SQLSTATEs:	428EC

SQ20093	
Message Text:	Attempt to ALTER table &1 in &2 failed. Reason code &3.
Cause Text:	The ALTER request involving a materialized query table cannot be completed. Reason codes and their meanings are: • 2 - The ALTER specified DROP MATERIALIZED QUERY or ALTER MATERIALIZED QUERY, but the specified table is not a materialized query table. • 4 - The table has one or more triggers defined. • 8 - The table is referenced in the definition of another materialized query table. • 9 - The table is referenced directly or indirectly (such as through a view) in the SELECT statement. • 10 - The ALTER specified ADD MATERIALIZED QUERY, but the specified table is already a materialized query table. • 11 - The number of columns in the existing table does not match the number of columns for the result of the SELECT statement. • 12 - Column data types in the existing table do not match the column data types for the result of the SELECT statement. • 16 - The ALTER specified ADD MATERIALIZED QUERY, but the specified table has more than one member.
Recovery Text:	Change the ALTER statement, specify a different table, or change the SELECT statement to specify different result columns. Try the request again.
SQLCODE or SQLCODEs:	-20093
SQLSTATE or SQLSTATEs:	428EW

SQ20110	
Message Text:	Cannot implicitly connect to a remote site with a savepoint outstanding.
Cause Text:	The statement referenced an object at a remote DBMS when an active savepoint exists.

SQ20110	
Recovery Text:	An implicit connect to a remote server is not allowed because a savepoint is outstanding. Issue a RELEASE TO SAVEPOINT or ROLLBACK TO SAVEPOINT before retrying the statement. Committing or rolling back the transaction will also release existing savepoints.
SQLCODE or SQLCODEs:	-20110
SQLSTATE or SQLSTATEs:	51036

SQ20112	
Message Text:	The maximum number of savepoints has been reached.
Cause Text:	A resource limit has been exceeded.
Recovery Text:	Issue a RELEASE TO SAVEPOINT or ROLLBACK TO SAVEPOINT before creating additional savepoints. You may issue a COMMIT or ROLLBACK to release existing savepoints.
SQLCODE or SQLCODEs:	-20112
SQLSTATE or SQLSTATEs:	3B002

SQ20120	
Message Text:	SQL TABLE function must return a table result.
Cause Text:	The RETURN statement in an SQL TABLE function must return a table result.
Recovery Text:	Specify a query in the RETURN statement of a table function. Try the request again.
SQLCODE or SQLCODEs:	-20120
SQLSTATE or SQLSTATEs:	428F1

SQ20141	
Message Text:	Truncation of value with length &1 occurred for &2.
Cause Text:	On assignment of a value to a host variable or parameter, truncation occurred and the length of the value that was truncated is too large to be returned in the indicator variable. This situation can occur when truncation occurs on assignment of: • A value to a parameter of a remote stored procedure and the value being truncated is greater than 127 bytes. In this case, the indicator variable will contain a value of 127. • A LOB value to a host variable and the value being truncated is greater than 32767 bytes. In this case, the indicator variable will contain a value of 32767. In these cases the actual length of the truncated value cannot be returned to the application using the indicator variable. The actual length of the value is returned as message token length.
Recovery Text:	Change the length of the host variable or parameter and try the request again.
SQLCODE or SQLCODEs:	+20141
SQLSTATE or SQLSTATEs:	01004

SQ20143	
Message Text:	Encryption or decryption function failed.
Cause Text:	An encryption or decryption function failed because the encryption password value was not set.
Recovery Text:	Use the SET ENCRYPTION PASSWORD statement to set the password to be used for encryption and decryption functions. The password can also be specified as an argument of the encryption and decryption functions.
SQLCODE or SQLCODEs:	-20143
SQLSTATE or SQLSTATEs:	51039

SQ20144	
Message Text:	Encryption password length not valid.
Cause Text:	The length of the encryption password must be from 6 to 127 characters.
Recovery Text:	Correct the length of the password value. Submit the request again.
SQLCODE or SQLCODEs:	-20144
SQLSTATE or SQLSTATEs:	428FC

SQ20145	
Message Text:	The decryption function failed.
Cause Text:	The password used for decryption does not match the password used to encrypt the data. The data must be decrypted using the same password that was used to encrypt the data.
Recovery Text:	Ensure that the same password is used to encrypt and decrypt the data.
SQLCODE or SQLCODEs:	-20145
SQLSTATE or SQLSTATEs:	428FD

SQ20146	
Message Text:	The decryption function failed. The data is not encrypted.
Cause Text:	The data must be a result of the ENCRYPT, ENCRYPT_RC2, or ENCRYPT_TDES function.
Recovery Text:	Ensure that the data is a result of the ENCRYPT, ENCRYPT_RC2, or ENCRYPT_TDES function. Data passed to a decryption function must be encrypted data.
SQLCODE or SQLCODEs:	-20146
SQLSTATE or SQLSTATEs:	428FE

SQ20147	
Message Text:	The ENCRYPT function failed. Data is already encrypted.
Cause Text:	An attempt was made to encrypt data that is already encrypted.
Recovery Text:	Ensure that the data has not already been encrypted.

SQ20147	
SQLCODE or SQLCODEs:	-20147
SQLSTATE or SQLSTATEs:	55048

SQ20148	
Message Text:	RETURN statement must be specified in function &1 in &2.
Cause Text:	A RETURN statement must be specified in an SQL function.
Recovery Text:	Specify a RETURN statement. Try the request again.
SQLCODE or SQLCODEs:	-20148
SQLSTATE or SQLSTATEs:	429BD

SQ20178	
Message Text:	INSTEAD OF trigger not created for view &1.
Cause Text:	View &1 in &2 already has an INSTEAD OF trigger defined for the specified operation. A view can have only one INSTEAD OF trigger defined for each of the INSERT, UPDATE, and DELETE operations.
Recovery Text:	Drop the existing trigger. Combine the function for the existing trigger and the new trigger into a single trigger. Create the trigger again.
SQLCODE or SQLCODEs:	-20178
SQLSTATE or SQLSTATEs:	428FP

SQ20179	
Message Text:	Trigger cannot be created for view &1 that uses a WITH CHECK OPTION.
Cause Text:	View &1 in &2 specifies a WITH CHECK OPTION or is dependent on a view which has a WITH CHECK OPTION. An INSTEAD OF trigger cannot be defined for a view that has a WITH CHECK OPTION.
Recovery Text:	Remove the WITH CHECK OPTION from the view.
SQLCODE or SQLCODEs:	-20179
SQLSTATE or SQLSTATEs:	428FQ

SQ20180	
Message Text:	Column &1 cannot be altered.
Cause Text:	Column &1 is a partitioning key column for table &2 in &3 and cannot be altered.
Recovery Text:	Change the table to a non-partitioned table before attempting to alter the column or remove the ALTER COLUMN clause from the statement. Try the request again.
SQLCODE or SQLCODEs:	-20180
SQLSTATE or SQLSTATEs:	428FR

SQ20183	
Message Text:	ALTER TABLE partitioning clause not valid for &1.
Cause Text:	Table &1 in &2 cannot be altered. One of the following errors has occurred: • ADD PARTITION BY was specified on an ALTER TABLE statement, but the table is already a partitioned table or is a physical file. • ADD PARTITION, ALTER PARTITION, or DROP PARTITION was specified on an ALTER TABLE statement, but the table is not a partitioned table. • DROP PARTITION of a RANGE partition was specified on the ALTER TABLE statement, but the PRESERVE ROWS clause was used. The PRESERVE ROWS clause is only valid for partitioned tables that have HASH partitioning.
Recovery Text:	Change the statement to one that is valid or specify a different table. Try the request again.
SQLCODE or SQLCODEs:	-20183
SQLSTATE or SQLSTATEs:	428FT

SQ20200	
Message Text:	The install or replace of &1 in &2 failed because &3 could not be located.
Cause Text:	The URL specified on the install or replace jar procedure did not identify a valid jar file.
Recovery Text:	Reissue the install or replace jar procedure with a URL that identifies a valid jar file.
SQLCODE or SQLCODEs:	-20200
SQLSTATE or SQLSTATEs:	46001

SQ20201	
Message Text:	The install, replace, or remove of &1 in &2 failed because the jar name is not valid.
Cause Text:	The jar name specified on the install, replace, or remove jar procedure is not valid. For example, the jar id may be of the improper format, may not exist to be replaced or removed, or can not be installed as it already exists.
Recovery Text:	Ensure the jar name is of the correct format. If the jar id exists, it may need to be removed before it can be installed. For the remove or replace procedures, ensure the jar id exists.
SQLCODE or SQLCODEs:	-20201
SQLSTATE or SQLSTATEs:	46002

SQ20202	
Message Text:	The replace or remove of &1 in &2 failed because &3 is in use.
Cause Text:	The specified class in the jar file is currently in use by a defined procedure, or the replacement jar file does not contain the specified class for which a procedure is defined.
Recovery Text:	Ensure all procedures referencing the classes being removed are dropped and resubmit the replace or remove procedure.
SQLCODE or SQLCODEs:	-20202

SQ20202	
SQLSTATE or SQLSTATEs:	46003

SQ20203	
Message Text:	Signature not valid for Java method in user defined function or procedure &1 in &2.
Cause Text:	The signature of the Java method used to implement the function or procedure is not valid. For example, the method may have parameters that are not compatible with the parameters on the corresponding CREATE statement or the method for a procedure may specify a return value.
Recovery Text:	Reissue the corresponding CREATE statement specifying parameters that match the Java method, or correct the parameters or return type of the Java method and rebuild the class.
SQLCODE or SQLCODEs:	-20203
SQLSTATE or SQLSTATEs:	46007

SQ20204	
Message Text:	The user defined function or procedure &1 in &2 was unable to map to a single Java method.
Cause Text:	The identified function or procedure either failed to find a matching Java method, or found more than 1 matching Java method.
Recovery Text:	Correct either the Java method or corresponding create statement so that the function or procedure call resolves to a single Java method.
SQLCODE or SQLCODEs:	-20204
SQLSTATE or SQLSTATEs:	46008

SQ20205	
Message Text:	User defined function or procedure &1 in &2 has an input argument with a null value.
Cause Text:	A function created with CALLED ON NULL INPUT or a procedure has an input parameter with a null value but the Java data type for this argument does not support null values. Examples of Java data types that do not support null values are BOOLEAN, BYTE, SHORT, INT, LONG, or DOUBLE.
Recovery Text:	If the method is to be called with null values, ensure the input Java types are capable of accepting a null value. If &1 is a function, RETURNS NULL ON NULL INPUT may be specified on the CREATE FUNCTION statement.
SQLCODE or SQLCODEs:	-20205
SQLSTATE or SQLSTATEs:	39004

SQ20206	
Message Text:	The procedure &1 in &2 returned too many result sets.
Cause Text:	The specified procedure returned more results sets than were specified on the procedure definition.

SQ20206	
Recovery Text:	Modify the procedure to return fewer result sets, or drop and re-create the procedure specifying the correct number of result sets.
SQLCODE or SQLCODEs:	+20206
SQLSTATE or SQLSTATEs:	0100E

SQ20207	
Message Text:	The install or remove jar procedure for &1 in &2 specified the use of a deployment descriptor.
Cause Text:	The DEPLOY or UNDEPLOY parameter of the install or replace jar procedure was non-zero; this parameter is not supported and must be zero.
Recovery Text:	Reissue the procedure with the DEPLOY or UNDEPLOY parameter set to zero.
SQLCODE or SQLCODEs:	-20207
SQLSTATE or SQLSTATEs:	46501

SQ20214	
Message Text:	Table designator &1 is not ordered.
Cause Text:	The ORDER OF clause was specified, but table designator &1 is not ordered.
Recovery Text:	Correct the ORDER OF clause. Try the request again.
SQLCODE or SQLCODEs:	-20214
SQLSTATE or SQLSTATEs:	428FI

SQ20223	
Message Text:	The encryption or decryption function failed. Encryption facility not available.
Cause Text:	An attempt to use function ENCRYPT, ENCRYPT_RC2, ENCRYPT_TDES, DECRYPT_BIN, DECRYPT_BINARY, DECRYPT_BIT, DECRYPT_CHAR, DECRYPT_DB, or GETHINT failed because the following product is not installed: 5722-AC3 IBM® Cryptographic Access Provider 128-bit for iSeries.
Recovery Text:	Install the encryption facility before using any encryption or decryption functions.
SQLCODE or SQLCODEs:	-20223
SQLSTATE or SQLSTATEs:	560BF

SQ20224	
Message Text:	Encrypted data that was originally a binary string cannot be decrypted to a character string.
Cause Text:	The DECRYPT_CHAR function was invoked with an argument that represents an encrypted string that was originally a binary string. A binary string cannot be cast to a character string using the DECRYPT_CHAR function.
Recovery Text:	The encrypted value can only be decrypted using the DECRYPT_BINARY function.

SQ20224	
SQLCODE or SQLCODEs:	+20224
SQLSTATE or SQLSTATEs:	01658

SQ20226	
Message Text:	The maximum number of stacked diagnostics areas has been exceeded.
Cause Text:	The storage available to hold condition information items or to hold stacked diagnostics has been exceeded.
Recovery Text:	Modify the application logic so that fewer SQL warnings occur, or specify additional condition areas using the SET TRANSACTION statement.
SQLCODE or SQLCODEs:	-20226
SQLSTATE or SQLSTATEs:	0Z001

SQ20227	
Message Text:	AS clause required for XMLATTRIBUTES or XMLFOREST function.
Cause Text:	The AS clause must be specified for argument &1 of the XMLATTRIBUTES or XMLFOREST function because the argument is not named. Argument &1 is specified as &2.
Recovery Text:	Specify an AS clause to name the argument. Try the request again.
SQLCODE or SQLCODEs:	-20227
SQLSTATE or SQLSTATEs:	42633

SQ20228	
Message Text:	A STACKED diagnostic is not available.
Cause Text:	A STACKED diagnostic is only accessible from a handler.
Recovery Text:	Modify the application logic so that the STACKED diagnostic is only accessed from within a handler.
SQLCODE or SQLCODEs:	-20228
SQLSTATE or SQLSTATEs:	0Z002

SQ20235	
Message Text:	Alter of materialized query table &1 in &2 not valid.
Cause Text:	A column of a materialized query table cannot be added, dropped, or altered.
Recovery Text:	Change the statement to one that is valid or specify another table. Try the request again.
SQLCODE or SQLCODEs:	-20235
SQLSTATE or SQLSTATEs:	428FY

SQ20236	
Message Text:	Cursor &1 cannot be returned.
Cause Text:	A SET RESULT SETS statement references cursor &1 that was declared with the WITHOUT RETURN attribute. Cursor &1 cannot be specified in a SET RESULT SETS statement.
Recovery Text:	Remove the reference to the cursor from the SET RESULT SETS statement. Try the request again.
SQLCODE or SQLCODEs:	-20236
SQLSTATE or SQLSTATEs:	428BA

SQ20243	
Message Text:	Index &1 cannot be both an encoded vector index and span all partitions.
Cause Text:	An attempt was made to create index &1 as an encoded vector index and have it span all partitions of a partitioned table. An encoded vector index can only be created over a single partition of a partitioned table or over a non-partitioned table.
Recovery Text:	Change the CREATE INDEX statement to remove either the NOT PARTITIONED clause or the ENCODED VECTOR clause. Try the request again.
SQLCODE or SQLCODEs:	-20243
SQLSTATE or SQLSTATEs:	42929

SQ20246	
Message Text:	&1 in &2 cannot be altered.
Cause Text:	Table &1 is a partitioned table and cannot be altered because DDS-created logical file &3 in &4 is based on the table.
Recovery Text:	Remove the logical file. Try the request again.
SQLCODE or SQLCODEs:	-20246
SQLSTATE or SQLSTATEs:	428G0

SQ20247	
Message Text:	Table &1 cannot be partitioned and contain an identity or ROWID column.
Cause Text:	An attempt was made to create or alter table &1 in &2 in such a way that would create a partitioned table that contained an identity column or ROWID column. A table cannot be both partitioned and contain an identity column or ROWID column.
Recovery Text:	Create the table with either an identity or ROWID column or as a partitioned table.
SQLCODE or SQLCODEs:	-20247
SQLSTATE or SQLSTATEs:	429BH

SQ20251	
Message Text:	Partition &1 cannot be dropped.

SQ20251	
Cause Text:	An attempt was made to drop partition &1 but it is the last partition in the table. The partition table must contain at least one partition.
Recovery Text:	Use the DROP TABLE statement to drop the table.
SQLCODE or SQLCODEs:	-20251
SQLSTATE or SQLSTATEs:	428G2

SQ20268	
Message Text:	Sort sequence cannot be applied. Reason code &4.
Cause Text:	Sort sequence table &2 in library &3 cannot be applied for reason code &4. The reason codes and their meanings follow: • 1 - The result of applying the sort sequence table to field &1 is greater than 32,742 bytes. • 2 - The sort sequence table cannot be applied to the following functions: LOCATE, POSITION and POSSTR. • 3 - The sort sequence table cannot be applied to a LIKE predicate or %WLDCRD function.
Recovery Text:	Sort sequence table &2 is an International Components of Unicode (ICU) table. Recovery for reason code 1: ICU sort sequence table results are longer than the fields they operate on. If the field can contain SBCS data, the result of applying the sort sequence table can be 6 times the length of the field. Applying this sort sequence table to a DBCS field can be 3 times the length of the field. Either specify a smaller field or a non-ICU type sort sequence table. Recovery for the other reason codes: Either omit the function or specify a non-ICU type of sort sequence table.
SQLCODE or SQLCODEs:	-20268
SQLSTATE or SQLSTATEs:	429BM

SQ20275	
Message Text:	XML name &1 not valid.
Cause Text:	The XML name &1 is not valid for reason code &2. Reason codes are: • 1 - `xmlns` is used as an XML attribute name or as an XML namespace prefix for an XML element or attribute name. • 2 - The XML namespace prefix in a qualified XML name is not declared within its scope. • 3 - The XML name for an XML element or XML attribute is not an XML QName.
Recovery Text:	Specify a valid XML name. Try the request again.
SQLCODE or SQLCODEs:	-20275
SQLSTATE or SQLSTATEs:	42634

SQ20276	
Message Text:	XML namespace prefix &1 not valid.

SQ20276	
Cause Text:	The XML namespace prefix &1 is not valid for reason code &2. Reason codes are: • 1 - The XML namespace prefix is not an XML NCName. • 2 - The name `xml` or `xmlns` cannot be used as an XML namespace prefix. • 3 - The XML namespace prefix is not unique.
Recovery Text:	Specify a valid and unique name for the XML namespace prefix. Try the request again.
SQLCODE or SQLCODEs:	-20276
SQLSTATE or SQLSTATEs:	42635

SQ20298	
Message Text:	VALUE specified for GET DESCRIPTOR is greater than current COUNT.
Cause Text:	The item number supplied in VALUE for GET DESCRIPTOR is greater than the current COUNT for the descriptor.
Recovery Text:	This is a warning only. No information requested for the descriptor item provided in VALUE was returned.
SQLCODE or SQLCODEs:	+20298
SQLSTATE or SQLSTATEs:	02505

SQ20299	
Message Text:	Item &1 already set for descriptor.
Cause Text:	Descriptor item &1 was already specified for this SET DESCRIPTOR statement. Each item can be set only one time.
Recovery Text:	Remove the duplicate item. Try the request again.
SQLCODE or SQLCODEs:	-20299
SQLSTATE or SQLSTATEs:	42747

SQ20313	
Message Text:	DEBUG MODE cannot be changed for routine &1 in &2.
Cause Text:	Routine &1 in &2 was created with a DEBUG MODE of DISABLE. The DEBUG MODE cannot be altered.
Recovery Text:	Drop the routine and re-create it with DEBUG MODE ALLOW or DEBUG MODE DISALLOW.
SQLCODE or SQLCODEs:	-20313
SQLSTATE or SQLSTATEs:	55058

SQ20326	
Message Text:	XML name &1 too long.

SQ20326	
Cause Text:	The XML name &1 is too long. The maximum length of an XML element name, and XML attribute name, and an XML namespace prefix is 128. The maximum length of an XML namespace URL is 1024.
Recovery Text:	Ensure the name is not too long. Try the request again.
SQLCODE or SQLCODEs:	-20326
SQLSTATE or SQLSTATEs:	54057

SQ30000	
Message Text:	Distributed Relational Database Architecture™ (DRDA) protocol error.
Cause Text:	Command or SQL statement failed due to a distribution protocol error that will not affect subsequent commands or SQL statements. The protocol error is &1 with a location code of &2. The location codes are: • '01'X - The error was detected at the application requester. • '02'X - The error was detected at the application server. A possible list of protocol errors is: • '121C'X - Not authorized to command. • '1245'X - Conversation protocol error. The error code is &3. • '124C'X - Data stream syntax error. The error code is &3. • '1254'X - Unexpected error condition. The error code is &3. • '125F'X - Application server does not support function requested. • '2202'X - Cursor not open. • '2204'X - Relational database not connected. • '2207'X - Relational database already connected. • '220A'X - Data descriptor not valid. The error code is &3. • '220E'X - Data descriptor did not match data. • '220F'X - Cursor already open. • '221D'X - Command not valid for conversation type. If the protocol error does not appear in the list, refer to the DDM Architecture Reference for DDM code point &1.
Recovery Text:	If the protocol error is '121C'X, then obtain the authorization required to use this command or SQL statement at the application server. If the protocol error is not '121C'X, then report the problem using Analyze Problem (ANZPRB).
SQLCODE or SQLCODEs:	-30000
SQLSTATE or SQLSTATEs:	58008

SQ30001	
Message Text:	Call to distributed SQL program not allowed.
Cause Text:	An attempt was made to use Submit Remote Command (SBMRMTCMD) to call a distributed SQL program from a Distributed Data Management (DDM) target job.
Recovery Text:	Call the SQL program from a job that is not a DDM target job.
SQLCODE or SQLCODEs:	-30001

SQ30001	
SQLSTATE or SQLSTATEs:	57042

SQ30020	
Message Text:	Distributed Relational Database Architecture (DRDA) protocol error.
Cause Text:	Command or SQL statement failed due to a distribution protocol error that will affect subsequent SQL statements or commands. The application has been disconnected and the process is in an unconnected state. The protocol error is &1 with a location code of &2. The location codes are: • '01'X - The error was detected at the application requester. • '02'X - The error was detected at the application server. A possible list of protocol errors is: • '1218'X - Unsupported function requested. The error code is &3. • '1232'X - Unexpected permanent error. • '1245'X - Conversation protocol error. The error code is &3. • '1254'X - Unexpected error condition. The error code is &3. If the protocol error does not appear in the list, refer to the DDM Architecture Reference for DDM code point &1.
Recovery Text:	Report the problem using Analyze Problem (ANZPRB).
SQLCODE or SQLCODEs:	-30020
SQLSTATE or SQLSTATEs:	58009

SQ30021	
Message Text:	Distributed relational database not supported by the remote system.
Cause Text:	An attempt was made to connect to a Distribute Data Management (DDM) server that does not support Distributed Relational Database Architecture (DRDA). On the remote system, the DDM manager class was &1 with manager level &2.
Recovery Text:	Change the program to connect to a relational database which supports DRDA.
SQLCODE or SQLCODEs:	-30021
SQLSTATE or SQLSTATEs:	58010

SQ30040	
Message Text:	DDM resources at relational database &1 unavailable.
Cause Text:	SQL statement or command failed due to an unavailable Distributed Data Management (DDM) resource that will not affect subsequent SQL statements and commands. DDM resource &2 is unavailable at relational database &1 with location code &3. The resource codes are: '1409'X - Communication buffer. The location codes are: • '01'X - Application requester or the local system. • '02'X - Application server.
Recovery Text:	Free the DDM resource and try the request again.
SQLCODE or SQLCODEs:	-30040

SQ30040	
SQLSTATE or SQLSTATEs:	57012

SQ30041	
Message Text:	DDM resources at relational database &1 unavailable.
Cause Text:	SQL statement or command failed due to an unavailable Distributed Data Management (DDM) resource that will affect subsequent commands and SQL statements. The application has been disconnected and the process is in an unconnected state. DDM resource &2 is unavailable at relational database &1 with location code &3. The resource codes are: • '1409'X - Communication buffer. • '1410'X - Local build space. • '1411'X - DDM diagnostic area. • '1412'X - Local build space. • '1412'X - LOB build space. The location codes are: • '01'X - Application requester or the local system. • '02'X - Application server.
Recovery Text:	Free the DDM resource and try the request again.
SQLCODE or SQLCODEs:	-30041
SQLSTATE or SQLSTATEs:	57013

SQ30050	
Message Text:	DDM command &1 not valid while bind process in progress.
Cause Text:	An attempt was made to run Distributed Data Management (DDM) command &1. This command is not valid while a bind process is in progress.
Recovery Text:	Report this problem using the Analyze Problem (ANZPRB) command.
SQLCODE or SQLCODEs:	-30050
SQLSTATE or SQLSTATEs:	58011

SQ30051	
Message Text:	Bind process for specified package name and consistency token not active.
Cause Text:	Attempted to run a BNDSQLSTT or ENDBND Distributed Data Management (DDM) command for a bind process that was not active.
Recovery Text:	Report this problem using the Analyze Problem (ANZPRB) command.
SQLCODE or SQLCODEs:	-30051
SQLSTATE or SQLSTATEs:	58012

SQ30052	
Message Text:	Program preparation assumptions not correct.

SQ30052	
Cause Text:	The application requester did not understand the SQL statement and assumed all host variables were input, but this assumption was not correct.
Recovery Text:	Refer to the CRTSQLxxx (where xxx=CBL, PKG, PLI, RPG, CI, CPPI, RPGI, or CBLI) listing to find all SQL statements that were not recognized. Remove all unrecognized SQL statements that contain output host variables. Precompile the program again.
SQLCODE or SQLCODEs:	-30052
SQLSTATE or SQLSTATEs:	42932

SQ30053	
Message Text:	Not authorized to create package for owner &1.
Cause Text:	Attempt to create the package failed because you are not authorized to owner &1.
Recovery Text:	Obtain the required authorization to &1 and try again.
SQLCODE or SQLCODEs:	-30053
SQLSTATE or SQLSTATEs:	42506

SQ30060	
Message Text:	User is not authorized to relational database &1.
Cause Text:	If relational database &1 is DB2 UDB for iSeries, a user exit program denied access to the user, or a failure in the user exit program occurred.
Recovery Text:	Obtain authorization to relational database &1 and try the request again.
SQLCODE or SQLCODEs:	-30060
SQLSTATE or SQLSTATEs:	08004

SQ30061	
Message Text:	Relational database &1 not found.
Cause Text:	Relational database &1 was either not in the relational database directory or defined at the remote location.
Recovery Text:	Do one of the following: • Use the Add Relational Database Directory Entry (ADDRDBDIRE) command to add the relational database name to the relational database directory. • Change the relational database name to match the relational database directory entry. • Verify the relational database name is defined on the remote location.
SQLCODE or SQLCODEs:	-30061
SQLSTATE or SQLSTATEs:	08004

SQ30070	
Message Text:	Distributed Data Management (DDM) command &1 not supported.

SQ30070	
Cause Text:	The remote system does not support the DDM command &1. If the DDM command is '2012'X, the remote system does not support the SQL DESCRIBE TABLE statement. If the DDM command is not '2012'X, to determine which command is not supported, see the DDM Architecture Reference. For a list of DDM commands, refer to the Architecture Books section of the Bibliography in the Distributed Database Programming topic in the Information Center, http://www.ibm.com/eserver/iseries/infocenter.
Recovery Text:	Remove the SQL statement from the program and precompile the program.
SQLCODE or SQLCODEs:	-30070
SQLSTATE or SQLSTATEs:	58014

SQ30071	
Message Text:	Distributed Data Management (DDM) object &1 not supported.
Cause Text:	DDM object &1 was not supported.
Recovery Text:	See previous messages for more information. See the DDM Architecture Reference for additional information about code point &1.
SQLCODE or SQLCODEs:	-30071
SQLSTATE or SQLSTATEs:	58015

SQ30072	
Message Text:	Distributed Data Management (DDM) parameter &1 not supported.
Cause Text:	DDM parameter &1 is not supported. The location code is &2 with an error code of &3. The location codes are: • '01'X - The error was detected at the application requester. • '02'X - The error was detected at the application server.
Recovery Text:	See previous messages for more information. See the DDM Architecture Reference for additional information about DDM parameter &1.
SQLCODE or SQLCODEs:	-30072
SQLSTATE or SQLSTATEs:	58016

SQ30073	
Message Text:	Distributed Data Management (DDM) parameter value &1 not supported.

SQ30073	
Cause Text:	DDM parameter value &1 is not supported. The location code is &2 with an error code of &3. The location code are: • ’01’X - The error was detected at the application requester. • ’02’X - The error was detected at the application server. A possible list of DDM parameter values is: • ’0035’X - The SBCS CCSID is not supported. • ’119C’X - The SBCS CCSID is not supported. • ’2112’X - The schema name or package name is longer than the maximum supported by the application server. • ’2120’X - The string delimiter is not supported. • ’2121’X - The decimal delimiter is not supported. • ’2128’X - The schema name is longer than the maximum supported by the application server. • ’2131’X - The userid is longer than the maximum supported by the application server. If &1 is not in the list above, refer to the DDM Architecture Reference for a description of the parameter value that was not supported.
Recovery Text:	See previous messages for more information. Change your job or SQL program to send a value that is supported by the application server and try again.
SQLCODE or SQLCODEs:	-30073
SQLSTATE or SQLSTATEs:	58017

SQ30074	
Message Text:	Distributed Data Management (DDM) reply message &1 not supported.
Cause Text:	DDM reply message &1 was not supported.
Recovery Text:	See previous messages for more information. See the DDM Architecture Reference for additional information about code point &1.
SQLCODE or SQLCODEs:	-30074
SQLSTATE or SQLSTATEs:	58018

SQ30080	
Message Text:	Communication error occurred during distributed database processing.

SQ30080	
Cause Text:	A communication error occurred. Possible reasons include: • The remote system is not available. • The communications network is not available. • The userid used to start the connection may not exist on the remote system. • The remote system may require the password to be encrypted. • The password may not be valid for the userid. The characters and case of the password specified must match exactly the password on the remote system. • A server authorization entry for the remote system, if used, may be incorrect. Server names must be in uppercase. The QRETSVRSEC system value must be set to '1' to retain passwords. If message CPE3425 (connection refused) preceeds this message, the cause may be: • The DDM/DRDA TCP/IP server is not started on the remote system. • An incorrect port was specified for the remote system. • The remote system is restricting DRDA ports. • The SOCKS server, if used, is not configured properly. The APPC major return code is &1 and the minor return code is &2. For TCP/IP, both return codes will be 00. If the return codes are not 00, their meaning can be found in the Information Center.
Recovery Text:	See previous messages for more information. Check the status of the remote system and the communications network for possible problems. If the application server is an iSeries, check QSYSOPR message queue for error messages.
SQLCODE or SQLCODEs:	-30080
SQLSTATE or SQLSTATEs:	08001

SQ30082	
Message Text:	Authorization failure on distributed database connection attempt.
Cause Text:	A connection attempt failed with reason code &2. The reason codes and their meanings are as follows: • 0 - Unknown cause. • 1 - Password expired. • 2 - Password not valid. • 3 - Password missing. • 4 - Protocol violation. • 5 - User ID not found. • 6 - User ID not valid. For a DB2 UDB for iSeries server, this could mean a damaged user profile or PASSWORD(*NONE). • 7 - User ID revoked or disabled. • 15 - Security processing at the server failed. • 16 - The new password is not valid. • 17 - The security mechanism requested by the client is not supported or allowed at the server. See recovery information below. • 22 - Security processing at the client failed. • 23 - CCSID conversion of the password failed.

SQ30082	
Recovery Text:	Correct the problem indicated by the reason code, if possible, and attempt to connect again. A common cause for reason code 17 is that the server requires a password, but because the client does not have a password to send, sends only a user ID. Or, the server requires an encrypted password and the client did not send an encrypted password. A password can be supplied by the user in either of the following two ways: • By using the USER ... USING ... clause on the SQL CONNECT statement. • By using the ADDSVRAUTE CL command to add a server authorization entry for the remote server under the user's profile. The server name (DRDA RDB NAME) must be entered in uppercase. The DRDA server can be configured with the CHGDDMTCPA CL command to not require a password, or to not require an encrypted password.
SQLCODE or SQLCODEs:	-30082
SQLSTATE or SQLSTATEs:	08001

SQ30089	
Message Text:	Communication error occurred during DB2 Multisystem processing.
Cause Text:	A communication error occurred. A possible list of reasons may include: • The remote system is not available. • The communications network is not available. • The userid used to start the connection may not exist on the remote system.
Recovery Text:	See previous messages for more information. Check the status of the remote system and the communications network for possible problems. Check QSYSOPR message queue for error messages.
SQLCODE or SQLCODEs:	-30089
SQLSTATE or SQLSTATEs:	08001

SQ30090	
Message Text:	Change request not valid for read-only application server.
Cause Text:	Application requester requested a read-only application server. The running of a statement which is not valid for a read-only application server was attempted. This message can occur when initially attempting to connect to a non-iSeries server using interactive SQL with the COMMIT option set to *NONE.
Recovery Text:	Remove the change request from the program and try again. If the problem is due to the use of COMMIT(*NONE) to a non-iSeries server, change to a different commitment control level and try again.
SQLCODE or SQLCODEs:	-30090
SQLSTATE or SQLSTATEs:	25000, 2D528, 2D529

SQ30104	
Message Text:	Bind option not valid.

SQ30104	
Cause Text:	The value &2 is not valid with generic bind option &1. Valid values are: • AS400NAMING - SQL or SYSTEM • OS400NAMING - SQL or SYSTEM • SORTSEQ - JOBRUN or HEX • MAXSCALE - between 0 and 31 • MINDIVSCALE - between 0 and 9, MINDIVSCALE must not exceed MAXSCALE
Recovery Text:	Correct the generic bind option value and try the request again.
SQLCODE or SQLCODEs:	-30104
SQLSTATE or SQLSTATEs:	56095

SQ30106	
Message Text:	The input data for a multiple row insert is not valid.
Cause Text:	A data error has been detected on the client when processing a row for a multi-row input operation. A null indicator value of -2 (X'FE') was received from the client which indicates that the server should return an error SQLCA containing SQLSTATE 22527 over a null row. For an atomic multi-row input operation, such an error terminates processing of the multi-row input request, and any changes that have resulted from this request will be undone. For a non-atomic multi-row operation, such an error terminates processing of the multi-row input request, but any changes for this request are not undone.
Recovery Text:	Determine which part of the multi-row input operation is in error and retry the request.
SQLCODE or SQLCODEs:	-30106
SQLSTATE or SQLSTATEs:	22527

SQ30107	
Message Text:	A request to release record read locks on close of cursor was ignored.
Cause Text:	DB2 UDB for iSeries does not support this feature. To avoid this warning do the following. For embedded SQL remove the WITH RELEASE clause from the CLOSE. For CLI remove the option to define cursors with release.
Recovery Text:	
SQLCODE or SQLCODEs:	+30107
SQLSTATE or SQLSTATEs:	01662

SQL0007	
Message Text:	Character &1 (HEX &2) not valid in SQL statement.
Cause Text:	The character &1 specified in the SQL statement is not permitted. The hexadecimal representation for the character is &2. The character is ignored by the precompiler and processing of the statement continues.
Recovery Text:	Correct the character. The character may need to be enclosed either in quotation marks or apostrophes. Precompile the program again.
SQLCODE or SQLCODEs:	-007

SQL0007	
SQLSTATE or SQLSTATEs:	42601

SQL0010	
Message Text:	String constant beginning &1 not delimited.
Cause Text:	The string delimiter is missing in the constant beginning with &1. The string is treated as if it were delimited by the end of the source file.
Recovery Text:	Delimit the string constant. Check for any missing or extra quotation marks and apostrophes. These errors are likely to cause other errors. Some statements may not have been processed as the result of either missing or extra string delimiters. Precompile the program again.
SQLCODE or SQLCODEs:	-010
SQLSTATE or SQLSTATEs:	42603

SQL0011	
Message Text:	Comment not closed.
Cause Text:	A comment is not closed correctly with a comment delimiter. The comment is treated as if it were closed by the end of source file.
Recovery Text:	Make certain that the comment delimiter is specified properly and that it is between the specified margins. This error is likely to cause other errors. Some statements may not have been processed as a result of a missing comment delimiter. Precompile the program again.
SQLCODE or SQLCODEs:	-011
SQLSTATE or SQLSTATEs:	42601

SQL0012	
Message Text:	Correlation without qualification occurred for column &1 to table &2.
Cause Text:	Column &1 which occurs in a subselect, is not explicitly qualified, and occurs in table &2 in library &3 specified in the FROM clause of an outer subselect or as the target of an update or delete operation. Consequently, the reference to the column in the subselect is an outer reference, and correlation will occur.
Recovery Text:	Ensure you intended to use the correlation. If you did not intend to use the correlation, the column does not exist in any of the tables or views identified in the FROM clause of the same level of the subselect that column &1 was referenced. Since it is a good practice to explicitly qualify any intended correlated references, it is recommended that the statement be changed so that the column &1 is qualified with a table designator.
SQLCODE or SQLCODEs:	+012
SQLSTATE or SQLSTATEs:	01545

SQL0029	
Message Text:	INTO clause missing from embedded statement.

SQL0029	
Cause Text:	SELECT and VALUES INTO statements embedded in a program must have an INTO clause to specify where the results of the statement are to be placed. A dynamic VALUES INTO statement must have an INTO clause.
Recovery Text:	Add the INTO clause to the statement and try the request again.
SQLCODE or SQLCODEs:	-029
SQLSTATE or SQLSTATEs:	42601

SQL0030	
Message Text:	Number of host variables less than result values.
Cause Text:	The number of host variables specified in the INTO clause is less than the number of result values. If the program is run, only the variables specified will have values assigned to them.
Recovery Text:	If all values should be received, specify the proper number of host variables. Precompile the program again.
SQLCODE or SQLCODEs:	+030
SQLSTATE or SQLSTATEs:	01503

SQL0051	
Message Text:	Cursor or procedure &1 previously declared.
Cause Text:	One of the following has occurred: • Cursor &1 has already been specified in a previous DECLARE CURSOR statement. A cursor name must be unique within the program. • Procedure &1 specified on a CALL statement is ambiguous.
Recovery Text:	Make certain that the cursor names on all DECLARE CURSOR statements and the procedure names on all DECLARE PROCEDURE statements are unique in the program. Precompile the program again.
SQLCODE or SQLCODEs:	-051
SQLSTATE or SQLSTATEs:	3C000

SQL0057	
Message Text:	RETURN statement in an SQL function must return a value.
Cause Text:	A RETURN statement is specified in an SQL function without a return value. The return value must be specified.
Recovery Text:	Add a value to return. Try the request again.
SQLCODE or SQLCODEs:	-057
SQLSTATE or SQLSTATEs:	42631

SQL0058	
Message Text:	Value specified on RETURN statement must be an integer.
Cause Text:	A RETURN statement is specified in an SQL procedure but the value specified is not integer. The return value for an SQL procedure must be integer.
Recovery Text:	Specify an integer value on the RETURN statement. Try the request again.
SQLCODE or SQLCODEs:	-058
SQLSTATE or SQLSTATEs:	428F2

SQL0060	
Message Text:	Value &3 for argument &1 of &2 function not valid.
Cause Text:	The length or scale specified as &3 for argument &1 of the &2 function is not valid. The length specified for numeric values must be an unsigned integer from 1 through 63. The scale specified for numeric values must be an unsigned integer between 0 and the specified length.
Recovery Text:	Correct the length or scale specified for the function. Try the request again.
SQLCODE or SQLCODEs:	-060
SQLSTATE or SQLSTATEs:	42815

SQL0074	
Message Text:	Descriptor COUNT &1 is not valid.
Cause Text:	A FETCH was attempted when the descriptor COUNT value does not match the number of columns in the result set.
Recovery Text:	Specify a different descriptor for the FETCH or specify a different COUNT value using the SET DESCRIPTOR statement.
SQLCODE or SQLCODEs:	-074
SQLSTATE or SQLSTATEs:	07008

SQL0075	
Message Text:	Descriptor item number &1 is not correct.
Cause Text:	Descriptor item number &1 is not correct for one of the following reasons: • The value given for the maximum number of items on an ALLOCATE DESCRIPTOR statement is not between 1 and 8000. • The item number given on a GET DESCRIPTOR or SET DESCRIPTOR statement is not between 1 and the maximum number of items assigned when the descriptor was allocated.
Recovery Text:	For the ALLOCATE DESCRIPTOR statement, ensure the maximum items specified is between 1 and 8000. For GET DESCRIPTOR and SET DESCRIPTOR statements, ensure the item number specified in VALUE is between 1 and the maximum number of items for the specified descriptor. Try the request again.
SQLCODE or SQLCODEs:	-075

SQL0075	
SQLSTATE or SQLSTATEs:	07009

SQL0076	
Message Text:	DATA item not correct.
Cause Text:	The DATA item is not correct for one of the following reasons: • The GET DESCRIPTOR statement requested the DATA item, but the DATA item has no value. • The GET DESCRIPTOR or SET DESCRIPTOR statements specified a host variable for the DATA item that is not valid. The host variable must match the DATA item in type, length, precision, scale, and CCSID.
Recovery Text:	If DATA does not have a known value, perform an operation such as FETCH or use SET DESCRIPTOR with DATA to establish a DATA value prior to the GET DESCRIPTOR. If the host variable on the GET DESCRIPTOR or SET DESCRIPTOR statement does not match the attributes of the DATA item, change the attributes of the host variable. Try the request again.
SQLCODE or SQLCODEs:	-076
SQLSTATE or SQLSTATEs:	22005

SQL0077	
Message Text:	Descriptor name &1 not correct.
Cause Text:	Descriptor name &1 is not correct for one of the following reasons: • The descriptor name is longer than 128 characters. • The descriptor name has not been allocated with an ALLOCATE DESCRIPTOR statement. • A descriptor with this name already exists for the connection and scope.
Recovery Text:	Make sure the name is not too long. Use the ALLOCATE DESCRIPTOR statement to define the descriptor before referring to it in another statement. If there is already a descriptor with the same name for the connection and scope, change the name in the ALLOCATE DESCRIPTOR statement. Try the request again.
SQLCODE or SQLCODEs:	-077
SQLSTATE or SQLSTATEs:	33000

SQL0078	
Message Text:	Parameter name required for routine &1 in &2.
Cause Text:	Parameter name must be specified when creating SQL routines.
Recovery Text:	Specify a parameter name. Try the request again.
SQLCODE or SQLCODEs:	-078
SQLSTATE or SQLSTATEs:	42629

SQL0079	
Message Text:	Schema &2 for object &1 not valid.
Cause Text:	Object &1 cannot be qualified with schema &2. A table in the DECLARE GLOBAL TEMPORARY TABLE statement must be qualified with SESSION. A reference to a temporary table must be qualified with SESSION or QTEMP. Indexes and views over a temporary table must be created in the schema SESSION or QTEMP.
Recovery Text:	Specify SESSION as the schema name. Try the request again.
SQLCODE or SQLCODEs:	-079
SQLSTATE or SQLSTATEs:	428EK

SQL0080	
Message Text:	Indicator variable &1 not SMALLINT type.
Cause Text:	The definition of indicator variable &1 must be a 2-byte binary with a zero scale.
Recovery Text:	Specify an indicator variable that is defined as a 2-byte binary with a zero scale. Try the request again.
SQLCODE or SQLCODEs:	-080
SQLSTATE or SQLSTATEs:	42978

SQL0084	
Message Text:	SQL statement not allowed.

SQL0084	
Cause Text:	The SQL statement is not allowed for one of the following reasons: • DECLARE CURSOR, DECLARE STATEMENT, FETCH, OPEN, CLOSE, WHENEVER, PREPARE, EXECUTE, EXECUTE IMMEDIATE, INCLUDE, DECLARE VARIABLE, DECLARE PROCEDURE, DESCRIBE, GET DIAGNOSTICS, SIGNAL, RESIGNAL, SET variable, ALLOCATE DESCRIPTOR, DEALLOCATE DESCRIPTOR, GET DESCRIPTOR, and SET DESCRIPTOR are not allowed in interactive SQL, dynamic SQL, or when using the RUNSQLSTM command. • VALUES INTO is not allowed in interactive SQL or when using the RUNSQLSTM command. • BEGIN DECLARE SECTION and END DECLARE SECTION are not allowed in interactive SQL, in dynamic SQL, in RPG, or in REXX. • A blocked INSERT statement is not allowed in interactive SQL or dynamic SQL. • The CONNECT, SET CONNECTION, RELEASE, and DISCONNECT statements are not allowed in dynamic SQL or REXX. CONNECT with constants specified for user ID and password is not allowed in a precompiled program. • SET ENCRYPTION PASSWORD with constants specified for user ID or hint is not allowed in a precompiled program. • SELECT cannot be issued from an EXECUTE IMMEDIATE statement or the RUNSQLSTM command. • The SET OPTION statement is only allowed in REXX, in a precompiled program, or in SQL routines. In a precompiled program, it must be the first SQL statement in the program. • DECLARE STATEMENT, DECLARE VARIABLE, DECLARE PROCEDURE, GET DIAGNOSTICS, INCLUDE, SELECT INTO, SET CURRENT DEGREE, SET RESULT SETS, SIGNAL, WHENEVER, blocked INSERT, blocked FETCH, locator statements, and SQL descriptor statements are not allowed in REXX. • The SET TRANSACTION statement is not allowed when the current connection is to a remote database. • The SET SESSION AUTHORIZATION statement can only be run in the default activation group. It is not allowed in REXX, or when using the RUNSQLSTM command. • The SQL statement specified is not a valid statement on the current release of DB2 UDB for iSeries. The statement may be valid on a future release of DB2 UDB for iSeries or on a system other than an iSeries.
Recovery Text:	The statement cannot be run in this mode. For a CONNECT or SET ENCRYPTION PASSWORD statement in a precompiled program, specify host variables instead of constants. If in interactive SQL, you may syntax check a statement by setting the statement processing value to *SYN.
SQLCODE or SQLCODEs:	+084, -084
SQLSTATE or SQLSTATEs:	01505, 42612

SQL0087	
Message Text:	Variable cannot be NULL.
Cause Text:	An SQL variable or parameter was specified in an SQL routine and is NULL. The variable is specified in an SQL statement where NULL is not allowed.
Recovery Text:	Assign a value to the SQL variable or parameter. Try the request again.
SQLCODE or SQLCODEs:	-087
SQLSTATE or SQLSTATEs:	22004

SQL0088	
Message Text:	&1 applies to entire table.
Cause Text:	The UPDATE or DELETE statement does not have a WHERE clause and will delete or update all the rows in the specified table.
Recovery Text:	Do one of the following: • Verify that all the rows in the specified table need to be deleted or updated and try the statement again. • If the rows in the specified table do not need to be deleted or updated, add a WHERE clause and precompile the program again.
SQLCODE or SQLCODEs:	+088
SQLSTATE or SQLSTATEs:	01504

SQL0090	
Message Text:	Host variable not permitted here.
Cause Text:	Host variable &1 is not allowed as used in this statement. Host variables are not allowed: • In a CREATE VIEW, CREATE TABLE, or ALTER TABLE statement. • In any interactive SQL statement when the Statement processing value is *RUN or *VLD. • In an SQL statement processed by the RUNSQLSTM command. • In an INSERT, UPDATE, DELETE, or DECLARE CURSOR statement in REXX.
Recovery Text:	Do one of the following and try the request again: • Specify either a constant or a column name to replace the host variable. The colon indicates that the name that follows is a host variable. Remove the colon to specify a column name. • If in interactive SQL, set the statement processing value to *SYN to syntax check a statement that contains a host variable. • If in REXX, change the host variables to parameter markers and prepare the INSERT, UPDATE, DELETE, or DECLARE CURSOR statement.
SQLCODE or SQLCODEs:	-090
SQLSTATE or SQLSTATEs:	42618

SQL0097	
Message Text:	Use of data type not valid.
Cause Text:	The data type specified in the statement can not be specified for a procedure or function. Data types such as LONG VARCHAR can only be specified for columns, and cannot be specified for parameters.
Recovery Text:	Correct the data type specified for the procedure or function. Try the request again.
SQLCODE or SQLCODEs:	-097
SQLSTATE or SQLSTATEs:	42601

SQL0100	
Message Text:	Row not found for &1.
Cause Text:	One of the following conditions has occurred: • If this is a FETCH statement, no more rows satisfy the selection values (end of file). The name of the cursor is &1 and the result set identifier is &2. If the result set identifier is non-zero, the result table for this cursor was being accessed as a stored procedure result set. • If this is a FETCH statement for a scrollable cursor, a record was not found. If NEXT was specified, end of file was reached. If PRIOR was specified, the beginning of the file was reached. If RELATIVE was specified, either the beginning of file or the end of file was reached, depending on the value specified. If FIRST or LAST was specified, then no records satisfy the selection criteria. The name of the cursor is &1. • If this is an embedded SELECT statement, no rows satisfy the selection values. • If this is an UPDATE, INSERT, or DELETE statement, no rows satisfy the subselect or WHERE clause. No rows were updated, inserted, or deleted.
Recovery Text:	No recovery is necessary.
SQLCODE or SQLCODEs:	+100
SQLSTATE or SQLSTATEs:	02000

SQL0101	
Message Text:	SQL statement too long or complex.
Cause Text:	The SQL statement is longer than the limit allowed for length or complexity. The reason code is &1. One of the following reason codes indicates the error: • 1 - The total number of subselects combined using UNION, EXCEPT, and INTERSECT is greater than 256. • 2 - The total number of columns, constants, and operators is greater than the SQL limits. • 3 - The sum of the lengths of the non-LOB columns in a select list, table, view definition, or user defined table function is greater than 32766 or the definition contains a LOB and the sum of the lengths specified on the ALLOCATE clause for varying-length fields and the non-varying field lengths is greater than 32740. The maximum length is reduced if any of the columns are varying-length or allow null values. • 4 - The total number of subselects referenced in an outer subselect is greater than 256. • 5 - The total length of the statement text is greater than 2097152, or the total length of the prepare attribute string text is greater than 65535. • 6 - The relative position value specified on the FETCH statement is outside the range of valid values. • 7 - A system name could not be generated.
Recovery Text:	Simplify the statement or divide the statement up into more than one statement and try the request again. For reason code 7, specify a different name for the table, view, index, or alias.
SQLCODE or SQLCODEs:	-101
SQLSTATE or SQLSTATEs:	54001, 54010, 54011

SQL0102	
Message Text:	String constant beginning with &1 too long.
Cause Text:	The string constant beginning with &1 is larger than 32740 bytes. If this is a graphic string constant, the string cannot be longer than 16370 DBCS characters.

SQL0102	
Recovery Text:	Reduce the length of the string. Try the request again.
SQLCODE or SQLCODEs:	-102
SQLSTATE or SQLSTATEs:	54002

SQL0103	
Message Text:	Numeric constant &1 not valid.
Cause Text:	The token &1 begins with a digit, but the token is not a valid integer, decimal, or floating point constant. Identifiers cannot begin with a digit except in a COBOL program or for the WHENEVER statement in a FORTRAN program.
Recovery Text:	Do one of the following and precompile the program again: • Verify that token &1 is valid. Use apostrophes or quotation marks if a character constant is required. • Remove the character or characters that are not valid if a number is required.
SQLCODE or SQLCODEs:	-103
SQLSTATE or SQLSTATEs:	42604

SQL0104	
Message Text:	Token &1 was not valid. Valid tokens: &2.
Cause Text:	A syntax error was detected at token &1. Token &1 is not a valid token. A partial list of valid tokens is &2. This list assumes that the statement is correct up to the token. The error may be earlier in the statement, but the syntax of the statement appears to be valid up to this point.
Recovery Text:	Do one or more of the following and try the request again: • Verify the SQL statement in the area of the token &1. Correct the statement. The error could be a missing comma or quotation mark, it could be a misspelled word, or it could be related to the order of clauses. • If the error token is <end-of-statement> correct the SQL statement because it does not end with a valid clause.
SQLCODE or SQLCODEs:	-104
SQLSTATE or SQLSTATEs:	42601

SQL0105	
Message Text:	Mixed, graphic, or UTF-8 string constant not valid.

SQL0105	
Cause Text:	Mixed, graphic, UTF-8 constants that are not valid were found in the value beginning &1. One of the following occurred: • An odd number of bytes were found between the shift-out and shift-in characters. • Multiple shift-out characters were found before a shift-in character was found. • A shift-out and shift-in were not found in the first and last byte, respectively, or were found in a position other than the first and last byte of a graphic string constant. • The PL/I form of the graphic string was used but the program is not PL/I. • A shift-out was found indicating a PL/I graphic string. The shift-out was not followed by a DBCS apostrophe, an even number of DBCS characters, another DBCS apostrophe, a DBCS G, and a shift-in. If this is a LABEL ON statement for a column, and the string is longer than 20 bytes, then one of the 20-byte segments has a DBCS constant that is not valid.
Recovery Text:	Specify the correct format for the constant. If this is a LABEL ON statement for a column, ensure each 20-byte segment is in the correct format. Check for a quotation mark, an apostrophe, shift-out or shift-in character, or an odd number of characters between the shift-out and shift-in characters. Ensure graphic string constants are specified in the correct form for the language. Try the request again.
SQLCODE or SQLCODEs:	-105
SQLSTATE or SQLSTATEs:	42604

SQL0106	
Message Text:	Precision specified for FLOAT not valid.
Cause Text:	The precision specified for the FLOAT column or parameter is not valid for floating point data. The precision allowed is from 1-53. If 1-23 is specified, the column or parameter is defined as single-precision floating point. If 24-53 is specified, the column or parameter is defined as double-precision floating point.
Recovery Text:	Change the precision specified. Try the request again.
SQLCODE or SQLCODEs:	-106
SQLSTATE or SQLSTATEs:	42611

SQL0107	
Message Text:	&1 too long. Maximum &2 characters.

SQL0107	
Cause Text:	The name or string beginning with &1 is too long. The maximum length allowed is &2. The maximum length for names depends on the type of the name: • System names for a schema, package, program, and other system objects cannot exceed 10 characters. • SQL names for a table, view, index, alias, constraint, correlation, parameter, user-defined type, trigger, sequence, or savepoint cannot exceed 128 characters. • SQL names for a column cannot exceed 128 characters. System-column names cannot exceed 10 characters. • Cursor names, statement names, or relational database names cannot exceed 18 characters. • Procedure or function names cannot exceed 128 characters. If the external program name is not specified, the name cannot exceed 10 characters because it is used for the program name. • Host variable names in C and C++ cannot exceed 128 characters. Host variable names in all other languages cannot exceed 64 characters. • Names for SQL variables, parameters, labels, or conditions in the routine body of an SQL routine cannot exceed 128 characters. The maximum length of the string for a password is 128. The maximum length of the string for a COMMENT ON statement is 2000. The string for COMMENT ON SEQUENCE cannot exceed 500 characters. The maximum length of the string for a LABEL ON statement is 50. If the label is specified as a column heading, the maximum is 60. The maximum length of the string for a package version is 64.
Recovery Text:	Change the name or string to a length of &2 or less. Try the request again.
SQLCODE or SQLCODEs:	-107
SQLSTATE or SQLSTATEs:	42622

SQL0109	
Message Text:	&1 clause not allowed.
Cause Text:	One of the following conditions was not allowed: • Embedded SELECT statements cannot include the FOR UPDATE clause, the FOR READ ONLY clause, the FOR FETCH ONLY clause, the OPTIMIZE clause, or the UNION, EXCEPT or INTERSECT operator. • SELECT statement used in cursor declarations or subselects cannot have an INTO clause. • CREATE VIEW statements may not have an INTO, ORDER BY, FOR UPDATE, FOR READ ONLY, FOR FETCH ONLY, or OPTIMIZE clause. • INSERT statements may not have an INTO clause in a subselect, a FOR UPDATE, FOR READ ONLY, FOR FETCH ONLY, or an OPTIMIZE clause. • WHERE CURRENT OF cursor clause is not allowed in statements processed in interactive SQL or statements processed by the RUNSQLSTM command. • The NOT NULL clause is not allowed in the definition of a column being added to a table unless a default value is specified. • OVERRIDING USER VALUE and OVERRIDING SYSTEM VALUE are only valid if the statement is changing a column that is defined with GENERATED ALWAYS. • The RAISE_ERROR function cannot be specified alone in the select list or in a CASE expression.
Recovery Text:	Remove the clause. Try the request again.
SQLCODE or SQLCODEs:	-109

SQL0109	
SQLSTATE or SQLSTATEs:	42601

SQL0110	
Message Text:	Hexadecimal constant beginning with &1 not valid.
Cause Text:	Either the hexadecimal constant &1 contains one or more characters that are not valid or the number of characters between the string delimiters is not valid. All embedded blank (X'40') characters are removed from the string before verification is done. Hexadecimal constants must contain an even number of characters. The length of a hexadecimal graphic string must be a multiple of 4 to ensure that it contains a valid number of DBCS characters. Hexadecimal constants representing MIXED data must have an even number of bytes between the shift-out (X'0E') and shift-in (X'0F') characters and must have paired shift characters. Hexadecimal graphic constants cannot contain a shift-out or a shift-in.
Recovery Text:	Correct the constant. Ensure that the constant contains an even number of digits and that the length for a hexadecimal graphic constant is a multiple of 4. The valid characters for hexadecimal digits are characters 0 through 9 and uppercase or lowercase A through F. Ensure MIXED data is in the correct format. Remove shift-out or shift-in characters from a hexadecimal graphic constant. Try the request again.
SQLCODE or SQLCODEs:	-110
SQLSTATE or SQLSTATEs:	42606

SQL0112	
Message Text:	Argument of function &1 contains another function.
Cause Text:	The argument of column function &1 contains another column function. Only expressions without column functions are allowed as arguments of a column function.
Recovery Text:	Correct the function specification. Try the request again.
SQLCODE or SQLCODEs:	-112
SQLSTATE or SQLSTATEs:	42607

SQL0113	
Message Text:	Name &1 not allowed.

SQL0113	
Cause Text:	&1 contains a character that is not allowed or does not contain any characters. An ordinary identifier must begin with (A-Z, $, #, or @) and be followed by zero or more (A-Z, 0-9, $, #, @, or _). A delimited identifier is a string of characters within SQL escape characters. The characters allowed in delimited names depends on the type of name: • The characters between the escape characters for system table names, schema names, package names and other system object names can be any characters except Hex 00-3F, Hex 40 (space), Hex 5C (*), Hex 6F (?), Hex 7D (’), Hex 7F (″), and Hex FF. • The characters between the escape characters for SQL table names, cursor names, statement names, correlation names, column names, procedure names, function names, parameter names, constraint names, user-defined type names, trigger names, or sequence names can be any characters except Hex 00-3F and Hex FF. • Delimited system column names must begin with (A-Z, $, #, or @) and be followed by zero or more (A-Z, 0-9, $, #, @, or an _). A relational database name must begin with A-Z and be followed by 0 or more letters, numbers 0-9, or an _. Ordinary identifiers that are contained within host variables must not include lowercase letters because they are not converted to uppercase.
Recovery Text:	Correct the name. Try the request again.
SQLCODE or SQLCODEs:	-113
SQLSTATE or SQLSTATEs:	28000, 2E000, 42602

SQL0114	
Message Text:	Relational database &1 not the same as current server &2.
Cause Text:	Relational database &1 was specified in a 3 part name in the statement. However, either the name is not the same as the current server &2, or the name is not the same as a relational database name specified previously in the statement.
Recovery Text:	Change the statement so the relational database name specified is the same as the current server or so that all relational database names in the statement match.
SQLCODE or SQLCODEs:	+114, -114
SQLSTATE or SQLSTATEs:	01536, 42961

SQL0115	
Message Text:	Comparison operator &1 not valid.
Cause Text:	Simple comparison operators other than equal and not equal cannot be used with a list of items. ANY, ALL, and SOME comparison operators must be followed by a fullselect, rather than an expression or a list of items, and cannot be specified in a JOIN condition or in a CASE expression.
Recovery Text:	Change either the comparison or the operand. Try the request again.
SQLCODE or SQLCODEs:	-115
SQLSTATE or SQLSTATEs:	42601

SQL0117	
Message Text:	Statement contains wrong number of values.

SQL0117	
Cause Text:	The following conditions may exist: • The number of values is not the same as the number of object columns in this INSERT or UPDATE statement. • The number of values is not the same as the number of target host variables in this SET or VALUES INTO statement. • The number of SELECT list items in the subselect is not the same as the number of object columns in this INSERT statement • The number of SELECT list items in the subselect in a SET clause is not the same as the number of object columns for the SET clause in this UPDATE statement. • One or more of the object columns not specified in the INSERT statement were created as NOT NULL. • One or more of the object columns specified in the INSERT statement were created as NOT NULL, and the statement specified DEFAULT as the value to be inserted.
Recovery Text:	Correct the statement to specify a single value for each of the object columns. Ensure that the character designated as the decimal point is used correctly in any numeric literals. If the object columns not specified in the INSERT statement were created as NOT NULL, specify valid values for those columns. Try the request again.
SQLCODE or SQLCODEs:	-117
SQLSTATE or SQLSTATEs:	42802

SQL0119	
Message Text:	Column &1 or expression in HAVING clause not valid.
Cause Text:	One of the following has occurred: • Column &1 specified in a HAVING clause is not within a column function and is not in the GROUP BY clause. • An expression specified in a HAVING clause is not within a column function and is not in the GROUP BY clause.
Recovery Text:	Remove the column or expression from the HAVING clause or add the column or expression to the GROUP BY clause. Try the request again.
SQLCODE or SQLCODEs:	-119
SQLSTATE or SQLSTATEs:	42803

SQL0120	
Message Text:	Use of function &2 not valid.
Cause Text:	Function &2 cannot be used where it was specified: • A column function or user-defined function sourced on a column function cannot be specified in a SET clause, in a GROUP BY clause, in a JOIN condition, as the return expression for a RETURN statement, or as an expression in a SET statement or VALUES INTO statement. A column function or user-defined function sourced on a column function is allowed in a WHERE clause only if the WHERE clause appears within a subquery of a HAVING clause. • An OLAP function cannot be specified in a WHERE clause, in a GROUP BY clause, in a HAVING clause, or in a JOIN condition. An OLAP function cannot be specified as an expression in a SET statement or VALUES INTO statement, in the SET clause of an UPDATE statement, or as the return expression for a RETURN statement.

SQL0120	
Recovery Text:	Remove the function. Try the request again.
SQLCODE or SQLCODEs:	-120
SQLSTATE or SQLSTATEs:	42903

SQL0121	
Message Text:	Duplicate name &1 not allowed.
Cause Text:	Name &1 is specified more than once in either the list of object columns of an INSERT statement, in the SET clause of an UPDATE statement, or in the list of target host variables in the SET or VALUES INTO statement. If the specified names are not the same then one of the following has occurred: • If the object is a view, the column they identify in the base table may be the same column. • The names may correspond to the same system column name.
Recovery Text:	Do one of the following and try the request again: • Remove the duplicate column. • Specify a column list on the INSERT statement to remove the duplicate column.
SQLCODE or SQLCODEs:	-121
SQLSTATE or SQLSTATEs:	42701

SQL0122	
Message Text:	Column &1 or expression in SELECT list not valid.
Cause Text:	One of the following has occurred: • The statement contains column name &1 and a column function in the SELECT clause and no GROUP BY clause is specified. • Column name &1 is specified in the SELECT clause but not in the GROUP BY clause. • An expression is specified in the SELECT clause but not in the GROUP BY clause. • A column or expression that is specified in the ORDER BY clause, but not in the SELECT clause, does not conform to the grouping rules listed above.
Recovery Text:	Do one of the following and try the request again: • If a GROUP BY clause is required, make certain that all columns or expressions in the SELECT list and ORDER BY clause are also in the GROUP BY clause. • If a GROUP BY clause is not needed, the SELECT list and ORDER BY clause should not contain column functions with column names.
SQLCODE or SQLCODEs:	-122
SQLSTATE or SQLSTATEs:	42803

SQL0125	
Message Text:	ORDER BY column number &1 not valid.
Cause Text:	The ORDER BY clause in the statement contains a column number that is either greater than the maximum number of values that can be selected (8000), or is greater than the number of columns in the result table select list.

SQL0125	
Recovery Text:	Correct the column number in the ORDER BY clause to specify a column in the result table. Try the request again.
SQLCODE or SQLCODEs:	-125
SQLSTATE or SQLSTATEs:	42805

SQL0128	
Message Text:	Use of NULL is not valid.
Cause Text:	The keyword NULL is not valid with the operator specified. NULL is only allowed in a predicate following IS or IS NOT. NULL is a reserved keyword and can only be used as the name of a column if the name is delimited when used in an SQL statement.
Recovery Text:	Either change the operator to IS or IS NOT or, if the word NULL was meant to be a column name, specify the name within delimiters.
SQLCODE or SQLCODEs:	-128
SQLSTATE or SQLSTATEs:	42601

SQL0129	
Message Text:	Too many tables in SQL statement.
Cause Text:	The SQL statement contains too many tables or views. A single SQL statement can refer to a maximum of 256 tables or views. This number includes the base tables of a view.
Recovery Text:	Do one of the following and try the request again: • Split the SQL statement into two or more simpler statements with a maximum of 256 tables in each. • If this is a CREATE VIEW statement, reduce the number of tables to a maximum of 256.
SQLCODE or SQLCODEs:	-129
SQLSTATE or SQLSTATEs:	54004

SQL0130	
Message Text:	ESCAPE character &1 or LIKE pattern is not valid.
Cause Text:	Either ESCAPE character &1 is not valid or the use of the ESCAPE character in the LIKE pattern is not valid. The ESCAPE character is not valid if: • The length is not 1 SBCS character or 1 graphic character. • The shift-in (X'0E') and the shift-out (X'0F') characters are specified. The LIKE pattern is not valid if: • The character string expression forming the pattern contains an ESCAPE character that is not followed by a percent sign, an underscore, or another ESCAPE character. • The graphic string expression forming the pattern contains an ESCAPE character that is not followed by a DBCS percent sign, a DBCS underscore, or another ESCAPE character.
Recovery Text:	Specify a valid LIKE pattern and ESCAPE character. Try the request again.
SQLCODE or SQLCODEs:	-130

SQL0130	
SQLSTATE or SQLSTATEs:	22019, 22025

SQL0131	
Message Text:	Operands of LIKE not compatible or not valid.
Cause Text:	The arguments of the LIKE predicate must be character, binary, graphic, or numeric. One of the following errors has occurred: • The operand to the right of the LIKE operator is not character, binary, graphic, or numeric. • The operands of the LIKE predicate are not compatible. • The ESCAPE character is not character, binary, or graphic.
Recovery Text:	Ensure the operands for the LIKE predicate are character, binary, graphic, or numeric. The ESCAPE character must be character, binary, or graphic. Try the request again.
SQLCODE or SQLCODEs:	-131
SQLSTATE or SQLSTATEs:	42818

SQL0132	
Message Text:	LIKE predicate not valid.
Cause Text:	Either the second operand or the ESCAPE character specified in a LIKE predicate is not valid. The second operand must be a string or numeric expression. The ESCAPE character must be a string expression but cannot be a special register.
Recovery Text:	Change the incorrect operand or the operator. Try the request again.
SQLCODE or SQLCODEs:	-132
SQLSTATE or SQLSTATEs:	42824

SQL0133	
Message Text:	Operator on correlated column in SQL function not valid.
Cause Text:	An SQL column function appearing in a subquery of a HAVING clause is not valid if the argument of the function is an expression that contains an operator (+, -, *, /, **), a concatenation operator, or a scalar function that is applied to a correlated reference. An operation cannot be performed on a correlated reference since the computed value of the group cannot be determined in the outer (correlated) subselect without a possible value from the inner subselect.
Recovery Text:	If the operator is a scalar function, make the column function the argument of the scalar function. Otherwise, remove the operator on the correlated reference or move the operator so it is not in the argument of the column function. For example, specifying the expression of the form: AVG(outertable.column1 + innertable.column2) is not valid, while the expression AVG(outertable.column1) + innertable.column2 is valid.
SQLCODE or SQLCODEs:	-133
SQLSTATE or SQLSTATEs:	42906

SQL0134	
Message Text:	String, argument, or path too long.
Cause Text:	One of the following errors has occurred: • The argument of a COUNT function is too long. The argument of a COUNT function cannot be longer than 2000 bytes if DISTINCT is specified. If the argument is graphic, then the argument cannot be longer than 1000 DBCS characters. • More than 268 libraries were specified on the SET PATH statement or on the SET OPTION SQLPATH statement. • A LOB column was used in a ORDER BY expression, GROUP BY expression, join specification, SELECT clause with DISTINCT, or in a UNION, EXCEPT, or INTERSECT in which the ALL keyword was omitted.
Recovery Text:	Change the argument of the function or the number of libraries in the path so that the length does not exceed the maximum. Remove the LOB column from the clause where it is not allowed. Try the request again.
SQLCODE or SQLCODEs:	-134
SQLSTATE or SQLSTATEs:	42907

SQL0136	
Message Text:	ORDER BY, GROUP BY, or join columns too long.
Cause Text:	The maximum number of elements in an ORDER BY list is 10000. The total length of all the ORDER BY elements cannot exceed 32766 bytes. The maximum number of columns in a GROUP BY list is 120. The total length of all the GROUP BY columns cannot exceed 32766 bytes. The total length of all the join columns in an exception join or outer join cannot exceed 32766 bytes. If the ORDER BY or GROUP BY list contains null capable columns, then an additional byte is required for each null capable column. If the ORDER BY or GROUP BY list contains varying-length columns, then the 2 byte length is included in the total length. If the allow copy data option is ALWCPYDTA(*NO), the total length of all the GROUP BY columns cannot exceed 2000 bytes.
Recovery Text:	The statement must be changed so that the length of the ORDER BY, GROUP BY, or join values does not exceed their limits. One or more column names must be removed from the clause. Try the request again.
SQLCODE or SQLCODEs:	-136
SQLSTATE or SQLSTATEs:	54005

SQL0137	
Message Text:	Result too long.
Cause Text:	A concatenation operator or a HEX scalar function was specified, but the resulting length of the operation exceeds the maximum allowed. The maximum length is: • 32766 bytes if the result is fixed-length character or fixed-length binary. • 32740 bytes if the result is varying-length character or varying-length binary. • 16383 DBCS characters if the result is fixed-length graphic. • 16370 DBCS characters if the result is varying-length graphic. • 2147483647 bytes if the result is a binary or character LOB. • 1073741823 DBCS characters if the result is a double-byte character LOB.

SQL0137	
Recovery Text:	Change the expression to decrease the resulting length to less than or equal to the maximum allowed. If converting from graphic to character data, the result length specified on the scalar function must be less than 8191. The SUBSTR scalar function can be used to decrease the length of an operand. Try the request again.
SQLCODE or SQLCODEs:	-137
SQLSTATE or SQLSTATEs:	54006

SQL0138	
Message Text:	Argument &1 of substringing function not valid.
Cause Text:	Argument 2 or 3 of the SUBSTRING function or argument 2 of the LEFT function is either out of range or is an expression that does not evaluate to an integer. • For the SUBSTRING function, argument 2 specifies the position of the first character of the result and argument 3 specifies the length of the result. Argument 2 must be a valid position of the first argument. Argument 3 must not exceed the length of argument 1 between argument 2 and the end of the string. • For the LEFT function, argument 2 specifies the length of the result. Argument 2 must not exceed the length of argument 1. • If argument 1 is a character string or a binary string, a character is a byte, and if argument 1 is graphic string, a character is a DBCS character. • If the argument is *N, then one of the arguments is not valid but the argument number is not known.
Recovery Text:	If the argument is *N, display the previously listed messages in the job log (DSPJOBLOG command) or press F10 (Display messages in job log) from this display to determine which argument is in error. Change one or more of the arguments specified in the SUBSTR function. The INTEGER scalar function may be used to convert the argument into an integer result. Try the request again.
SQLCODE or SQLCODEs:	+138, -138
SQLSTATE or SQLSTATEs:	01544, 22011

SQL0142	
Message Text:	SET SESSION AUTHORIZATION statement is not allowed.
Cause Text:	Use of the SET SESSION AUTHORIZATION statement is restricted through this interface.
Recovery Text:	Use the SET SESSION AUTHORIZATION statement through a different interface.
SQLCODE or SQLCODEs:	-142
SQLSTATE or SQLSTATEs:	42612

SQL0144	
Message Text:	Section number &1 not valid. Current high section number is &3. Reason &2.
Cause Text:	Reason code is &2. Reason code 1, section number &1 has already been assigned. Reason code 2, section number &1 is smaller than next possible number. Reason code 3, section number on ENDBND is smaller than highest one assigned. Reason code 4, section number is not in the SQL package. Reason code 5, section number of zero is not valid.

SQL0144	
Recovery Text:	Contact your IBM representative to report the problem.
SQLCODE or SQLCODEs:	-144
SQLSTATE or SQLSTATEs:	58003

SQL0145	
Message Text:	Recursion not supported for application server other than iSeries.
Cause Text:	Program &1 in &2 was called recursively when connected to an application server that is not an iSeries. The program was connected to application server &3 with product identification of &4. If the application server is an IBM product, the identification is in the form pppvvrrm, where: ppp identifies the product as follows: DSN for DB2 UDB for OS/390® and z/OS® ARI for VM and VSE QSQ for DB2 UDB for iSeries SQL for all other DB2 products vv is a two-digit version identifier such as ' 06 ' rr is a two-digit release identifier such as ' 01 ' m is a one-digit modification level such as ' 0 ' . For example, if the application server is Version 6 Release 1 of DB2 UDB for OS/390 and z/OS, the value of the product identification is ' DSN06010 ' .
Recovery Text:	Change your application so that it is not recursively called when connected to a server other than an iSeries.
SQLCODE or SQLCODEs:	-145
SQLSTATE or SQLSTATEs:	55005

SQL0150	
Message Text:	View or logical file &1 in &2 read-only.
Cause Text:	Update, delete, or insert is not allowed. &1 in &2 can be used only for read operations. A view or logical file can be used only for read operations if one or more of the following conditions are true: • The view contains a DISTINCT keyword, GROUP BY clause, HAVING clause, or a column function in the outer-most subselect. • The view or logical file contains a join function. • The view contains a subquery that refers to the same table as the table of the outer-most subselect. A view of this type may be used for inserting rows. • All the columns of the view are expressions, scalar functions, constants, or special registers. • All the columns of the logical file are input only. • The select list of the view omits a column of the based on table that does not allow null values or default values. Inserting into the view is not allowed.
Recovery Text:	Change the statement to insert, delete, or update data into the base table of view &1. All columns of the table that do not allow null values or default values must be assigned a value when inserting a row into a table or view. Try the request again.
SQLCODE or SQLCODEs:	-150
SQLSTATE or SQLSTATEs:	42807

SQL0151	
Message Text:	Column &1 in table &2 in &3 read-only.

SQL0151	
Cause Text:	&1 is a column of an implicit or explicit column list for an INSERT statement or a SET clause on an UPDATE statement. &1 is read only because it is: • Derived from an expression, a constant, or a special register. • Defined on a column of an underlying view that cannot be updated. • In a column of a logical file that is defined as input only.
Recovery Text:	Remove column &1 from the column list or the SET clause. If this is an INSERT and a column list was not specified, then specify a column list to remove column &1. Try the request again.
SQLCODE or SQLCODEs:	-151
SQLSTATE or SQLSTATEs:	42808

SQL0152	
Message Text:	Constraint type not valid for constraint &1 in &2.
Cause Text:	An attempt was made to drop constraint &1 in &2 using an ALTER TABLE statement. The constraint was specified as CHECK, UNIQUE, PRIMARY, or FOREIGN KEY and is not the same as the constraint found.
Recovery Text:	Verify the name and type of the constraint you want to drop. Try the request again.
SQLCODE or SQLCODEs:	-152
SQLSTATE or SQLSTATEs:	42809

SQL0153	
Message Text:	Column list required.
Cause Text:	A column list must be specified because the result columns are unnamed. Result columns are unnamed for one of the following reasons: • An element of the SELECT list is not a column and the AS clause is not specified. • Corresponding columns of the subselects in a UNION, EXCEPT, or INTERSECT do not have the same name. • Two result columns have the same column name. Every column name and system-column name must be unique in a table or view definition. If two column names are the same, the column name is &1.
Recovery Text:	Do one of the following and try the request again: • Provide a list of names for the columns in the table, view, or derived table. • Specify an AS clause to assign a unique name to the unnamed elements or to rename the duplicate columns in the SELECT list.
SQLCODE or SQLCODEs:	-153
SQLSTATE or SQLSTATEs:	42908

SQL0155	
Message Text:	Transition table &1 read-only.
Cause Text:	Statement is not allowed. Transition table &1 in an SQL trigger can be used only for read operations.

SQL0155	
Recovery Text:	Change the statement to specify a table other than the transition table or remove the statement. Transition tables can be specified on SELECT statements. Qualify table names in an SQL trigger that may have the same name as transition tables. Try the request again.
SQLCODE or SQLCODEs:	-155
SQLSTATE or SQLSTATEs:	42807

SQL0156	
Message Text:	&1 in &2 not correct type.
Cause Text:	A DROP TABLE, CREATE INDEX, LOCK TABLE, ALTER TABLE, CREATE TRIGGER or ALTER PROCEDURE statement was specified but &1 in &2 is the wrong type. DROP TABLE, CREATE INDEX, LOCK TABLE, and ALTER TABLE must specify a table. CREATE TRIGGER for a BEFORE or AFTER trigger must specify a table. CREATE TRIGGER for an INSTEAD OF trigger must specify a view. ALTER PROCEDURE must specify a procedure that was created as LANGUAGE SQL.
Recovery Text:	Change the statement to use the correct object type. Try the request again.
SQLCODE or SQLCODEs:	-156
SQLSTATE or SQLSTATEs:	42809

SQL0157	
Message Text:	&1 in &2 not valid in FOREIGN KEY clause.
Cause Text:	View or logical file &1 in &2 was specified in the REFERENCES clause in the definition of a FOREIGN KEY referential constraint on a CREATE TABLE or an ALTER TABLE statement. Views and logical files cannot be specified in a FOREIGN KEY clause.
Recovery Text:	Specify the base table that contains the parent key in the FOREIGN KEY clause. Try the request again.
SQLCODE or SQLCODEs:	-157
SQLSTATE or SQLSTATEs:	42810

SQL0158	
Message Text:	Number of columns specified not consistent.
Cause Text:	One of the following has occurred: • The number of column names specified for a view in a CREATE VIEW statement is not the same as the number of elements specified in the following SELECT clause. • The number of column names specified for a table in a CREATE TABLE statement is not the same as the number of elements specified in the following SELECT clause. • The number of column names specified in a correlation clause is not the same as the number of elements in the table, derived table, or table function. • The number of column names specified in a column list of a common table expression is not the same as the number of elements specified in the subselect. • The number of columns specified in the RETURNS TABLE clause of a user defined table function is not the same as the number of elements specified in the subselect in the RETURN statement.

SQL0158	
Recovery Text:	Specify a column name for each column in the view, table, or table function. Try the request again.
SQLCODE or SQLCODEs:	-158
SQLSTATE or SQLSTATEs:	42811

SQL0159	
Message Text:	&1 in &2 not correct type.
Cause Text:	A DROP INDEX, DROP VIEW, DROP ALIAS, COMMENT ON INDEX, COMMENT ON ALIAS, RENAME INDEX or REFRESH TABLE statement was specified but &1 in &2 is not the correct type. DROP INDEX can only be used to drop an index. DROP VIEW can only be used to drop a view. DROP ALIAS can only be used to drop an alias. COMMENT ON INDEX can only be used to comment on an index. COMMENT ON ALIAS can only be used to comment on an alias. RENAME INDEX can only be used to rename an index. REFRESH TABLE can only be used to refresh a materialized query table.
Recovery Text:	Do one of the following to correct the problem and try the request again: • If this is a DROP statement: – If &1 is an index, use a DROP INDEX statement. – If &1 is a view, use a DROP VIEW statement. – If &1 is a table, use a DROP TABLE statement. – If &1 is an alias, use a DROP ALIAS statement. • If this is a COMMENT ON INDEX or COMMENT ON ALIAS statement and &1 is a table or view, use a COMMENT ON TABLE statement. • If this is a RENAME INDEX statement and &1 is a table or view, use a RENAME TABLE statement. • If this is a REFRESH TABLE statement, specify a materialized query table.
SQLCODE or SQLCODEs:	-159
SQLSTATE or SQLSTATEs:	42809

SQL0160	
Message Text:	WITH CHECK OPTION not allowed for view &1 in &2.

SQL0160	
Cause Text:	The WITH CHECK OPTION clause is not allowed in the CREATE VIEW statement for view &1 in &2 because the fullselect contains one of the following: • The first FROM clause identifies more than one table or view. • The first SELECT clause contains the DISTINCT keyword. • A UNION, EXCEPT, or INTERSECT was specified. • The outer subselect contains a GROUP BY clause. • The outer subselect contains a HAVING clause. • The outer subselect contains a column function. • The outer subselect contains a UNION, EXCEPT, or INTERSECT. • All the select items in the outer subselect are expressions. • The SELECT statement contains a scalar subselect. • The WITH CASCADED CHECK OPTION clause was specified and a view in the FROM clause contains a scalar subselect. • A function that is not deterministic is in the select list of a scalar subselect. • A view in the FROM clause has an INSTEAD OF trigger defined on it.
Recovery Text:	Remove the WITH CHECK OPTION clause.
SQLCODE or SQLCODEs:	-160
SQLSTATE or SQLSTATEs:	42813

SQL0161	
Message Text:	INSERT or UPDATE not allowed because a resulting row does not satisfy view definition &1 in &2.
Cause Text:	The INSERT or UPDATE could not be done because a resulting row did not satisfy the view definition for &1 in &2. Either the view or an underlying view contains a WITH CHECK OPTION clause.
Recovery Text:	Change the data being inserted or updated so that it conforms to the view definition.
SQLCODE or SQLCODEs:	-161
SQLSTATE or SQLSTATEs:	44000

SQL0170	
Message Text:	Number of arguments for function &1 not valid.
Cause Text:	

SQL0170	
Recovery Text:	A built-in function has been specified with an incorrect number of arguments. If this function is meant to be a user defined function reference, a function with this signature was not found using the current path. • The CURDATE, CURTIME, DATABASE, GENERATE_UNIQUE, IDENTITY_VAL_LOCAL, NOW, and PI functions must be specified without any arguments. • The ADD_MONTHS, CONCAT, IFNULL, LEFT, MOD, MULTIPLY_ALT, NEXT_DAY, NULLIF, POSSTR, POWER, RAISE_ERROR, REPEAT, RIGHT, ROUND, TRUNCATE, and VARCHAR_FORMAT functions must have two arguments. • The REPLACE function must have three arguments. • The INSERT function must have four arguments. • The RAND function may have zero or one argument. • The BINARY, BLOB, CHAR, and TIMESTAMP functions may have one or two arguments. • The CLOB, DBCLOB, GRAPHIC, STRIP, VARCHAR, VARGRAPHIC, and the encryption and decryption functions may have between one and three arguments. • The DECIMAL, TRANSLATE, and ZONED functions may have between one and four arguments. • The COALESCE, MAX, and MIN functions must have at least two arguments. • The LOCATE and SUBSTRING functions may have two or three arguments.
SQLCODE or SQLCODEs:	Correct the number of arguments specified for the function. If this is a user defined function, correct the path or the function signature. Try the request again.
SQLSTATE or SQLSTATEs:	42605

SQL0171	
Message Text:	Argument &1 of function &2 not valid.
Cause Text:	The data type, length, or value of argument &1 of function &2 specified is not valid.
Recovery Text:	Refer to the DB2 UDB for iSeries SQL Reference topic in the Information Center for more information on scalar functions. Correct the arguments specified for the function. Try the request again.
SQLCODE or SQLCODEs:	-171
SQLSTATE or SQLSTATEs:	42815

SQL0175	
Message Text:	COMMIT, ROLLBACK, or SAVEPOINT failed.
Cause Text:	A commit, rollback, savepoint failed due to reason code &2. The logical unit of work identifier is &1. Reason codes and their meanings are: • 1 - A transaction program error occurred. • 2 - A commit resulted in a rollback. • 3 - The requested transaction operation failed. • 4 - The savepoint operation failed.
Recovery Text:	Display the previous messages in the joblog and take the appropriate action.
SQLCODE or SQLCODEs:	-175

SQL0175	
SQLSTATE or SQLSTATEs:	58028

SQL0177	
Message Text:	CHECK condition text too long.
Cause Text:	The CHECK condition text for a CHECK constraint is longer than 2000 bytes and does not fit in the SYSCHKCST catalog view. The CHECK condition text cannot be stored in the system catalog views. The CHECK_CLAUSE column of the SYSCHKCST catalog view will contain the null values for this constraint.
Recovery Text:	No recovery is necessary. If the complete text is required in the system catalog views, consider using multiple CHECK constraints instead of one large CHECK constraint.
SQLCODE or SQLCODEs:	+177
SQLSTATE or SQLSTATEs:	01009

SQL0178	
Message Text:	Query expression text for view &1 in &2 too long.
Cause Text:	The query expression text for view &1 in &2 is longer than 10000 bytes and does not fit in the SYSVIEWS catalog view. The statement text cannot be stored in the system catalog views. The VIEW_DEFINITION column of the SYSVIEWS catalog view will contain the null value for this view.
Recovery Text:	No recovery is necessary. If the complete text is required in the system catalog views, re-create the view with the length of the query expression less than or equal to 10000 bytes.
SQLCODE or SQLCODEs:	+178
SQLSTATE or SQLSTATEs:	0100A

SQL0180	
Message Text:	Syntax of date, time, or timestamp value not valid.
Cause Text:	The string representation of a date, time, or timestamp value does not conform to the syntax for the specified or implied data type and format. &2 is either the character string constant that is not valid or the column or host variable that contained the string. If the name is *N, then the value is an expression specified in the statement. If the string was found in a host variable, the host variable number is &1.
Recovery Text:	Ensure that the date, time, or timestamp value conforms to the syntax for the data type it represents. Try the request again.
SQLCODE or SQLCODEs:	+180, -180
SQLSTATE or SQLSTATEs:	01534, 22007

SQL0181	
Message Text:	Value in date, time, or timestamp string not valid.

SQL0181	
Cause Text:	The string representation of a date, time or timestamp value is not in the acceptable range. &2 is either the character string constant that is not valid or the column or host variable that contained the string. If the name is *N, then the value was found in an expression specified in the statement. If the value was found in a host variable, then the host variable number is &1. The proper ranges for date, time, or timestamp values are as follows: • The range for years is from 0001 to 9999. • The range for months is from 1 to 12. • The range for days is from 1 - 30 for April, June, September, and November, from 1 - 28 for February and from 1 to 31 for all other months. • In a leap year, the range for February can be from 1 to 29. • The range for days in a Julian date is from 001 to 366 for a leap year or 001 to 365 days for all other years. • The range for hours is from 0 to 24. If the hour is 24, then the other parts of the time values must be zeros. If the time format is USA, then the hour cannot be greater than 12. • The range for minutes is from 0 to 59. • The range for seconds is from 0 to 59. • The range for microseconds is from 0 to 999999.
Recovery Text:	Ensure that the date, time, or timestamp value conforms to the ranges for the data type it represents. Try the request again.
SQLCODE or SQLCODEs:	+181, -181
SQLSTATE or SQLSTATEs:	01534, 22007

SQL0182	
Message Text:	A date, time, or timestamp expression not valid.
Cause Text:	One of the following has occurred: • An operand of addition is a date and the other is not a date duration. • An operand of addition is a time and the other is not a time duration. • An operand of addition is a timestamp and the other is not a duration. • An operand of subtraction is a date and the other is not a date, character, or date duration. • An operand of subtraction is a time and the other is not a time, character, or time duration. • An operand of subtraction is a timestamp and the other is not a timestamp, character, or duration.
Recovery Text:	Correct the arithmetic expression so that it contains a valid date, time, or timestamp expression. Try the request again.
SQLCODE or SQLCODEs:	-182
SQLSTATE or SQLSTATEs:	42816

SQL0183	
Message Text:	Result of date or timestamp expression not valid.

SQL0183	
Cause Text:	The result of an arithmetic operation is a date or timestamp that is not within the valid range of dates which are between 0001-01-01 and 9999-12-31. If the result is a date in the format YMD, MDY, DMY, or JUL then the year must be between 1940 and 2039. If this is a FETCH, embedded SELECT, SET or VALUES INTO, then the relative position of the host variable in the INTO clause is &1 and the host variable name is &2.
Recovery Text:	Correct the arithmetic expression or the data that was being processed at the time the error occurred. If the date format is YMD, MDY, DMY or JUL and the result is not between 1940 and 2039, then specify USA, ISO, EUR, or JIS for the date format. The date format can be specified on the STRSQL or CRTSQLxxx commands or can be changed for the job by using the CHGJOB command. Try the request again.
SQLCODE or SQLCODEs:	+183, -183
SQLSTATE or SQLSTATEs:	01535, 22008

SQL0184	
Message Text:	Parameter marker not valid in expression.
Cause Text:	A parameter marker cannot be used as an operand in a date/time arithmetic expression.
Recovery Text:	Correct the arithmetic expression. Try the request again.
SQLCODE or SQLCODEs:	-184
SQLSTATE or SQLSTATEs:	42610

SQL0187	
Message Text:	Use of labeled duration not valid.
Cause Text:	One of the following has occurred: • A labeled duration is specified but is not the operand of the operators plus or minus. • A labeled duration of years, months, or days is specified as the operand of addition or subtraction and the other operand is not date or timestamp. • A labeled duration of hours, minutes, or seconds is specified as the operand of addition or subtraction and the other operand is not time or timestamp. • A labeled duration of microseconds is specified as the operand of addition or subtraction and the other operand is not timestamp. • A labeled duration is specified as the left operand of subtraction. • The value specified for the labeled duration is not a numeric type.
Recovery Text:	Correct the use of the labeled duration. Try the request again.
SQLCODE or SQLCODEs:	-187
SQLSTATE or SQLSTATEs:	42816

SQL0188	
Message Text:	&1 not a valid string representation of a name.

SQL0188	
Cause Text:	The host variable contains a string representation of a name that is not valid for one of the following reasons: • The host variable is empty. • The first character is a period, a slash, or a blank. • The number of identifiers is greater than the maximum allowed for the name of the object. For example, the host variable identifies a table name but the host variable contains 4 or more identifiers. A table name can contain a maximum of 3 identifiers. A relational database name can contain a maximum of 1 identifier. • An identifier is too long. • A period not contained in a delimited identifier is followed by a period or a blank. • A slash not contained in a delimited identifier is followed by a slash or a blank. • A blank is followed by characters other than blanks. • A delimited identifier contains no characters. • A delimited identifier is followed by a character other than a period, a slash or a blank. • The ending delimiter is missing from a delimited identifier.
Recovery Text:	Change the name. Try the request again.
SQLCODE or SQLCODEs:	-188
SQLSTATE or SQLSTATEs:	22503, 28000, 2E000

SQL0189	
Message Text:	Coded Character Set Identifier &1 not valid.
Cause Text:	Coded Character Set Identifier (CCSID) &1 is not valid for one of the following reasons: • The CCSID is not EBCDIC. • The CCSID is not supported by the system. • The CCSID is not valid for the data type. • If the CCSID is specified for graphic data, then the CCSID must be a DBCS CCSID. • If the CCSID is specified for UCS-2 or UTF-16 data, then the CCSID must be a UCS-2 or UTF-16 CCSID. • If the CCSID is specified for CLOB, DBCLOB or DATALINK data, then the CCSID must not be 65535. • If there are multiple DataLink columns with FILE LINK CONTROL, they must all have the same CCSID. • The NORMALIZED clause can only be specified for a UTF-8 or UTF-16 CCSID.
Recovery Text:	Ensure that all CCSID values in the statement are supported by the system and are valid for the data type. For a list of valid CCSID values, refer to the DB2 UDB for iSeries SQL Reference topic in the Information Center, http://www.ibm.com/eserver/iseries/infocenter.
SQLCODE or SQLCODEs:	-189
SQLSTATE or SQLSTATEs:	22522

Message Text:	Attributes of column &3 in &1 in &2 not compatible.

Cause Text:	The attributes specified for column &3 in &1 in &2 are not compatible with the attributes of the existing column. Either the data type, the length, or the clause is not valid. • A numeric column cannot be changed to a type that is not numeric. • A character column cannot be changed to a DATE, TIME, TIMESTAMP, or a numeric column. • A DATE, TIME, or TIMESTAMP column can only be changed to a column of the same type. • A character column cannot be changed to DBCS-only column. • A character, graphic, DataLink, or Unicode column cannot be changed to a column with an incompatible CCSID. • A binary column cannot be changed to a column that is not binary. • A column that is not binary cannot be changed to a binary column. • A DataLink column cannot be changed to a column with a shorter length. • A column can be changed to a user-defined type if the type is promotable to the new type. A column that is a user-defined type cannot be changed to a different type. • Columns cannot be changed to or from a DataLink. • Columns cannot be changed to or from the ROWID data type. • The length of a column that allows null values cannot be greater than 32765 for fixed-length character or fixed-length binary, 32739 for varying-length character or varying-length binary, 16382 for fixed-length graphic, and 16369 for varying-length graphic. The length of a DataLink column cannot be greater than 32717. The length of a binary or character LOB column cannot be greater than 2147483647. The length of a double-byte LOB column cannot be greater than 1073741823. The length of a DBCS-open column cannot be less than 4. • Identity attributes can only be specified for a column defined as an identity column. DROP NOT NULL cannot be specified for an identity column. • DROP DEFAULT can only be specified if a default value is defined for the existing column and the column does not have NOT NULL as the null attribute.
Recovery Text:	Specify attributes that are compatible with column &3. Try the request again.
SQLCODE or SQLCODEs:	-190
SQLSTATE or SQLSTATEs:	42837
SQL0190	

SQL0191	
Message Text:	Mixed data or UTF-8 data not properly formed.
Cause Text:	A mixed data string or UTF-8 data string does not have the proper format. For mixed data, every shift-out character ('0E'X) must have a corresponding shift-in character ('0F'X). If these characters are not paired, the data is not valid. The conversion was from column or host variable &2 to column or host variable &4.
Recovery Text:	Ensure that all mixed character data has paired shift characters and that all UTF-8 data is valid. For more information about UTF-8, refer to the Programming support topic in the Information Center.
SQLCODE or SQLCODEs:	+191, -191
SQLSTATE or SQLSTATEs:	01547, 22504

SQL0192	
Message Text:	Argument of translation function not valid.
Cause Text:	The argument of the TRANSLATE, UCASE, UPPER, LCASE, or LOWER scalar function is a DBCS-only string. The argument must be SBCS, DBCS-open, or DBCS-either.
Recovery Text:	Change the argument of the function to one that is valid. Try the request again.
SQLCODE or SQLCODEs:	-192
SQLSTATE or SQLSTATEs:	42948

SQL0194	
Message Text:	KEEP LOCKS not allowed.
Cause Text:	KEEP LOCKS was specified for cursor &1 but is not allowed because the cursor is not opened for read only. The cursor must be opened for read only to allow locks to be kept. For an explanation of read only cursors, see the DB2 UDB for iSeries SQL Programming topic in the Information Center, http://www.ibm.com/eserver/iseries/infocenter.
Recovery Text:	Do not specify KEEP LOCKS, or specify a cursor that is read only.
SQLCODE or SQLCODEs:	-194
SQLSTATE or SQLSTATEs:	42948

SQL0195	
Message Text:	Last column of &1 in &2 cannot be dropped.
Cause Text:	An attempt was made to drop one or more columns using an ALTER TABLE statement. The columns cannot be dropped from table &1 in &2 because at least one of the existing columns must be preserved when altering a table.
Recovery Text:	Ensure table &1 in &2 will have at least one column once the ALTER statement is complete. Either remove the DROP of one of the columns and try the request again, or, if all of the columns should be removed, drop the table and create it again.
SQLCODE or SQLCODEs:	-195
SQLSTATE or SQLSTATEs:	42814

SQL0196	
Message Text:	Column &3 in &1 in &2 cannot be dropped.
Cause Text:	An attempt was made to drop column &3. The column cannot be dropped because a view, a constraint, or an index is dependent on the column and RESTRICT was specified, or the column is part of the partition key.
Recovery Text:	Specify CASCADE on the ALTER TABLE statement to drop the column and the views, constraints, or indexes that are dependent on it. If the column is part of the partition key, specify DROP PARTITIONING on the ALTER TABLE statement to remove the partitioning for the table. Try the request again.
SQLCODE or SQLCODEs:	-196
SQLSTATE or SQLSTATEs:	42817

SQL0197	
Message Text:	Column &1 cannot be qualified.
Cause Text:	Column names in an ORDER BY clause of a SELECT statement cannot be qualified if a UNION, EXCEPT, or INTERSECT operator is specified.
Recovery Text:	Remove the qualifier from the column name. Ensure the name specified in the ORDER BY clause is a named column of the result table. Try the request again.
SQLCODE or SQLCODEs:	-197
SQLSTATE or SQLSTATEs:	42877

SQL0198	
Message Text:	SQL statement empty or blank.
Cause Text:	The SQL statement is empty or blank. One of the following has occurred: • During precompiling, the SQL statement referred to has no text between the EXEC SQL and the ending delimiter. The statement is ignored. • While running a program containing SQL statements, the operand of a PREPARE or EXECUTE IMMEDIATE statement is blank or empty. The operand, host variable, or literal string that was the object of either the PREPARE or EXECUTE IMMEDIATE statement contained all blanks or was an empty string.
Recovery Text:	If precompiling, correct the statement or remove it and precompile the program again. If running a program containing SQL statements, correct the logic of the program to make certain that a valid SQL statement is provided before issuing a PREPARE or EXECUTE IMMEDIATE statement.
SQLCODE or SQLCODEs:	-198
SQLSTATE or SQLSTATEs:	42617

SQL0199	
Message Text:	Keyword &1 not expected. Valid tokens: &2.
Cause Text:	The keyword &1 was not expected here. A syntax error was detected at keyword &1. The partial list of valid tokens is &2. This list assumes that the statement is correct up to the unexpected keyword. The error may be earlier in the statement but the syntax of the statement seems to be valid up to this point.
Recovery Text:	Examine the SQL statement in the area of the specified keyword. A colon or SQL delimiter may be missing. SQL requires reserved words to be delimited when they are used as a name. Correct the SQL statement and try the request again.
SQLCODE or SQLCODEs:	-199
SQLSTATE or SQLSTATEs:	42601

SQL0203	
Message Text:	Name &1 is ambiguous.

SQL0203	
Cause Text:	The name &1 is ambiguous for one of the following reasons: • Two or more of the tables specified in a FROM clause contain columns with the name &1. The name specified can refer to a column name or a system column name in the table. • The name is specified in an ORDER BY clause and is the same as more than 1 result column name. • OLD_ROW and NEW_ROW are specified for an SQL trigger and a transition variable specified in the routine body is not qualified.
Recovery Text:	Qualify the column name with a table name or correlation name or use the AS clause to provide a unique result column name that can be specified in the ORDER BY clause. Qualify the transition variable in the trigger with the name specified for OLD_ROW or NEW_ROW. Try the request again.
SQLCODE or SQLCODEs:	-203
SQLSTATE or SQLSTATEs:	42702

SQL0204	
Message Text:	&1 in &2 type *&3 not found.
Cause Text:	&1 in &2 type *&3 was not found. If the member name is *ALL, the table is not partitioned. If this is an ALTER TABLE statement and the type is *N, a constraint or partition was not found. If this is not an ALTER TABLE statement and the type is *N, a function, procedure, trigger or sequence object was not found. If a function was not found, &1 is the service program that contains the function. The function will not be found unless the external name and usage name match exactly. Examine the job log for a message that gives more details on which function name is being searched for and the name that did not match.
Recovery Text:	Change the name and try the request again. If the object is a node group, ensure that the DB2 Multisystem product is installed on your system and create a nodegroup with the CRTNODGRP CL command. If an external function was not found, be sure that the case of the EXTERNAL NAME on the CREATE FUNCTION statement exactly matches the case of the name exported by the service program.
SQLCODE or SQLCODEs:	+204, -204
SQLSTATE or SQLSTATEs:	01532, 42704

SQL0205	
Message Text:	Column &1 not in table &2 in &3.
Cause Text:	A column with the name &1 does not exist in table or view &2 in schema &3.

SQL0205	
Recovery Text:	Do one of the following and try the request again: • Make certain that the column names, table names, and any qualifiers are specified correctly. • If the column is not qualified, the column &1 is no longer in table &2. It was originally found in table &2, but it no longer exists. If the column is now available in a different table and is referenced by this statement, a precompile may be necessary. • If more than one table is referenced in a SQL statement, the column name should be qualified. • If this is a CREATE TABLE statement and column &1 is specified in a partitioning key or constraint for the table being created, add a column definition for column &1 or remove it from the constraint or partitioning key. • For a recursive common table expression, the SET sequence column name and the USING column name cannot be referenced in the fullselect that defines the common table expression.
SQLCODE or SQLCODEs:	-205
SQLSTATE or SQLSTATEs:	42703

SQL0206	
Message Text:	Column &1 not in specified tables.
Cause Text:	&1 is not a column of table &2 in &3. If the table is *N, &1 is not a column of any table or view that can be referenced.
Recovery Text:	Do one of the following and try the request again: • Ensure that the column and table names are specified correctly in the statement. • If this is a SELECT statement, ensure that all the required tables were named in the FROM clause. • If the column was intended to be a correlated reference, qualify the column with the correct table designator.
SQLCODE or SQLCODEs:	-206
SQLSTATE or SQLSTATEs:	42703

SQL0208	
Message Text:	ORDER BY column &1 or expression not in result table.
Cause Text:	Column &1 is specified in the ORDER BY clause and is not valid because it does not refer to a named column of the result table when a UNION, EXCEPT, or INTERSECT operator is specified. The result column is named if the corresponding columns in each SELECT list have the same name.
Recovery Text:	Do one of the following and try the request again: • Ensure &1 is a named result column if UNION, EXCEPT, or INTERSECT is specified. • Specify a numeric column identifier in place of &1 in the ORDER BY clause.
SQLCODE or SQLCODEs:	-208
SQLSTATE or SQLSTATEs:	42707

SQL0212	
Message Text:	Duplicate table designator &1 not valid.
Cause Text:	One of the following has occurred: • More than one table in a FROM clause of a subselect has a table designator with the name &1. If a correlation name is specified, the correlation name is the table designator. If one is not specified, the table name or view name is the table designator. If SQL naming is specified, the table name consists of the implicit or explicit schema name followed by the actual table name. If system naming is specified, the table name itself is used without a qualifier as the table designator. The table designator must be unique on each level of a subselect. • Names specified in the REFERENCING clause of a CREATE TRIGGER statement are not unique. The names specified for the NEW and OLD correlation name and the NEW_TABLE and OLD_TABLE names must be unique and must not be the same as the table on which the trigger is being defined.
Recovery Text:	Make certain there is a unique table designator for every table in a FROM clause for the same level of a subselect. Since schema-name/table-name cannot be used to qualify a column, the table name must be unique or a correlation name must be specified. Specify unique names in the REFERENCING clause of the CREATE TRIGGER statement. Correct any errors and try the request again.
SQLCODE or SQLCODEs:	-212
SQLSTATE or SQLSTATEs:	42712

SQL0213	
Message Text:	Parameter &1 not in routine &2 in &3.
Cause Text:	A parameter with a name &1 does not exist in routine &2 in schema &3.
Recovery Text:	Make certain that the parameter name, routine name, and any qualifiers are specified correctly. Try the request again.
SQLCODE or SQLCODEs:	-213
SQLSTATE or SQLSTATEs:	42703

SQL0214	
Message Text:	ORDER BY expression is not valid.
Cause Text:	The expression in the ORDER BY clause in position &1 is not valid for reason code &3. 1 - The SELECT statement contains a UNION, EXCEPT, or INTERSECT. 2 - DISTINCT is specified in the SELECT clause and the expression or column cannot be matched exactly with an expression or column in the select list. 3 - The select list uses a column function or there is a GROUP BY clause and the expression is not a column function or does not match exactly with an expression in the select list. 4 - A column function in the ORDER BY clause requires grouping.
Recovery Text:	Make the change listed for reason &3 and try the request again: 1 - Remove the expression from the ORDER BY clause. 2 - Remove DISTINCT from the select clause or change the ORDER BY expression or column to refer to a select list item by using a numeric column identifier or a column name. 3 - Change the expression in the ORDER BY clause to a column function or change to use a numeric column identifier or a column name. 4 - Add a GROUP BY clause or remove the column function from the ORDER BY clause.
SQLCODE or SQLCODEs:	-214

SQL0214	
SQLSTATE or SQLSTATEs:	42822

SQL0216	
Message Text:	Number of values in predicate lists do not match.
Cause Text:	The number of values on the left side of the predicate does not match the number of values on the right side of the predicate. If one side of the predicate is a fullselect that returns more than one result column, the other side of the predicate must be a list of expressions containing the same number of values. The fullselect must explicitly list the result columns. If both sides of the predicate are expression lists, the lists must contain the same number of values. An expression list used in an IN or NOT IN predicate cannot contain an untyped parameter marker.
Recovery Text:	Change the number of values in the predicate list or in the select list of a row fullselect so both sides contain the same number of values. Change a SELECT * in the fullselect to list the actual columns. Use a casting function for untyped parameter markers in an IN expression list.
SQLCODE or SQLCODEs:	-216
SQLSTATE or SQLSTATEs:	428C4

SQL0221	
Message Text:	Number of rows &2 not valid.
Cause Text:	A blocked FETCH, a blocked INSERT, or a SET RESULT SETS statement is not valid. The number of rows specified is not between 0 and 32767 or is greater than the dimension of the host structure array. The number of rows specified is &2 and the dimension of the array is &3. If this is a FETCH statement, the cursor name is &1.
Recovery Text:	Either ensure the number of rows is from 0 through 32767 and less than or equal to the dimension of the array, or increase the size of the array.
SQLCODE or SQLCODEs:	-221
SQLSTATE or SQLSTATEs:	42873

SQL0225	
Message Text:	FETCH not valid; cursor &1 not scrollable.
Cause Text:	A FETCH statement was specified with PRIOR, FIRST, LAST, BEFORE, AFTER, CURRENT, or RELATIVE for cursor &1, but cursor &1 is not scrollable. Only NEXT may be used for cursors that are not scrollable.
Recovery Text:	In order to specify PRIOR, FIRST, LAST, BEFORE, AFTER, CURRENT or RELATIVE on the FETCH statement, the cursor must be scrollable. To create a scrollable cursor, add the SCROLL keyword to the DECLARE CURSOR statement for cursor &1. SCROLL can also be specified in the attributes string for the prepared statement associated with the cursor.
SQLCODE or SQLCODEs:	-225
SQLSTATE or SQLSTATEs:	42872

SQL0226	
Message Text:	Current row deleted or moved for cursor &1.
Cause Text:	A FETCH CURRENT was specified for scrollable cursor &1. The current row was either deleted or updated. If the row was updated, one of the following could have occurred: • A value of an ORDER BY column of the current row has changed. • A value of a column in the index has changed. • A column has been changed so it no longer meets the record selection criteria.
Recovery Text:	Specify NEXT, PRIOR, FIRST, LAST, BEFORE, AFTER, or RELATIVE on the FETCH statement to position the cursor and fetch another row.
SQLCODE or SQLCODEs:	-226
SQLSTATE or SQLSTATEs:	24507

SQL0227	
Message Text:	FETCH not valid, cursor &1 in unknown position.
Cause Text:	A previous blocked FETCH for cursor &1 resulted in an error (SQLCODE &2, SQLSTATE &3) in the middle of processing a block of rows retrieved from the database manager. One or more rows left in the block could not be returned to the program following the error, leaving the position of the cursor unknown. If the SQLSTATE is *N, the error is unknown.
Recovery Text:	Close and reopen the cursor to set the position. For scrollable cursors, FIRST, LAST, BEFORE, or AFTER may also be used to position the cursor.
SQLCODE or SQLCODEs:	-227
SQLSTATE or SQLSTATEs:	24513

SQL0228	
Message Text:	FOR UPDATE clause not valid with SCROLL for cursor &1.
Cause Text:	The FOR UPDATE clause and SCROLL keyword are specified for cursor &1. The FOR UPDATE clause is not valid with the SCROLL keyword unless the DYNAMIC keyword is also specified. If SCROLL is specified and DYNAMIC is not specified, the cursor is read-only. If DYNAMIC SCROLL is specified, the cursor can be updated.
Recovery Text:	To declare a scrollable cursor that is read-only, specify the SCROLL keyword but do not specify the FOR UPDATE clause. To declare a scrollable cursor that can be updated, specify DYNAMIC SCROLL. Precompile the program again.
SQLCODE or SQLCODEs:	-228
SQLSTATE or SQLSTATEs:	42620

SQL0231	
Message Text:	Position of cursor &1 not valid for FETCH of current row.
Cause Text:	A FETCH CURRENT or a FETCH RELATIVE 0 was specified for scrollable cursor &1. The operation is not valid because the cursor is not positioned on a record. A FETCH of the current row is not allowed following a FETCH BEFORE, a FETCH AFTER, or a FETCH that resulted in an SQLCODE of +100.
Recovery Text:	Ensure the cursor is positioned on a record before attempting to fetch the current row.

SQL0231	
SQLCODE or SQLCODEs:	-231
SQLSTATE or SQLSTATEs:	22006

SQL0237	
Message Text:	Not enough SQLVAR entries were provided in the SQLDA.
Cause Text:	The SQLDA only provided &2 SQLVAR entries. Since at least one of the columns being described is a distinct type or a LOB, &3 SQLVAR entries should have been specified. None of the secondary SQLVAR entries have been set. Since at least one of the columns is a distinct type or a LOB, space should be provided for twice as many SQLVAR entries as the number of columns. Only the base SQLVAR entries have been set.
Recovery Text:	If there is no need for the additional information about the distinct type(s) or LOB(s), then no action is required. If this information is needed, the value of the SQLN field in the SQLDA should be increased to the value indicated in the message, and the statement should be resubmitted.
SQLCODE or SQLCODEs:	+237
SQLSTATE or SQLSTATEs:	01594

SQL0239	
Message Text:	Not enough entries were provided in the SQLDA or descriptor area.
Cause Text:	The SQLDA or descriptor area only provided &1 entries. This is the number of SQLVAR entries for the SQLDA or the value of DB2_MAX_ITEMS for a descriptor area. At least &2 entries should have been specified. None of the entries have been set. If a SQLDA is being used, at least one of the columns being described is a distinct type or a LOB. For the SQLDA, if any of the columns is a distinct type or a LOB, then space should be provided for twice as many SQLVAR entries as the number of columns.
Recovery Text:	For a descriptor, allocate more entries. For a SQLDA, if the distinct type or LOB information is needed, the value of the SQLN field should be increased to the value indicated in the message, and the statement should be resubmitted. If there is no need for the additional information about the distinct type(s) or LOB(s), then it is possible to resubmit the statement only providing enough SQLVAR entries to accommodate the number of columns.
SQLCODE or SQLCODEs:	+239
SQLSTATE or SQLSTATEs:	01005

SQL0242	
Message Text:	Duplicate partition name or number &1.
Cause Text:	Partition name or partition number &1 was already specified. The partition name or number must be unique.
Recovery Text:	Specify a unique name or number for the partition. Try the request again.
SQLCODE or SQLCODEs:	-242
SQLSTATE or SQLSTATEs:	42713

SQL0243	
Message Text:	SENSITIVE cursor &1 cannot be defined for the specified SELECT statement.
Cause Text:	The cursor &1 is defined as SENSITIVE but the query requires the creation of a temporary result table. A SENSITIVE cursor cannot be implemented.
Recovery Text:	Redefine the cursor as ASENSITIVE or INSENSITIVE or change the query so that it no longer requires the creation of a temporary result table.
SQLCODE or SQLCODEs:	-243
SQLSTATE or SQLSTATEs:	36001

SQL0250	
Message Text:	Local relational database not defined in the directory.
Cause Text:	One of the following has occurred: • Three part names were used and the relational database name is not defined in the relational database directory. • A connect was attempted and the relational database name is not defined in the relational database directory. • The SQL statement uses the CURRENT SERVER special register and the local relational database name is not defined in the relational database directory. • The SQL statement referred to a view which used the CURRENT SERVER special register and the the local relational database name is not defined in the relational database directory.
Recovery Text:	Define the local relational database name using the Add Relational Database Directory Entry (ADDRDBDIRE) command.
SQLCODE or SQLCODEs:	-250
SQLSTATE or SQLSTATEs:	42718

SQL0251	
Message Text:	Character in relational database name &1 not valid.
Cause Text:	&1 contains either a #, @, ., or a $, which are not valid character for a relational database name. Valid characters include A-Z, 0-9, and underscore.
Recovery Text:	Correct the name. Try the request again.
SQLCODE or SQLCODEs:	-251
SQLSTATE or SQLSTATEs:	2E000, 42602

SQL0255	
Message Text:	Function not supported for query.

SQL0255	
Cause Text:	The reason code is &1: • 1 - Scalar subselects and lateral correlation from a nested table expression are not allowed with distributed files • 2 - Error occurred while using a temporary distributed file. • 3 - EXCEPT or INTERSECT not supported for this query. • 4 - A sequence reference is not supported with distributed files. • 5 - A recursive common table expression is not supported for this query. • 6 - An OLAP function is not supported for this query.
Recovery Text:	A list of corrective actions follow: • If code 1, change the query so it does not use scalar subselects or correlation from a nested table expression. • If code 2, see the previous messages for more information. • If code 3, remove EXCEPT or INTERSECT from the query. • If code 4, remove the sequence reference from the query. • If code 5, remove the recursive common table expression from the query. • If code 6, remove the OLAP function from the query.
SQLCODE or SQLCODEs:	-255
SQLSTATE or SQLSTATEs:	42999

SQL0256	
Message Text:	Constraint &1 in &2 not allowed on distributed file.
Cause Text:	Constraint &1 in &2 not allowed for one of the following reasons: - The columns that make up the partitioning key must be a subset of the columns that make up the foreign key. The columns may appear in any order. - The node group of the dependent table in a foreign key constraint must match the node group of the parent table.
Recovery Text:	Ensure that every column that is in the partitioning key is also in the foreign key for the table. Also ensure that the dependent table and the parent table are built over the same nodegroup.
SQLCODE or SQLCODEs:	-256
SQLSTATE or SQLSTATEs:	42998

SQL0270	
Message Text:	Function not allowed for table &1 in &2.
Cause Text:	Table &1 in &2 is a distributed table or a partitioned table. The function is not allowed for one of the following reasons: - The unique index or unique constraint is not allowed because all unique indexes or unique constraints of a distributed table or a partitioned table must contain all columns that make up the partitioning key. If this is a CREATE TABLE statement and the PARTITIONING KEY clause was not specified, then the default partitioning key is the first column of the primary key, or the first valid column of the table. - Data in one of the partitioning key columns was changed by an UPDATE statement which would have forced the row to a different node. - The table contains a LOB column. LOB columns are not allowed in a distributed table. - The node group of the materialized query table is different than the node group of one or more tables referenced by the materialized query table. - A column of the result table is an XML type.

SQL0270	
Recovery Text:	Ensure that all unique indexes or unique constraints contain all the columns of the partitioning key. Ensure that data in the partitioning key columns is not changed, or is changed to a value that would reside on the same node. Ensure the table does not contain any LOB columns. Ensure that the materialized query table references tables within the same node group. Ensure that XML columns are not included in the result table.
SQLCODE or SQLCODEs:	-270
SQLSTATE or SQLSTATEs:	42990

SQL0301	
Message Text:	Input host variable &2 or argument &1 not valid.
Cause Text:	The value in relative position &1 in the statement is a type that is not compatible with the requested operation. The value is host variable &2, entry &1 in a descriptor area, or argument &2 in a CALL statement. A name *N indicates that a user's descriptor area was used or that a constant or special register was specified on the CALL statement.
Recovery Text:	Do one of the following and try the request again: • Use a host variable that is the correct type. • Specify an argument in the CALL that is the correct type. • Change the type specified for parameter &1 in the DECLARE PROCEDURE statement.
SQLCODE or SQLCODEs:	-301
SQLSTATE or SQLSTATEs:	07006, 42895

SQL0302	
Message Text:	Conversion error on host variable or parameter &2.
Cause Text:	Host variable or parameter &2 or entry &1 in a descriptor area contains a value that cannot be converted to the attributes required by the statement. Error type &3 occurred. Error types and their meanings are: • 1 - Overflow. • 2 - Floating point overflow. • 3 - Floating point underflow. • 4 - Floating point conversion error. • 5 - Not an exact result. • 6 - Numeric data that is not valid. • 7 - Double-byte character set (DBCS) or UTF-8 data that is not valid. • 8 - C NUL-terminator is missing for character host variables or double NUL-terminator is missing for graphic host variables and the program was compiled with the *CNULRQD option. • 9 - Truncation when mapping a host variable or constant to a character or binary parameter on a CALL statement. • 10 - Incompatible conversion from the input SQLDATA value to the specified SQLTYPE in a REXX application. • 11 - Overflow on translation of UTF-8 character data. If the host variable name is *N and the statement is FETCH, a descriptor area was specified. If the parameter name is *N and the statement is CALL, a descriptor area, a constant, or a special register was specified.
Recovery Text:	Change the value of the host variable or parameter or entry in the descriptor area so that it can be converted and is valid. Try the request again.

SQL0302	
SQLCODE or SQLCODEs:	-302
SQLSTATE or SQLSTATEs:	22001, 22003, 22023, 22024

SQL0303	
Message Text:	Host variable &2 not compatible.
Cause Text:	A FETCH, SELECT, CALL, SET, VALUES INTO, GET DIAGNOSTICS, GET DESCRIPTOR, or SET DESCRIPTOR cannot be performed because the data type of host variable &2 is not compatible with the data type of the corresponding list item: • When selecting a date value, a character host variable must be at least 6 bytes for a Julian date, at least 8 bytes for a date in the MDY, YMD, DMY formats, or at least 10 bytes for all other formats. • When selecting a time value, a character host variable must be at least 8 bytes for a time in the USA format and at least 5 bytes for all other formats. • When selecting a timestamp value, a character host variable must be at least 19 bytes. • If the host variable is C NUL-terminated and the program was compiled with *CNULRQD option, then an additional byte is required for the NUL-terminator for date/time values. • For GET DIAGNOSTICS ALL, the host variable must be varying length character or varying length graphic. The relative position of the host variable in the INTO clause, the SQLDA, or the CALL statement is &1. If the host variable name is *N, a descriptor area was specified on a FETCH statement.
Recovery Text:	Ensure that the data types are compatible for each of the corresponding list items. Ensure the host variables are defined correctly for date, time, and timestamp values.
SQLCODE or SQLCODEs:	-303
SQLSTATE or SQLSTATEs:	22001, 42806

SQL0304	
Message Text:	Conversion error in assignment to host variable &2.
Cause Text:	During an attempt to return a value to host variable &2 on a FETCH, an embedded SELECT statement, a CALL statement, a SET statement, or a VALUES INTO statement. error type &3 occurred. A list of the error types follows: • 1 - Overflow. • 2 - Floating point overflow. • 3 - Floating point underflow. • 4 - Floating point conversion error. • 5 - Not an exact result. • 6 - Numeric data that is not valid. • 7 - Double-byte character set (DBCS) data that is not valid. The relative position of the host variable is &1. If the host variable name is *N, a descriptor area was specified on the FETCH or CALL statement.
Recovery Text:	Change the size and, if necessary, the type of the host variable or entry in the descriptor area so that it can contain the result value or correct the data that is not valid. Precompile the program again.
SQLCODE or SQLCODEs:	+304, -304

SQL0304	
SQLSTATE or SQLSTATEs:	01515, 01547, 01565, 22003, 22023, 22504

SQL0305	
Message Text:	Indicator variable required.
Cause Text:	A FETCH, embedded SELECT, CALL, GET DESCRIPTOR, or a SET or VALUES INTO statement has resulted in a null value, but an indicator variable was not specified for host variable &2. The relative position of the host variable in the INTO clause or parameter list is &1. If the host variable name is *N, a descriptor area was specified. If this error occurs on a GET DESCRIPTOR statement, the null value is being returned but the INDICATOR item was not specified on the GET DESCRIPTOR statement.
Recovery Text:	Specify an indicator variable, and precompile the program again. If this is a GET DESCRIPTOR statement, specify both the DATA item and the INDICATOR item. Precompile the program again.
SQLCODE or SQLCODEs:	-305
SQLSTATE or SQLSTATEs:	22002, 22004

SQL0306	
Message Text:	REXX input host variable &1 not defined.
Cause Text:	The REXX input host variable &1 appears in an SQL statement, but it is not defined because a value has not been assigned to the variable.
Recovery Text:	Verify that &1 is spelled correctly in the SQL statement and that a value is assigned to the host variable before the SQL statement is run.
SQLCODE or SQLCODEs:	-306
SQLSTATE or SQLSTATEs:	42863

SQL0311	
Message Text:	Length in a varying-length or LOB host variable not valid.
Cause Text:	Host variable &2 was specified. The value in the length portion of the variable length or LOB host variable is either negative or greater than the declared length. If the host variable is graphic the length should be the number of DBCS characters. The host variable number is &1. The specified length is &4. The variable is declared to have length &3.
Recovery Text:	Change the length portion of the varying-length or LOB host variable to a valid positive number or zero. Try the request again.
SQLCODE or SQLCODEs:	-311
SQLSTATE or SQLSTATEs:	22501

SQL0312	
Message Text:	Variable &1 not defined or not usable.

SQL0312	
Cause Text:	The variable &1 appears in the SQL statement, but one of the following conditions exists: • No declaration for the variable exists. • The attributes are not correct for the use specified. • The host variable was specified in dynamic SQL. Host variables are not valid in dynamic SQL. • In REXX, host variable names cannot contain embedded blanks. • The variable name is used in the routine body of an SQL procedure or function, but the variable is not declared as an SQL variable or parameter. The scope of an SQL variable is the compound statement that contains the declaration. • The variable is used in the routine body of an SQL trigger, but the variable is not declared as an SQL variable or the variable is an OLD transition variable and cannot be modified. • The variable is a transition variable in an AFTER trigger and is used in statement where the variable could be modified. Modifying transition variables in AFTER triggers is not allowed.
Recovery Text:	Do one of the following and try the request again: • Verify that &1 is spelled correctly in the SQL statement. • Verify that the program contains a declaration for that variable. • Verify that the attributes of the variable are compatible with its use in the statement. • Use parameter markers in dynamic SQL instead of host variables. • Remove embedded blanks from REXX host variable names. • Declare the variable as an SQL variable or parameter in the SQL procedure or function. • Declare the variable as an SQL variable or specify a NEW transition variable when the variable is modified in an SQL trigger. • Remove the transition variable from the statement. Copying the transition variable to a local variable and then using the local variable in the statement is also acceptable.
SQLCODE or SQLCODEs:	-312
SQLSTATE or SQLSTATEs:	42618

SQL0313	
Message Text:	Number of host variables not valid.
Cause Text:	The number of host variables or entries in an SQLDA or descriptor area specified in either an EXECUTE or OPEN statement is not the same as the number of parameter markers specified in the prepared SQL statement &1. If the statement name is *N, the number of host variables or entries in a SQLDA or descriptor area was specified in an OPEN statement and is not the same as the number of host variables specified in the DECLARE CURSOR statement for cursor &2.
Recovery Text:	Change the number of host variables specified in the USING clause or the number of entries in the SQLDA or descriptor area to equal the number of parameter markers in the prepared SQL statement or the number of host variables in the DECLARE CURSOR statement. Precompile the program again.
SQLCODE or SQLCODEs:	-313
SQLSTATE or SQLSTATEs:	07001, 07004

SQL0326	
Message Text:	Too many host variables specified.
Cause Text:	&1 host variables were specified on the FETCH, embedded SELECT, SET, or VALUES INTO statement, but only &2 columns were returned from the query. Extra host variables will be filled with the appropriate default value for the specified type. • Character host variables will be filled with blanks. • Binary host variables will be filled with hex zeros. • Date host variables will be filled with the current date. • Time host variables will be filled with the current time. • Timestamp host variables will be filled with the current timestamp. • Graphic host variables will be filled with double-byte, UCS-2, or UTF-16 blanks. • Varying-length character, varying-length graphic, varying-length binary, CLOB, DBCLOB, and BLOB host variables will be set to a length of 0. • C NUL-terminated character host variables will have a NUL-terminator set into the first character position. • C NUL-terminated graphic host variables will have a double NUL-terminator set into the first DBCS character position. • Numeric host variables will be set to a value of 0. • REXX host variables will be defaulted to variable length character with the length set to 0.
Recovery Text:	No recovery is necessary.
SQLCODE or SQLCODEs:	+326
SQLSTATE or SQLSTATEs:	01557

SQL0327	
Message Text:	Partitioning key does not correspond to a defined partition.
Cause Text:	An INSERT, CREATE TABLE, or ALTER TABLE statement was attempted for a partitioned table but the values supplied for the partitioning key did not qualify the row for any partition.
Recovery Text:	If the error occurred on an INSERT statement, change the values supplied for the row being inserted so that they correspond to a partition of the table or alter the partition definition to allow this key value. If the error occurred on a CREATE TABLE or ALTER TABLE statement, alter the partition definitions so that all existing rows will fit in the table or delete the rows which do not fit. Try the request again.
SQLCODE or SQLCODEs:	-327
SQLSTATE or SQLSTATEs:	22525

SQL0328	
Message Text:	Column &1 not allowed in partitioning key.
Cause Text:	Column &1 is not allowed to be part of the partitioning key. If the column name is *N, then there are no valid columns for use as the default partitioning key for DB2 Multisystem partitioning. Columns of type DATE, TIME, TIMESTAMP, FLOAT, DATALINK, and LOB or a distinct type based on one of these types are not allowed in the partitioning key.
Recovery Text:	Remove the column from the list of partitioning key columns, or change the data type of the column.

SQL0328	
SQLCODE or SQLCODEs:	-328
SQLSTATE or SQLSTATEs:	42996

SQL0329	
Message Text:	The SET PATH name list is not valid.
Cause Text:	String constant or input host variable &1 contains a SET PATH name list that is not valid. A host variable name *N indicates that an incorrect string constant was specified on the SET PATH statement or for the SQLPATH on the SET OPTION statement. The name list must have the following attributes: • The length must be greater than 0. • The length cannot be greater than 3483. • It must contain a list of valid schema names, separated by commas. • The list can contain a maximum of 268 schema names. • Each schema name must be capitalized unless it is a delimited name. • The list cannot contain the special values, *LIBL, CURRENT PATH, SYSTEM PATH or USER.
Recovery Text:	Use a string constant or host variable with the correct value.
SQLCODE or SQLCODEs:	-329
SQLSTATE or SQLSTATEs:	0E000

SQL0330	
Message Text:	Character conversion cannot be performed.
Cause Text:	An attempt was made to convert column or host variable &2 to column or host variable &3. The conversion cannot be performed. If the source data is character and has a mixed Coded Character Set Identifier (CCSID), then double-byte characters were found. These mixed data conversions are only allowed if the source data does not contain any double-byte data. If the data is graphic, the CCSID values are not compatible or the string contains single-byte characters. The source CCSID is &4, and the target CCSID is &5.
Recovery Text:	Ensure that all character or graphic comparison, concatenation, or assignment is between columns or host variables with compatible CCSID values. If character data and the source CCSID is mixed, the source data should not contain any double-byte characters. If graphic data, the string cannot contain single-byte characters. Use a casting function like VARCHAR to convert between character, DBCS graphic, and Unicode graphic data.
SQLCODE or SQLCODEs:	-330
SQLSTATE or SQLSTATEs:	22021

SQL0331	
Message Text:	Character conversion cannot be performed.

SQL0331	
Cause Text:	An attempt was made to convert column or host variable &2 to column or host variable &3. The conversion cannot be performed. If the source data is character and has a mixed Coded Character Set Identifier (CCSID), then double-byte characters were found. Mixed CCSID character conversions are only allowed if the source data does not contains any double-byte data. If the data is graphic, the CCSID values are not compatible. The source CCSID is &4, and the target CCSID is &5.
Recovery Text:	Ensure that all character or graphic assignments are between columns or host variables with compatible CCSID values. If character and the source CCSID is mixed, the source data should not contain any double-byte characters. Use a casting function like VARCHAR to convert between character, DBCS graphic, and UCS-2 and UTF-16 graphic data.
SQLCODE or SQLCODEs:	+331, -331
SQLSTATE or SQLSTATEs:	01520, 22021

SQL0332	
Message Text:	Character conversion between CCSID &1 and CCSID &2 not valid.
Cause Text:	Character or graphic conversion has been attempted for data that is not compatible. There is no conversion defined between CCSID &1 and CCSID &2. If one CCSID is 65535, the other CCSID is a graphic CCSID. Conversion is not defined between 65535 and a graphic CCSID. If this is a CONNECT statement, conversion is not defined between the default application requester SBCS CCSID and the application server SBCS CCSID. If the second CCSID is 0, the application server did not return its default SBCS CCSID. An application server other than an iSeries may not support a CCSID of 65535.
Recovery Text:	Ensure that all character or graphic comparisons, concatenation, or assignments are between columns or host variables with compatible CCSID values. If this is a CONNECT statement, change either the SBCS CCSID of the application requester or the application server, so conversion between the CCSID values is defined.
SQLCODE or SQLCODEs:	-332
SQLSTATE or SQLSTATEs:	57017

SQL0334	
Message Text:	Character conversion resulted in truncation.
Cause Text:	Character conversion of column or host variable &2 has resulted in truncation. An attempt was made to convert mixed ASCII data to mixed EBCDIC data or to convert UCS-2 or UTF-16 graphic data to mixed EBCDIC data. The length of the data has increased due to the insertion of shift characters. The resulting string did not fit in the target, and truncation occurred.
Recovery Text:	When converting from mixed ASCII to mixed EBCDIC or from UCS-2 or UTF-16 graphic to mixed EBCDIC, ensure that there is enough space in the target.
SQLCODE or SQLCODEs:	-334
SQLSTATE or SQLSTATEs:	22524

SQL0335	
Message Text:	Character conversion resulted in substitution characters.

SQL0335	
Cause Text:	Character column or host variable &2 has been converted to character column or host variable &3. The conversion defines that several different character values in the source data will translate to the same value in the target data. It will no longer be possible to separate these values. The CCSID of the source data is &4 and the CCSID of the target is &5.
Recovery Text:	Change the definition of the columns or host variables so that CCSID values that are used will allow all character values in the source to be converted to character values in the target.
SQLCODE or SQLCODEs:	+335
SQLSTATE or SQLSTATEs:	01517

SQL0336	
Message Text:	Identity or sequence attribute is not valid.
Cause Text:	For an identity column or a sequence, the values specified for the START WITH, INCREMENT BY, MINVALUE, MAXVALUE, and RESTART WITH options must have a scale of zero.
Recovery Text:	Change the value to one that is valid. Try the request again.
SQLCODE or SQLCODEs:	-336
SQLSTATE or SQLSTATEs:	428FA

SQL0338	
Message Text:	JOIN predicate not valid.
Cause Text:	The JOIN predicate is not valid because a column is specified that exists in a table that is outside the scope of the join predicate. The scope is generally determined from left to right but is also based on the position of the join-condition. If parentheses are used, columns inside the parentheses can not come from a table outside the parentheses.
Recovery Text:	Do one of the following and try the request again: • Make certain that the column names, table names, and any qualifiers are specified correctly. • Specify parentheses around joined tables to specify a join order other than left to right. Ensure columns exist in tables that are in the same scope.
SQLCODE or SQLCODEs:	-338
SQLSTATE or SQLSTATEs:	42972

SQL0340	
Message Text:	Duplicate name &1 for common table expressions.
Cause Text:	Name &1 cannot be used to define more than one table expression.
Recovery Text:	Change the name for one of the common table expressions. Try the request again.
SQLCODE or SQLCODEs:	-340

SQL0340	
SQLSTATE or SQLSTATEs:	42726

SQL0341	
Message Text:	Cyclic references between common table expressions.
Cause Text:	The common table expressions specified are not valid. The subselect for table &1 refers to table &2 and the subselect for table &2 refers to table &1. Cyclic references between common table expressions are not allowed.
Recovery Text:	Change the common table expressions to refer to a table that exists or a common table expression that has already been defined. Try the request again.
SQLCODE or SQLCODEs:	-341
SQLSTATE or SQLSTATEs:	42835

SQL0342	
Message Text:	Keyword not allowed in recursive common table expression &1.
Cause Text:	The common table expression &1 is recursive. The recursive common table expression contains one of the following errors: • A fullselect within the common table expression cannot start with SELECT DISTINCT because the common table expression is recursive. • A fullselect within the common table expression specified UNION instead of UNION ALL as required for recursive common table expressions.
Recovery Text:	Remove the keyword DISTINCT from the fullselect. Change the UNION to UNION ALL, or remove the recursive reference within the common table expression. Try the request again.
SQLCODE or SQLCODEs:	-342
SQLSTATE or SQLSTATEs:	42925

SQL0343	
Message Text:	Column list not valid for table.
Cause Text:	For a recursive common table expression: • The column name list must be specified following the table name of the common table expression. &1 is the common table expression name. • The sequence column name and the set cycle column name cannot be referenced in the column list of the recursive common table expression. &1 is the sequence column name or the set cycle column name.
Recovery Text:	Correct the recursive common table expression. Try the request again.
SQLCODE or SQLCODEs:	-343
SQLSTATE or SQLSTATEs:	42908

SQL0345	
Message Text:	Recursive common table expression &1 is not valid.

SQL0345	
Cause Text:	The recursive common table expression is not valid for one of the following reasons: • The initialization fullselect of the common table expression cannot refer to itself. • Grouping and column functions are not allowed within the fullselects of the UNION that define the common table expression. • EXCEPT DISTINCT is not allowed if the common table is specified as the right operand of a join. • LEFT OUTER JOIN is not allowed if the common table is specified as the right operand of the join. • RIGHT OUTER JOIN is not allowed if the common table is specified as the left operand of the join. • The common table expression cannot be referenced more than once in a FROM clause and cannot be referenced in a subquery.
Recovery Text:	Correct the recursive common table expression. Try the request again.
SQLCODE or SQLCODEs:	-345
SQLSTATE or SQLSTATEs:	42836

SQL0346	
Message Text:	Recursion not allowed for common table expressions.
Cause Text:	The common table expression specified is not valid. The subselect for table &1 refers to itself. Recursive common table expressions are not allowed.
Recovery Text:	Change the common table expressions to refer to a table that exists or a common table expression that has already been defined. Try the request again.
SQLCODE or SQLCODEs:	-346
SQLSTATE or SQLSTATEs:	42836

SQL0348	
Message Text:	&1 expression not used correctly.

SQL0348	
Cause Text:	The &1 sequence expression for &2 in &3 is not allowed. A NEXT VALUE or PREVIOUS VALUE sequence expression cannot be specified: • In a CREATE VIEW statement. • In a CREATE FUNCTION statement. • In a CREATE FUNCTION statIn a CHECK constraint.ement. • In a CREATE TABLE or ALTER TABLE statement that uses a select-statement to define the table. A PREVIOUS VALUE expression cannot be specified in a CREATE TRIGGER statement. A NEXT VALUE expression can only be specified: • In the SELECT clause of the outermost SELECT. It cannot be used if the outermost SELECT uses the DISTINCT keyword, contains a GROUP BY or ORDER BY clause, or is part of a fullselect. It cannot be used in a CASE expression. • In the SELECT clause of the outermost SELECT of a common table expression. It cannot be used if the outermost SELECT uses the DISTINCT keyword, contains a GROUP BY or ORDER BY clause, or is part of a fullselect. • As an expression in the SET clause of an UPDATE. It cannot be used in a subselect in the SET clause of an UPDATE. • As an INSERT value in either the VALUES clause or in the SELECT clause of the fullselect. • As a value in an SET or VALUES statement.
Recovery Text:	Remove the NEXT VALUE or PREVIOUS VALUE expression.
SQLCODE or SQLCODEs:	-348
SQLSTATE or SQLSTATEs:	428F9

SQL0350	
Message Text:	Column &1 not valid.
Cause Text:	One of the following errors has occurred: • A LOB or DataLink column is not valid as a key field, the key of an index, or the foreign key of a referential constraint. • A DataLink column with the FILE LINK CONTROL option cannot be in a table in QTEMP, QSYS, QSYS2, or SYSIBM, or in a temporary table. • A LOB or DataLink column or a distinct type based on a LOB or Datalink type is not valid as a partitioning key column for RANGE partitioning.
Recovery Text:	Remove the LOB or DataLink column from the specification of the table, index, key, or constraint.
SQLCODE or SQLCODEs:	-350
SQLSTATE or SQLSTATEs:	42962

SQL0351	
Message Text:	The AR is not at the same level and DB2 cannot transform the data type to a compatible type.

SQL0351	
Cause Text:	The data type of entry &1 is not supported on the Application Requester. The usual cause it that the Application Requester is at less than Level 6 for the SQL Access Manager, and the Application Server cannot transform the data type to a compatible type. It can also mean that an attempt was made to use SQLCI (SQL Client Integration) with an unsupported data type such as BLOB or DataLink.
Recovery Text:	Change the data type to one that is supported by the corresponding Application Requester or SQLCI.
SQLCODE or SQLCODEs:	-351
SQLSTATE or SQLSTATEs:	56084

SQL0352	
Message Text:	The AS is not at the same level and DB2 cannot transform the data type to a compatible type.
Cause Text:	The data type of entry &1 is not supported on the Application Server. The Application Server is at less than Level 6 for the SQL Access Manager, and the Application Requester cannot transform the data type to a compatible type.
Recovery Text:	Change the data type to one that is supported by the corresponding Application Server.
SQLCODE or SQLCODEs:	-352
SQLSTATE or SQLSTATEs:	56084

SQL0357	
Message Text:	File server &1 used in DataLink not currently available.
Cause Text:	Server &1 in the URL of the DataLink value is not available for reason code &2. The reason codes are as follows: • 1 - The file server in a Datalink value is not available. • 2 - The database server, instance, or database from which the operation was attempted is not registered with the file server. • 3 - Restart recovery is pending or is in progress on a file server involved in the operation. • 4 - The file server in a Datalink value is registered with the database but is an unknown server.
Recovery Text:	Verify that the server is running and can be accessed. Try the request again.
SQLCODE or SQLCODEs:	-357
SQLSTATE or SQLSTATEs:	57050

SQL0358	
Message Text:	Error &1 occurred using DataLink data type.

SQL0358	
Cause Text:	An error occurred while using a DataLink. Possible errors are: • 21 - Format of DataLink value is not valid. • 22 - The DataLink File Manager (DLFM) is not properly configured on the server. • 23 - Link type is not valid. • 24 - File does not exist. • 25 - File is already linked. • 26 - File is not available. • 27 - Length of comment or URL is not valid. • 28 - User is not authorized to link the file. • 29 - Datalink cannot be unlinked.
Recovery Text:	Correct that error in the DataLink and try the request again. For error type 22, it may be that the host database or the prefix have not been added to the DLFM on the server. If that is the case, use the commands Add Host Database to DLFM (ADDHDBDLFM) or Add Prefix to DLFM (ADDPFXDLFM) to correct the error.
SQLCODE or SQLCODEs:	-358
SQLSTATE or SQLSTATEs:	428D1

SQL0359	
Message Text:	Value for identity column or sequence not available.
Cause Text:	The value for the identity column or sequence is not available for one of the following reasons: • The INSERT or UPDATE statement cannot be run because all values for the identity column have already been assigned. • The NEXT VALUE expression cannot be evaluated because all values for the sequence have already been assigned.
Recovery Text:	For identity columns, alter the column to allow a larger range of values for the identity column or alter the column to allow for cycling of identity values. For sequences, alter the sequence to allow a larger range of values or to allow cycling of the sequence.
SQLCODE or SQLCODEs:	-359
SQLSTATE or SQLSTATEs:	23522

SQL0360	
Message Text:	DataLink in table &1 in &2 may not be valid due to pending links.
Cause Text:	Table &1 in schema &2 has DataLinks in link pending mode. While the DataLink can be retrieved using FETCH or SELECT INTO, the DataLink may not be valid because the table has DataLinks in link pending mode.
Recovery Text:	Verify that the value retrieved is a valid URL. The command WRKPFDL (Work with Physical File DataLinks) can be used to determine which tables have DataLinks in link pending mode.
SQLCODE or SQLCODEs:	+360
SQLSTATE or SQLSTATEs:	01627

SQL0372	
Message Text:	Only one ROWID or IDENTITY column allowed for table &2 in &3.
Cause Text:	Column &1 cannot be created in table &2 in &3 because column &4 has already been defined. There can be at most one IDENTITY column and one ROWID column defined in a table.
Recovery Text:	Remove one of the columns or change the attributes so that only a single column is defined as ROWID or as an IDENTITY column. Try the request again.
SQLCODE or SQLCODEs:	-372
SQLSTATE or SQLSTATEs:	428C1

SQL0373	
Message Text:	DEFAULT cannot be specified for column &1.
Cause Text:	One of the following has occurred: • The DEFAULT clause is specified for an IDENTITY column or ROWID column on the CREATE TABLE or ALTER TABLE statement. • A SET statement is specified in an SQL trigger that assigns DEFAULT to an IDENTITY column or a ROWID column that was defined as GENERATED BY DEFAULT.
Recovery Text:	Remove the DEFAULT clause from the definition of the column or change the SET statement so it does not assign a value of DEFAULT. Try the request again.
SQLCODE or SQLCODEs:	-373
SQLSTATE or SQLSTATEs:	42623

SQL0385	
Message Text:	SQL routine &1 in &2 created with assignment to SQLCODE or SQLSTATE.
Cause Text:	SQL routine &1 in schema &2 was created, but contains an assignment statement that specifies the SQLCODE or SQLSTATE variable as the target. Assignment statements that modify SQLCODE and SQLSTATE only change the value in the variable, the error or warning is not signaled.
Recovery Text:	No recovery is necessary. If the intent was to signal an error or warning, specify the SIGNAL or RESIGNAL statement instead of the assignment.
SQLCODE or SQLCODEs:	+385
SQLSTATE or SQLSTATEs:	01643

SQL0387	
Message Text:	No additional result sets returned.
Cause Text:	Procedure &1 in &2 was defined to return a maximum number of &4 result sets. The procedure returned &3 result sets.
Recovery Text:	None.
SQLCODE or SQLCODEs:	+387
SQLSTATE or SQLSTATEs:	02001

SQL0390	
Message Text:	Use of function &1 in &2 not valid.
Cause Text:	Use of function &1 in schema &2 is not valid. The specific name is &3. One of the following has occurred: • A table function was specified in a clause other than the FROM clause. • A function was specified in the FROM clause but the function is not a table function. • A table function was specified as a source function in a CREATE FUNCTION statement.
Recovery Text:	Remove the function from the clause or change the function name, arguments, or path so that a different function is found. Try the request again.
SQLCODE or SQLCODEs:	-390
SQLSTATE or SQLSTATEs:	42887

SQL0391	
Message Text:	Table function cannot be argument of function &1.
Cause Text:	The table designator for a table function cannot be used as the argument of function &1 in schema &2.
Recovery Text:	Use a table designator that does not represent a table function as the argument of this function. Try the request again.
SQLCODE or SQLCODEs:	-391
SQLSTATE or SQLSTATEs:	42881

SQL0392	
Message Text:	Assignment of LOB to specified host variable not allowed.
Cause Text:	The target host variable for all fetches of this LOB value for cursor &1 must be a locator or a LOB host variable.
Recovery Text:	Change the target of this fetch to either a LOB host variable or a LOB locator to be consistent with other fetches for this cursor. If it is necessary to use both LOB host variables and LOB locators as targets for this fetch, use the *NOOPTLOB compiler option.
SQLCODE or SQLCODEs:	-392
SQLSTATE or SQLSTATEs:	42855

SQL0393	
Message Text:	Value specified for condition or diagnostic is not valid.
Cause Text:	The value specified for the DIAGNOSTICS SIZE on the SET TRANSACTION statement is out of range, or the condition number specified on the GET DIAGNOSTICS statement is greater than the number of conditions available for the previous SQL statement.
Recovery Text:	Specify a correct value.
SQLCODE or SQLCODEs:	-393

SQL0393	
SQLSTATE or SQLSTATEs:	35000

SQL0398	
Message Text:	AS LOCATOR cannot be specified for a non-LOB parameter.
Cause Text:	AS LOCATOR is only allowed for LOB parameters to a procedure or function.
Recovery Text:	Use AS LOCATOR only for a LOB parameter to a procedure or function.
SQLCODE or SQLCODEs:	-398
SQLSTATE or SQLSTATEs:	428D2

SQL0399	
Message Text:	Value for ROWID column &1 not valid.
Cause Text:	The INSERT or UPDATE statement cannot be run because the value specified for ROWID column &1 is not valid.
Recovery Text:	Remove column &1 from the column list or specify a valid ROWID value or DEFAULT for column &1.
SQLCODE or SQLCODEs:	-399
SQLSTATE or SQLSTATEs:	22511

SQL0401	
Message Text:	Comparison operator &1 operands not compatible.
Cause Text:	The operands of comparison operator &1 are not compatible: • Numeric operands are compatible with any other numeric operands and with character and graphic operands. • Character operands are compatible with operands that are character, graphic, date, time, timestamp, or numeric. • Date, time, and timestamp operands are compatible with character and graphic operands or with another operand of the same type. • Graphic operands are compatible with graphic, character, date, time, timestamp, or numeric operands. • Binary operands are compatible only with binary operands. • Operands that are user-defined types can only be compared to operands that are the same exact type. • DataLink and XML operands cannot be compared.
Recovery Text:	Check the data types of all operands to see if the data types are compatible. If all the operands of the SQL statement are correct and a view is being accessed, then check the data types of all the operands in the view definition. Correct the errors. Try the request again.
SQLCODE or SQLCODEs:	-401
SQLSTATE or SQLSTATEs:	42818

SQL0402	
Message Text:	&1 use not valid.
Cause Text:	An operand has been specified for the arithmetic function or operator &1 that is not valid. • User-defined types cannot be specified as operands of operators or scalar functions. User-defined types can only be specified with operators and within user-defined functions created specifically for that type. • The operand of DIGITS can be any numeric or numeric compatible type except floating-point. • The operand of INTEGER, SMALLINT, BIGINT, FLOAT, DOUBLE, and DOUBLE_PRECISION cannot be date, time, or timestamp. • The other functions or operators require numeric or numeric compatible operands.
Recovery Text:	Ensure all operands of function or operator &1 are valid. Correct the operands. Try the request again.
SQLCODE or SQLCODEs:	-402
SQLSTATE or SQLSTATEs:	42819

SQL0403	
Message Text:	Alias &1 in &2 created but table or view not found.
Cause Text:	The alias &1 was created in &2, but the referenced table or view, &3 in &4, could not be found.
Recovery Text:	The alias was created, but cannot be used until &3 in &4 is created.
SQLCODE or SQLCODEs:	+403
SQLSTATE or SQLSTATEs:	01522

SQL0404	
Message Text:	Value for column or variable &1 too long.
Cause Text:	An INSERT or UPDATE statement or a SET or VALUES INTO statement or a GET DIAGNOSTICS statement specifies a value that is longer than the maximum length string that can be stored in &1. The length of &1 is &2 and the length of the string is &3.
Recovery Text:	Reduce the length of the string from &3 to a maximum of &2 and try the request again.
SQLCODE or SQLCODEs:	-404
SQLSTATE or SQLSTATEs:	22001

SQL0405	
Message Text:	Numeric constant &1 out of range.

SQL0405	
Cause Text:	The numeric constant beginning &1 is out of range. • Decimal and integer constants may contain a maximum of 63 significant digits. The range allowed for a floating point literal is 2.2250738585072013 * 10**-308 to 1.7976931348623158 * 10**308. • In a SELECT or DECLARE CURSOR statement, 0 is not valid for the number of rows in the OPTIMIZE clause. • The INCREMENT BY value must be an integer value. • Identity column and sequence attribute constants must be values that can be assigned to the identity column or sequence. • The expansion factor for REC2XML must be greater than 0.0 and less than or equal to 6.0.
Recovery Text:	Ensure all numeric constants are within the range allowed for the data type and conform to the values allowed for their specific use. Try the request again.
SQLCODE or SQLCODEs:	-405
SQLSTATE or SQLSTATEs:	42820

SQL0406	
Message Text:	Conversion error on assignment to column &2.
Cause Text:	During an attempt to assign a value to column &2 with an INSERT, UPDATE, ALTER TABLE, or REFRESH TABLE statement, conversion error type &3 occurred. If precompiling, the error occurred when converting a numeric constant to the same attributes as column &2. A list of the error types follows: • 1 - Overflow. • 2 - Floating point overflow. • 3 - Floating point underflow. • 4 - Floating point conversion error. • 5 - Not an exact result. • 6 - Numeric data that is not valid. • 7 - Double-byte character set (DBCS) data that is not valid.
Recovery Text:	Change the statement so that the result value fits in column &2 and is valid, or create the table or view again, specifying a new type or length for column &2 so that the result value can be assigned.
SQLCODE or SQLCODEs:	-406
SQLSTATE or SQLSTATEs:	22003, 22023, 22504

SQL0407	
Message Text:	Null values not allowed in column or variable &5.

SQL0407	
Cause Text:	One of the following has occurred: • Column &5 is a target column in an UPDATE or INSERT statement for table &3 in &4. Either a null value was specified to be inserted or updated into this column or a value for the column was not specified in an INSERT statement and the column does not allow null values. The null value was specified in the relative entry number &6 in the VALUES list, select list, or SET clause. • Column &5 is specified in an ALTER statement for table &3 in &4. The attribute of column &5 can not be changed to NOT NULL because a null value exists in relative entry number &6 of the column. • Variable &5 is a target variable in an SQL procedure, function, or trigger. A null value was specified to be set into this variable using a SET or VALUES statement, but the variable does not allow null values. The null value was specified in relative entry number &6 in the SET or VALUES INTO clause. The null value was specified as either NULL, a host variable with an associated indicator variable that contains a negative value, a column containing a null value, or an expression that evaluated to NULL. If it is a host variable or column then the name is &7. The null value for a column may be disallowed by a CHECK constraint that was added implicitly to enforce the NOT NULL attribute of the column specified on the CREATE or ALTER of the column. The null value for a column may be disallowed by a CHECK constraint that was added implicitly to enforce the partitioning key attributes of the column specified for the base table on the CREATE TABLE or ALTER TABLE statement.
Recovery Text:	If this is an ALTER TABLE statement, change the existing null values in the column to a non-null value. Otherwise, change the value so the result is not null. If a host variable is specified, change the value in the related indicator variable to be greater than or equal to zero. Try the request again.
SQLCODE or SQLCODEs:	-407
SQLSTATE or SQLSTATEs:	23502

SQL0408	
Message Text:	Value for column or variable &1 not compatible.
Cause Text:	The data type of the source value is not compatible with the data type of the target column or variable &1. If the statement is INSERT or UPDATE, &1 is a column in table &2 in schema &3. • Any numeric type value can be assigned to any numeric, character, or graphic type. • Any character or graphic value can be assigned to any character, graphic, or numeric type. • Any binary value can be assigned to any binary type. • A date, time, or timestamp value can be assigned to any character or graphic type. • Any character or graphic value can be assigned to a date, time, or timestamp type. • A date value can be assigned to a date. • A time value can be assigned to a time. • A timestamp value can be assigned to a timestamp. • A value being assigned to a user-defined type must be promotable to the source type. • The DLVALUE function must be specified when assigning to a DataLink.
Recovery Text:	Change the column, host variable, constant, or function assigned to &1 to one that is compatible. Try the request again.
SQLCODE or SQLCODEs:	-408

SQL0408	
SQLSTATE or SQLSTATEs:	42821

SQL0410	
Message Text:	Floating point literal &1 not valid.
Cause Text:	The number of characters in the floating point constant &1 cannot exceed 24 excluding leading zeros. The number of digits in the first number cannot exceed 17 excluding leading zeros, and the number of digits in the second number cannot exceed 3.
Recovery Text:	Correct the indicated literal &1. Make certain that the floating point literal is in the correct form shown by the following examples: +1.2E+3, 15E1, 2.E5, 2.2e-1, +5.E+2, -.1e1. Try the request again.
SQLCODE or SQLCODEs:	-410
SQLSTATE or SQLSTATEs:	42820

SQL0412	
Message Text:	Subselect with more than one result column not valid.
Cause Text:	The subselect of a predicate must have one result column specified in its SELECT list when the other operand of the predicate is a single expression. The result of the subselect can be zero, one, or many rows to form a list, but it must have only one result column.
Recovery Text:	Change the number of items in the SELECT list of the subselect so only one result column is specified or change the other operand of the predicate to be a list of expressions.
SQLCODE or SQLCODEs:	-412
SQLSTATE or SQLSTATEs:	42823

SQL0414	
Message Text:	Operand not valid in LIKE predicate.
Cause Text:	Operand 1 of a LIKE predicate has a type of date, time, timestamp, or DataLink or is a user-defined type. Operands specified in LIKE predicates must be binary, character, graphic, or numeric. If the ESCAPE character is specified, operand 1 cannot be DBCS-only. If the operand is a column, the column name is &1.
Recovery Text:	Change operand 1 of the LIKE predicate to be a binary, character, graphic, or numeric type. Use a different predicate for date, time, or timestamp comparisons. Do not specify an ESCAPE character if operand 1 is DBCS-only. Try the request again.
SQLCODE or SQLCODEs:	-414
SQLSTATE or SQLSTATEs:	42824

SQL0415	
Message Text:	UNION, EXCEPT, or INTERSECT operands not compatible.

SQL0415	
Cause Text:	Column &2 is not compatible with the corresponding column in another subselect of the UNION, EXCEPT, or INTERSECT. The relative position of the value in the select list is &1. One of the following conditions exists: • One column is character, graphic, or numeric and the other is not character, graphic, or numeric. • One of the columns is date, time, or timestamp and the other is not character or the same type. • One column is binary, and the other is not binary. • If the column name is *N, the column is not a named column.
Recovery Text:	Change the columns of the operands of the UNION, EXCEPT, or INTERSECT clause so that they are compatible. A casting function can be used to make the columns compatible types. Try the request again.
SQLCODE or SQLCODEs:	-415
SQLSTATE or SQLSTATEs:	42825

SQL0417	
Message Text:	Combination of parameter markers not valid.
Cause Text:	The statement string specified as the object of a PREPARE statement contains a predicate or expression where parameter markers have been used as operands of the same operator. The following restrictions apply to the use of parameter markers: • Both the operands in a predicate cannot be parameter markers. For example, specifying predicates of the form: ? = ? or ? = (SELECT ? FROM x) are not valid. • Both the operands in a expression cannot be parameter markers. For example, specifying an expression of the form: ? + ? is not valid. • At least one of the operands in the BETWEEN predicate cannot be a parameter marker. For example, specifying the predicate of the form: ? BETWEEN ? and ? is not valid. • At least, one of the operands of the IN predicate must not be a parameter marker. For example, specifying the predicate of the form: ? IN (?, ?, ?) is not valid.
Recovery Text:	Correct the statement so that all operands of the predicate or expression are not parameter markers. A CAST specification can be used in most cases to assign attributes to a parameter marker. Try the request again.
SQLCODE or SQLCODEs:	-417
SQLSTATE or SQLSTATEs:	42609

SQL0418	
Message Text:	Use of parameter marker not valid.

SQL0418	
Cause Text:	Parameter markers and the RAISE_ERROR scalar function are not allowed: • As a value in a VALUES INTO statement. • As an operand of a concatenation operation. • As the operand of a scalar function. If the scalar function is VALUE, COALESCE, IFNULL, MIN, MAX, LAND, LOR, or XOR, then at least one of the arguments must not be a parameter marker. • As the left operand of the LIKE predicate. • As the operand of a unary minus. Parameter markers are also not allowed: • In the SELECT clause of the statement string to be prepared. • In an SQL statement in embedded SQL or in interactive SQL. • In an EXECUTE IMMEDIATE statement. • In a CREATE VIEW, CREATE TABLE, or ALTER TABLE statement. • In a statement processed by the RUNSQLSTM command. • In a blocked INSERT statement.
Recovery Text:	Ensure parameter markers and the RAISE_ERROR scalar function are only specified where they are allowed. A CAST specification can be used in many situations. Correct any errors. Try the request again.
SQLCODE or SQLCODEs:	-418
SQLSTATE or SQLSTATEs:	42610

SQL0419	
Message Text:	Negative scale not valid.
Cause Text:	A decimal division operation has produced a negative scale. To view the algorithm used to determine the scale for decimal division, refer to the DB2 UDB for iSeries SQL Reference topic in the Information Center, http://www.ibm.com/eserver/iseries/infocenter.
Recovery Text:	Change one of the operands to floating point by using the FLOAT scalar function. This will change the result of division to floating point. If a decimal result is desired, use the DECIMAL scalar function in the floating point result. If one of the operands is integer, small integer, or big integer, SQL has converted it to decimal prior to the division. The DECIMAL function can be used to explicitly convert the integer, small integer, or big integer to a precision that will not cause the division to produce a negative scale. Try the request again.
SQLCODE or SQLCODEs:	-419
SQLSTATE or SQLSTATEs:	42911

SQL0420	
Message Text:	Character in CAST argument not valid.
Cause Text:	A character in the argument for the CAST function was not correct.
Recovery Text:	Change the result data type to one that recognizes the characters in the CAST argument, or change the argument to contain a valid representation of a value for the result data type. Try the request again.
SQLCODE or SQLCODEs:	+420, -420

SQL0420	
SQLSTATE or SQLSTATEs:	01565, 22018

SQL0421	
Message Text:	Number of columns not consistent.
Cause Text:	The subselects of a UNION, INTERCEPT, or EXCEPT must have the same number of result columns. All rows specified for a multiple row insert must have the same number of values.
Recovery Text:	Correct the SQL statement so that the same number of columns are defined for each row. Try the request again.
SQLCODE or SQLCODEs:	-421
SQLSTATE or SQLSTATEs:	42826

SQL0423	
Message Text:	LOB locator &1 not valid.
Cause Text:	The value of locator &1 is not currently valid. The locator may have been freed by a previous FREE LOCATOR statement or a COMMIT or ROLLBACK.
Recovery Text:	Ensure that the locator value refers to an active locator that has not been freed because of a FREE LOCATOR, COMMIT, or ROLLBACK statement. A LOB value can be assigned to a locator variable by means of a SELECT INTO statement, a VALUES INTO or SET statement, or a FETCH statement.
SQLCODE or SQLCODEs:	-423
SQLSTATE or SQLSTATEs:	0F001

SQL0426	
Message Text:	Dynamic COMMIT is not valid for the application environment.
Cause Text:	An application using DRDA two-phase commit protocols has attempted to issue a dynamic COMMIT, or has called a stored procedure which performed a COMMIT ON RETURN.
Recovery Text:	Remove the dynamic COMMIT statement, or change the stored procedure definition to not use COMMIT ON RETURN.
SQLCODE or SQLCODEs:	-426
SQLSTATE or SQLSTATEs:	2D528

SQL0427	
Message Text:	Dynamic ROLLBACK is not valid for the application environment.
Cause Text:	An application using DRDA two-phase commit protocols has attempted to issue a dynamic ROLLBACK.
Recovery Text:	Remove the dynamic ROLLBACK statement.
SQLCODE or SQLCODEs:	-427

SQL0427	
SQLSTATE or SQLSTATEs:	2D529

SQL0428	
Message Text:	SQL statement cannot be run.
Cause Text:	The statement cannot be run in the current application state. A SET TRANSACTION, DISCONNECT, or SET SESSION AUTHORIZATION statement was encountered and a connection is not at a commit boundary. A SET SESSION AUTHORIZATION statement was encountered and one of the following has occurred: • The SYSTEM_USER is one of the system-supplied user profiles such as QSYS, QDFTOWN, or QSPL. • The current server is a local relational database and there is an active connection to a remote relational database. • A stored procedure, user-defined function, or trigger is running. • Resources are being held because a COMMIT HOLD or HOLD LOCATOR statement has been run. • The maximum number of ProfileHandles have been generated.
Recovery Text:	Do one of the following actions prior to running the SQL statement. Try the request again. • If the connection is not at a commit boundary, issue a COMMIT or ROLLBACK SQL statement. • If connections to remote relational databases are active, disconnect the remote connections. • If the SYSTEM_USER is one of the system-supplied user profiles, sign-on with a different user profile. • If resources are being held because a COMMIT HOLD has been run, issue a COMMIT or ROLLBACK SQL statement. • If resources are being held because a HOLD LOCATOR has been run, issue a ROLLBACK or FREE LOCATOR SQL statement. • If the maximum number of ProfileHandles have been generated, release some of the ProfileHandles that are no longer needed using the Release ProfileHandle (QSYRLSPH) program.
SQLCODE or SQLCODEs:	-428
SQLSTATE or SQLSTATEs:	25501

SQL0429	
Message Text:	The maximum number of concurrent LOB locators has been reached.
Cause Text:	The LOB locator could not be generated because there are already 16000000 valid locators for this process.
Recovery Text:	Use the FREE LOCATOR statement to free LOB locators.
SQLCODE or SQLCODEs:	-429
SQLSTATE or SQLSTATEs:	54028

SQL0432	
Message Text:	A parameter marker cannot have the user-defined type name &1.

SQL0432	
Cause Text:	A parameter marker in the statement has been determined as having the user-defined type &1 based on the context in which it is used. A parameter marker cannot have a user-defined type as its data type unless it is part of an assignment (VALUES clause of INSERT or SET clause of UPDATE) or it is being explicitly cast to a user-defined type using the CAST specification.
Recovery Text:	Use an explicit cast to the user-defined distinct type for the parameter marker or cast the columns that are user-defined types to their corresponding source data type.
SQLCODE or SQLCODEs:	-432
SQLSTATE or SQLSTATEs:	42841

SQL0433	
Message Text:	Significant data truncated during CAST to character.
Cause Text:	The length of the resulting character string is not large enough to hold the character representation of the value.
Recovery Text:	Change the result data type to a character string long enough to hold the result. Try the request again.
SQLCODE or SQLCODEs:	-433
SQLSTATE or SQLSTATEs:	22001

SQL0435	
Message Text:	SQLSTATE value &1 not valid.
Cause Text:	SQLSTATE value &1 specified in a handler or condition, in a SIGNAL or RESIGNAL statement, or in a RAISE_ERROR function is not valid. SQLSTATE values must have a length of 5 and must contain uppercase characters A-Z or numbers 0-9. The first two characters of the SQLSTATE value cannot be '00'. The SQLSTATE value in a RAISE_ERROR function cannot begin with '00', '01', or '02'. If the SQLSTATE value shown is '*N', an empty string or null value was passed for the SQLSTATE.
Recovery Text:	Change the SQLSTATE to one that is valid. Try the request again.
SQLCODE or SQLCODEs:	-435
SQLSTATE or SQLSTATEs:	428B3

SQL0440	
Message Text:	Routine &1 in &2 not found with specified parameters.
Cause Text:	A function or procedure with the specified name and compatible arguments was not found.
Recovery Text:	Specify the correct number and type of parameters on the CALL statement or function invocation. Try the request again.
SQLCODE or SQLCODEs:	-440
SQLSTATE or SQLSTATEs:	42884

SQL0441	
Message Text:	Clause or keyword &1 not valid where specified.
Cause Text:	One of the following errors has occurred: • AS LOCATOR is specified for a parameter in a procedure or as a parameter or in the returns clause of a function and the parameter is not defined as BLOB, CLOB, or DBCLOB. If the parameter is defined as BLOB, CLOB, or DBCLOB, a length, a CCSID value, or a FOR BIT DATA, FOR MIXED DATA, or FOR SBCS DATA clause was specified. • ALL or DISTINCT is specified in a function that is not a column function.
Recovery Text:	Specify the type as BLOB, CLOB, or DBCLOB or remove the AS LOCATOR clause. Remove the ALL or DISTINCT keyword from the function.
SQLCODE or SQLCODEs:	-441
SQLSTATE or SQLSTATEs:	42601

SQL0442	
Message Text:	Too many parameters for procedure &1 in &2 on CALL statement.
Cause Text:	Only 1024 parameters are allowed on the CALL statement. If the procedure is a REXX procedure, only 32766 bytes of data can be passed on the CALL statement.
Recovery Text:	Reduce the number of parameters specified to the maximum of 1024. If calling a REXX procedure, limit the total number of bytes of parameter data to be less than 32766. Try the request again.
SQLCODE or SQLCODEs:	-442
SQLSTATE or SQLSTATEs:	54023

SQL0443	
Message Text:	Trigger program or external routine detected an error.
Cause Text:	Either a trigger program, external procedure, or external function detected and returned an error to SQL. If the error occurred in a trigger program, the trigger was on table &4 in schema &5. If the error occurred in an external procedure or function, the external name is &4 in schema &5. The associated text is &6. If the error occurred in a trigger program, the associated text is the type of trigger program. If the error occurred in an external function, the associated text is the text of the error message returned from the external function.
Recovery Text:	Refer to the joblog for more information regarding the detected error. Correct the error and try the request again.
SQLCODE or SQLCODEs:	-443
SQLSTATE or SQLSTATEs:	38xxx, 38501

SQL0444	
Message Text:	External program &4 in &5 not found.
Cause Text:	An attempt was made to CALL procedure or invoke function &1 in &2. External program or service program &4 in schema &5 was not found.

SQL0444	
Recovery Text:	The external program or service program associated with the procedure or function cannot be found. Ensure that an object exists with the name specified on the DECLARE PROCEDURE, CREATE PROCEDURE, or CREATE FUNCTION statement. If no name was specified, ensure that an object with a name which matches the procedure or function name specified exists. If a program name was specified, a program object must exist. If an entry point name was specified, then a service program object must exist. Try the request again.
SQLCODE or SQLCODEs:	-444
SQLSTATE or SQLSTATEs:	42724

SQL0445	
Message Text:	Value of parameter &4 in procedure &1 in &2 too long.
Cause Text:	Parameter &4, which is declared as OUT or INOUT, contains a value that is longer than the maximum length string that can be stored in host variable &8. Parameter &4 is being returned from procedure &1 in &2 to host variable &8. Trailing blanks are not included in the length of the string for character values. Trailing hex zeros are not included in the length of the string for binary values. The length of the parameter is &6 and the length of the host variable is &7.
Recovery Text:	Increase the length of the host variable from &7 to &6. Try the request again.
SQLCODE or SQLCODEs:	+445
SQLSTATE or SQLSTATEs:	01004

SQL0446	
Message Text:	Conversion error in assignment of argument &2.
Cause Text:	During an attempt to assign input argument number &1 on a CALL statement to the corresponding parameter for the call, error type &3 occurred. A list of the error types follows: • 1 - Overflow. • 2 - Floating point overflow. • 3 - Floating point underflow. • 4 - Floating point conversion error. • 5 - Not an exact result. • 6 - Numeric data that is not valid. • 7 - Double-byte character set (DBCS) data that is not valid. The parameter name is &2.
Recovery Text:	Change the attribute declaration for parameter &1 in the CREATE, ALTER, or DECLARE PROCEDURE statement to match the attributes of argument &1 in the CALL statement or correct the data that is not valid. Try the request again.
SQLCODE or SQLCODEs:	-446
SQLSTATE or SQLSTATEs:	22003

SQL0448	
Message Text:	Too many parameters or result sets for routine &1 in &2.

SQL0448	
Cause Text:	One of the following limits has been exceeded: • 1024 parameters in a DECLARE PROCEDURE, CREATE PROCEDURE, or ALTER PROCEDURE statement. The actual number may be less and depends on the language. • 1023 parameters if GENERAL WITH NULLS is specified. • 90 parameters if PARAMETER STYLE SQL or DB2SQL is specified. • 1024 parameters for an SQL procedure. • 90 parameters in a CREATE FUNCTION statement. • 124 parameters and return values in a CREATE FUNCTION(Table) statement. • 255 for languages other than C and C++. • 32767 result sets.
Recovery Text:	Reduce the number of parameters defined to the maximum or change the value for the number of result sets to be less than or equal to 32767. Try the request again.
SQLCODE or SQLCODEs:	-448
SQLSTATE or SQLSTATEs:	54023

SQL0449	
Message Text:	External program name for routine &1 in &2 not valid.
Cause Text:	The external program name specified on a DECLARE PROCEDURE, CREATE PROCEDURE, or CREATE FUNCTION statement is not valid for the routine or the language specified. • The external program name for a procedure or function must be of the form 'library-name/program-name' or 'library-name/program-name(entry-point-name)'. • The external program name for a JAVA procedure or function must be 'class-name!method-name' or 'class-name.method-name'. • The external program name for a REXX procedure must be 'library-name/source-file-name(member-name)'.
Recovery Text:	Specify the correct form of the external program name. Try the request again.
SQLCODE or SQLCODEs:	-449
SQLSTATE or SQLSTATEs:	42878

SQL0451	
Message Text:	Attributes of parameter &1 not valid for procedure or function &3 in &4.

SQL0451	
Cause Text:	The data type, length, or value of parameter &1 is not valid for the language specified for procedure or function &3 in &4. The parameter name is &2. A list of conditions for the parameters follows: • For C: NUMERIC is not a valid data type. • For PL/I: NUMERIC, BIGINT, GRAPHIC, VARGRAPHIC, CLOB, BLOB, and DBCLOB are not valid data types. • For COBOL: precision for DECIMAL or NUMERIC cannot be greater than 18. FLOAT, GRAPHIC, BIGINT, VARGRAPHIC, CLOB, BLOB, and DBCLOB are not valid data types. • For REXX: SMALLINT, BIGINT, NUMERIC, Unicode graphic, CLOB, BLOB, and DBCLOB are not valid data types. FLOAT is not valid if the precision is from 1 to 24. Precision for DECIMAL cannot be greater than 63. • For RPG: FLOAT, BIGINT, VARCHAR, VARBINARY, GRAPHIC, VARGRAPHIC, CLOB, BLOB, DBCLOB and ROWID are not valid data types. • For CL: BIGINT, NUMERIC, VARCHAR, VARBINARY, FLOAT, GRAPHIC, VARGRAPHIC, CLOB, BLOB, and DBCLOB are not valid data types. GENERAL WITH NULLS cannot be specified for CL. • DataLinks are not valid data types for parameters unless the routine is an SQL procedure or function. • LOBs with AS LOCATOR are not valid data types for parameters of an SQL procedure or function.
Recovery Text:	Correct the data type or precision specified for the parameter on the DECLARE PROCEDURE, CREATE PROCEDURE, ALTER PROCEDURE, or CREATE FUNCTION statement. Try the request again.
SQLCODE or SQLCODEs:	-451
SQLSTATE or SQLSTATEs:	42815

SQL0452	
Message Text:	Unable to access a file that is referred to by a file reference variable.
Cause Text:	The file referred to by the file reference variable (host variable &1) could not be accessed because of reason code &2. The reason codes and their meanings are: • 1 - The file name or path has a format that is not valid. • 2 - The length of the file name is greater than the maximum allowed length. • 3 - The file option is not valid. • 4 - The file or directory cannot be found. • 5 - A file already exists with the same name as that specified for a file that has the NEW option.
Recovery Text:	Do one of the following: • If the reason code is 1, correct the format of the filename or path and then try the request again. • If the reason code is 2, correct the file name and then try the request again. • If the reason code is 3, correct the file option and then try the request again. • If the reason code is 4, specify SQL_FILE_CREATE for the file option, or verify the directory exists, and then try the request again. • If the reason code is 5, specify SQL_FILE_OVERWRITE or SQL_FILE_APPEND and then try the request again.
SQLCODE or SQLCODEs:	-452

SQL0452	
SQLSTATE or SQLSTATEs:	428A1

SQL0453	
Message Text:	Return type for function &1 in &2 not compatible with CAST TO type.
Cause Text:	The data types specified in the RETURNS clause for function &1 in &2 are not valid. The CAST TO and CAST FROM data types are not compatible.
Recovery Text:	Correct the data type specified in the RETURNS clause for the function. Try the request again.
SQLCODE or SQLCODEs:	-453
SQLSTATE or SQLSTATEs:	42880

SQL0454	
Message Text:	Routine &1 in &2 already exists.
Cause Text:	One of the following has occurred: • Procedure &1 with the same number of parameters already exists in schema &2. Procedures in a schema cannot have the same name and number of parameters. • Function &1 with the same signature already exists in schema &2. All functions in the same schema must have a unique signature. The database uses the name of the function and the number and data types of the arguments to determine the signature for the function.
Recovery Text:	Change the routine name or the parameters or drop the existing routine. Try the request again.
SQLCODE or SQLCODEs:	-454
SQLSTATE or SQLSTATEs:	42723

SQL0455	
Message Text:	Schema &2 for specific name not same as routine schema &3.
Cause Text:	The specific name schema &2 specified on a CREATE PROCEDURE, DECLARE PROCEDURE, ALTER PROCEDURE, or CREATE FUNCTION statement is not the same as schema &3 for procedure or function &1.
Recovery Text:	Specify the same schema for the specific name as for the procedure or function name.
SQLCODE or SQLCODEs:	-455
SQLSTATE or SQLSTATEs:	42882

SQL0456	
Message Text:	Specific name &3 in &2 already exists.
Cause Text:	An attempt was made to create a function or procedure &1 in &2 with specific name &3, but specific name &3 already exists in the schema. All routines (functions and procedures) in the same schema must have unique specific names.

SQL0456	
Recovery Text:	Specify a SPECIFIC NAME that does not exist or do not specify a SPECIFIC NAME and a unique name will be generated for you. Otherwise, delete the existing routine. Try the request again.
SQLCODE or SQLCODEs:	-456
SQLSTATE or SQLSTATEs:	42710

SQL0457	
Message Text:	Name &1 in &2 not allowed for function.
Cause Text:	Function &1 in &2 cannot be created or cannot be used as the source function name. Either the function name is a reserved word or the schema is specified as QSYS, QSYS2, SYSIBM, or QTEMP. Functions cannot be created in QSYS, QSYS2, SYSIBM, or QTEMP.
Recovery Text:	Change the name of the function to one that is not reserved or specify a different schema. Try the request again.
SQLCODE or SQLCODEs:	-457
SQLSTATE or SQLSTATEs:	42939

SQL0458	
Message Text:	Function &1 in &2 not found with matching signature.
Cause Text:	Function &1 is specified in schema &2. The name of the function and the number and data types of the parameters make up the function signature. A function with a matching signature was not found.
Recovery Text:	Ensure that the function name specified exists and that the number and data types of the parameters match those in the function definition. Try the request again.
SQLCODE or SQLCODEs:	-458
SQLSTATE or SQLSTATEs:	42883

SQL0460	
Message Text:	Truncation of data may have occurred for ALTER TABLE of &1 in &2.
Cause Text:	Table &1 in &2 has been altered. The length of column &3 has been reduced and data may have been truncated.
Recovery Text:	No recovery is necessary.
SQLCODE or SQLCODEs:	+460
SQLSTATE or SQLSTATEs:	01593

SQL0461	
Message Text:	CAST from &1 to &2 not supported.
Cause Text:	CAST is not supported from data type &1 to data type &2. If the CAST is from date, time, or timestamp to character, the length of the character result is too small.

SQL0461	
Recovery Text:	Change the result data type or length to one that is supported for the CAST function or change the expression to have a data type that can be cast to &2. Try the request again.
SQLCODE or SQLCODEs:	-461
SQLSTATE or SQLSTATEs:	42846

SQL0462	
Message Text:	Procedure or user defined function &1 in &2 returned a warning SQLSTATE.
Cause Text:	An SQLSTATE of the form 01Hxx was returned by the procedure or user defined function &1 in &2 (with specific name &3), along with message text &4.
Recovery Text:	The user must understand the meaning of the warning. See your database administrator, or the author of the UDF or procedure.
SQLCODE or SQLCODEs:	+462
SQLSTATE or SQLSTATEs:	01Hxx

SQL0464	
Message Text:	Procedure &1 returned &3 result sets, which exceeds the defined limit of &4.
Cause Text:	The stored procedure &1 in &2 was successful. However, the stored procedure exceeded the defined limit on the number of result sets that can be returned. Only &4 result sets are returned to the SQL program that issued the SQL CALL statement. The possible causes are: • The number of result sets is greater than the maximum specified when the procedure was created. • The stored procedure is unable to return &3 result sets due to DRDA limitations imposed by the client.
Recovery Text:	The SQL statement is successful. The SQLWARN9 field of the SQLCA is set to 'Z'.
SQLCODE or SQLCODEs:	+464
SQLSTATE or SQLSTATEs:	0100E

SQL0466	
Message Text:	&3 result sets are available from procedure &1 in &2.
Cause Text:	Procedure &1 in &2 was called and has returned one or more result sets.
Recovery Text:	None.
SQLCODE or SQLCODEs:	+466
SQLSTATE or SQLSTATEs:	0100C

SQL0467	
Message Text:	Another result set exists for procedure &1 in &2.
Cause Text:	A result set was closed for &1 in &2. Another result set exists for the stored procedure. A maximum of &3 result sets are possible for this procedure.

SQL0467	
Recovery Text:	None.
SQLCODE or SQLCODEs:	+467
SQLSTATE or SQLSTATEs:	0100D

SQL0469	
Message Text:	IN, OUT, or INOUT not valid for parameter &4 in procedure &1 in &2.
Cause Text:	The IN, INOUT, or OUT attribute specified for parameter &4 when the procedure was defined is not valid. The parameter name is &5. One of the following errors occurred: • The attribute is not consistent with the parameter on the CALL statement. If the parameter was declared INOUT or OUT, the parameter on the CALL statement must be specified as a host variable. • The attribute was specified as INOUT or OUT and REXX was specified as the language. The attribute must be IN if REXX is specified.
Recovery Text:	Either change the attribute of the parameter on the DECLARE PROCEDURE, CREATE PROCEDURE, or ALTER PROCEDURE statement or change the parameter. Try the request again.
SQLCODE or SQLCODEs:	-469
SQLSTATE or SQLSTATEs:	42886

SQL0470	
Message Text:	Null values not allowed for parameter &4 in procedure &1 in &2.
Cause Text:	Null values are not allowed on the CALL statement for procedure &1 because the procedure was declared with GENERAL specified. The null value was specified as either the NULL keyword or a host variable with an associated indicator variable that contains a negative value. The parameter number is &4 and the parameter name is &5.
Recovery Text:	Specify PARAMETER STYLE SQL, DB2SQL, or GENERAL WITH NULLS on the CREATE PROCEDURE or DECLARE PROCEDURE statement or pass a value other than the null value on the CALL statement. Try the request again.
SQLCODE or SQLCODEs:	-470
SQLSTATE or SQLSTATEs:	39004

SQL0473	
Message Text:	Name &1 cannot be used for user-defined type.
Cause Text:	Name &1 specified for a user-defined type is the same as a system predefined type or is a function name that is reserved. Function names that cannot be used include CAST, DATAPARTITIONNAME, DATAPARTITIONNUM, DBPARTITIONNAME, DBPARTITIONNUM, EXTRACT, NODENAME, NODENUMBER, PARTITION, POSITION, RRN, STRIP, SUBSTRING, and TRIM.
Recovery Text:	Change the name for the user-defined type. Try the request again.
SQLCODE or SQLCODEs:	-473

SQL0473	
SQLSTATE or SQLSTATEs:	42918

SQL0475	
Message Text:	RETURNS data type for function &3 in &4 not valid.
Cause Text:	The data type specified for the RETURNS clause or the CAST FROM clause for function &3 in schema &4 is not appropriate for the data type returned from the sourced function or the value specified on the RETURN statement in the SQL function body. The data type specified in the RETURNS clause is &1 and the data type returned from the sourced function or SQL function is &2.
Recovery Text:	Correct the data types specified or specify another sourced function. Try the request again.
SQLCODE or SQLCODEs:	-475
SQLSTATE or SQLSTATEs:	42866

SQL0476	
Message Text:	Routine &1 in &2 not unique.
Cause Text:	Function or procedure &1 in &2 was specified, not by signature or specific name, and more than one specific instance of the routine was found.
Recovery Text:	Request the routine either by its specific name, or by its signature (function or procedure name with parameter types). Try the request again.
SQLCODE or SQLCODEs:	-476
SQLSTATE or SQLSTATEs:	42725

SQL0478	
Message Text:	Object &1 in &2 of type &3 cannot be dropped.
Cause Text:	The base object &1 cannot be dropped because another object depends on it. The dependent object &4 in &5 is of type &6. • If base object type is *N, the object being dropped is a function. When dropping a function, the dependent object is sourced on the base object. For objects other than functions, it may be that the dependency is indirect. That is, the named object is dependent on another object which is dependent on the object being dropped. • If the base object is a table and if there are other tables with triggers or foreign key constraints dependent on the base table, then the RESTRICT clause of the DROP statement will prevent the base table from being dropped. • If the base object is *LIB and if there are tables, views, functions, procedures, distinct types, aliases, or triggers in the schema, then the RESTRICT clause of the DROP statement will prevent the schema from being dropped. A trigger can be defined in one schema on a table that exists in a different schema. • If the base object is *SQLUDT and if there are tables, views, indexes, functions, procedures, sequences, or triggers dependent on the distinct type, then the RESTRICT clause of the DROP statement will prevent the base distinct type from being dropped.
Recovery Text:	Drop the dependent objects first. Try the request again.
SQLCODE or SQLCODEs:	-478

SQL0478	
SQLSTATE or SQLSTATEs:	42893

SQL0483	
Message Text:	Parameters for function &1 in &2 not same as sourced function.
Cause Text:	The number of parameters specified for function &1 in &2 is not the same as the number of parameters specified for the sourced function.
Recovery Text:	Specify the correct number of parameters for the function or specify another sourced function. If not qualified, ensure the correct sourced function exists in the current path. Try the request again.
SQLCODE or SQLCODEs:	-483
SQLSTATE or SQLSTATEs:	42885

SQL0484	
Message Text:	Routine &1 in &2 already exists.
Cause Text:	An attempt was made to create routine &1 in &2, but &1 already exists. All procedures and functions in the same schema must have unique specific names.
Recovery Text:	Specify a SPECIFIC name that does not exist or do not specify a SPECIFIC name and a unique name will be generated for you. Otherwise, delete the existing routine. Try the request again.
SQLCODE or SQLCODEs:	-484
SQLSTATE or SQLSTATEs:	42733

SQL0487	
Message Text:	SQL statements not allowed.
Cause Text:	One of the following errors has occurred: • A procedure was called or a function was invoked that was created with NO SQL specified as the data access attribute. A routine created with NO SQL, or any subsequent routines, cannot contain SQL statements and cannot invoke a routine that has the CONTAINS SQL DATA, READS SQL DATA, or MODIFIES SQL DATA attribute. • NO SQL cannot be specified when creating an SQL procedure or function. • A trigger containing SQL statements was activated. If the error occurred in a procedure or function, the routine name is &1 and the specific name is &2.
Recovery Text:	Ensure routines created as NO SQL only invoke routines that do not contain SQL statements. Do not specify NO SQL for an SQL procedure or function.
SQLCODE or SQLCODEs:	-487
SQLSTATE or SQLSTATEs:	38001

SQL0490	
Message Text:	Numeric value &1 not valid.

SQL0490	
Cause Text:	&1 was specified, but it is not in the valid range of values for its use. The valid range of values are &2 through &3.
Recovery Text:	Change the value and try the request again.
SQLCODE or SQLCODEs:	-490
SQLSTATE or SQLSTATEs:	428B7

SQL0491	
Message Text:	Clause not correct for CREATE FUNCTION or CREATE PROCEDURE.
Cause Text:	A clause for CREATE PROCEDURE or CREATE FUNCTION is missing or not allowed. The reason code is &1. Reason codes and their meanings are: • 1 - For CREATE FUNCTION, the RETURNS clause is required. • 2 - For CREATE FUNCTION, the DBINFO, FINAL CALL, and SCRATCHPAD clauses cannot be specified if the parameter style is SQL or GENERAL. For CREATE PROCEDURE, DBINFO cannot be specified unless the parameter style is DB2SQL. • 3 - For CREATE FUNCTION, GENERAL can only be specified if the EXTERNAL NAME specifies a service program. • 4 - For CREATE PROCEDURE and CREATE FUNCTION, parameter style JAVA or DB2GENERAL can only be specified for LANGUAGE JAVA. • 5 - For CREATE FUNCTION, a table function cannot have parameter style SQL, JAVA, GENERAL, or GENERAL WITH NULLS. • 6 - For CREATE PROCEDURE and CREATE FUNCTION, PROGRAM TYPE MAIN is not allowed for JAVA or REXX. For CREATE FUNCTION, PROGRAM TYPE SUB is only allowed for service programs. • 7 - For CREATE FUNCTION, a table function must specify DISALLOW PARALLEL. • 8 - The CARDINALITY clause is only allowed for table functions.
Recovery Text:	Add the missing clause or remove the clause that is not allowed. Try the request again.
SQLCODE or SQLCODEs:	-491
SQLSTATE or SQLSTATEs:	42601

SQL0492	
Message Text:	Data type for function &1 in &2 not valid for source type.
Cause Text:	The data type specified for parameter &3 for function &1 in library &2 is not valid for the corresponding type of the SOURCE function.
Recovery Text:	Correct the data type specified for parameter &3 or specify another sourced function. Try the request again.
SQLCODE or SQLCODEs:	-492
SQLSTATE or SQLSTATEs:	42879

SQL0501	
Message Text:	Cursor &1 not open.

SQL0501	
Cause Text:	The cursor &1 was specified in a FETCH or CLOSE statement, but the cursor is not open. Cursor &1 has one of the following conditions: • Cursor &1 was never opened. • The cursor &1 was opened in another program or another call of this program and the program was created with CLOSQLCSR(*ENDPGM). • The cursor &1 was opened in another module or another call of this module and the module was created with CLOSQLCSR(*ENDMOD). • The cursor &1 was opened in another call of this program and programs which have run SQL statements have ended and the program was created with CLOSQLCSR(*ENDSQL). • The cursor &1 was opened in another call of this module and the activation group ended between calls. The module was created with CLOSQLCSR(*ENDACTGRP). • The cursor was closed by a CLOSE, COMMIT, or ROLLBACK statement. • The cursor &1 was opened under a transaction which is different than the current transaction.
Recovery Text:	Do one of the following and precompile again: • Make certain that cursor &1 is opened in the same program or module call prior to using the cursor in an FETCH or CLOSE statement. • Specify either CLOSQLCSR(*ENDSQL), CLOSQLCSR(*ENDJOB), or CLOSQLCSR(*ENDACTGRP) when precompiling the application. • If the cursor was closed by a COMMIT or ROLLBACK, specify HOLD on the COMMIT or ROLLBACK statement to preserve any open cursors, prepared statements, and locks on tables.
SQLCODE or SQLCODEs:	-501
SQLSTATE or SQLSTATEs:	24501

SQL0502	
Message Text:	Cursor &1 already open.
Cause Text:	The cursor specified in an OPEN statement is already open for this call of the program.
Recovery Text:	Close cursor &1 and then try the OPEN statement again or change the name of the cursor, and then precompile the program again.
SQLCODE or SQLCODEs:	-502
SQLSTATE or SQLSTATEs:	24502

SQL0503	
Message Text:	Column &3 cannot be updated.
Cause Text:	An UPDATE statement attempted to update column &3 in table or view &1 in &2. The column cannot be updated because it was not specified in the FOR UPDATE OF clause in the associated DECLARE CURSOR statement.
Recovery Text:	Add column &3 to the FOR UPDATE OF clause in the related DECLARE CURSOR statement. Precompile the program again.
SQLCODE or SQLCODEs:	-503
SQLSTATE or SQLSTATEs:	42912

SQL0504	
Message Text:	Cursor &1 not declared.
Cause Text:	Cursor &1 is not declared in the program before it is referred to. A cursor must be declared before being referred to in other statements. All cursors used in the SET RESULT SETS statement must be declared WITH RETURN if any cursors in the program are declared WITH RETURN or WITHOUT RETURN.
Recovery Text:	Verify that the application program is complete and has no spelling errors in the cursor declarations. Make certain the declaration for a cursor is in an application program before it is referred to by other statements. If any cursors are declared WITH RETURN or WITHOUT RETURN, make sure all cursors used by the SET RESULT SETS statement are declared WITH RETURN. Precompile the program again.
SQLCODE or SQLCODEs:	-504
SQLSTATE or SQLSTATEs:	34000

SQL0507	
Message Text:	Cursor &1 not open.
Cause Text:	Cursor &1 was specified in an UPDATE or DELETE statement, but the cursor is not open. Cursor &1 has one of the following conditions: • Cursor &1 was never opened. • The cursor &1 was opened in another program or another call of this program and the program was created with CLOSQLCSR(*ENDPGM). • The cursor &1 was opened in another module or another call of this module and the module was created with CLOSQLCSR(*ENDMOD). • The cursor &1 was opened in another call of this program and programs which have run SQL statements have ended and the program was created with CLOSQLCSR(*ENDSQL). • The cursor &1 was opened in another call of this module and the activation group was ended between calls. The program was created with CLOSQLCSR(*ENDACTGRP). • The cursor was closed by a CLOSE, COMMIT, or ROLLBACK statement.
Recovery Text:	Do one of the following and precompile again: • Make certain that cursor &1 is opened in the same program or module call prior to using the cursor in an UPDATE or DELETE statement. • Specify either CLOSQLCSR(*ENDSQL), CLOSQLCSR(*ENDJOB), or CLOSQLCSR(*ENDACTGRP) when precompiling the application. • If the cursor was closed by a COMMIT or ROLLBACK, specify HOLD on the COMMIT or ROLLBACK statement to preserve any open cursors, prepared statements, and locks on tables.
SQLCODE or SQLCODEs:	-507
SQLSTATE or SQLSTATEs:	24501

SQL0508	
Message Text:	Cursor &1 not positioned on locked row.
Cause Text:	An UPDATE or DELETE statement with a WHERE CURRENT OF &1 was attempted, but the cursor is not positioned on a row or is positioned on a row, but the row is not locked because a COMMIT HOLD or ROLLBACK HOLD statement released the lock on the row. A FETCH statement must be issued to position the cursor on a row and lock the row.

SQL0508	
Recovery Text:	Issue a FETCH statement to position the cursor on a row and lock the row; then, try the request again.
SQLCODE or SQLCODEs:	-508
SQLSTATE or SQLSTATEs:	24504

SQL0509	
Message Text:	Table &2 in &3 not same as table in cursor &1.
Cause Text:	An UPDATE or DELETE statement with a WHERE CURRENT OF &1 specified the table &2 in &3, but cursor &1 refers to a different table. The table specified in the UPDATE or DELETE statement and the table referred to by cursor &1 must be the same.
Recovery Text:	Change the specified table name to match the table specified in the cursor &1 and precompile the program again.
SQLCODE or SQLCODEs:	-509
SQLSTATE or SQLSTATEs:	42827

SQL0510	
Message Text:	Cursor &1 for table &2 read-only.
Cause Text:	An UPDATE or DELETE statement with a WHERE CURRENT OF clause was specified, but the cursor is read only. Either the cursor &1 is read-only or the view &2 in schema &3 is read-only. A view or logical file is read-only if one or more of the following conditions are true: • The view contains a DISTINCT keyword, GROUP BY clause, HAVING clause, a column function, or UNION, EXCEPT, or INTERSECT in the outer-most subselect. • The view or logical file contains a join function. • The view contains a subquery that refers to the same table as the table of the outer-most subselect. • All the columns of the view are expressions, scalar functions, or constants. • All the columns of the logical file are input only. A cursor is read-only if one or more of the following conditions is true: • The DECLARE CURSOR statement specified an ORDER BY clause but did not specify a FOR UPDATE OF clause. • The DECLARE CURSOR statement specified a FOR READ ONLY clause. • The DECLARE CURSOR statement specified the SCROLL keyword without DYNAMIC. • The cursor referred to a read-only view or logical file in the select list. • The subselect specified in the DECLARE CURSOR statement contains any of the above restrictions that would make a view read only. • A second INSTEAD OF trigger is required for the update or delete on a dependent view.

SQL0510	
Recovery Text:	Do one of the following and precompile the program again: • If the DECLARE CURSOR statement specified an ORDER BY clause but not a FOR UPDATE OF clause, add a FOR UPDATE OF clause. • If the DECLARE CURSOR statement specified a FOR READ ONLY clause, remove the FOR READ ONLY clause. • If the DECLARE CURSOR statement specified the SCROLL keyword, specify DYNAMIC SCROLL. • If the referred to view or logical file is read only, remove the UPDATE or DELETE statement. • If the DECLARE CURSOR statement contains any conditions that make the cursor read only, remove the UPDATE or DELETE statement.
SQLCODE or SQLCODEs:	-510
SQLSTATE or SQLSTATEs:	42828

SQL0511	
Message Text:	FOR UPDATE clause not valid.
Cause Text:	The FOR UPDATE clause cannot be used for cursor &1 because the result table is read only. The result table is read only if: • The first SELECT clause of the statement includes the DISTINCT keyword, a column function, a GROUP BY clause, a HAVING clause, or a UNION, EXCEPT, or INTERSECT operator. • The first FROM clause of the SELECT statement identifies more than one table, more than one view, or a read-only view.
Recovery Text:	Cursor &1 cannot be updated. Remove the FOR UPDATE clause.
SQLCODE or SQLCODEs:	-511
SQLSTATE or SQLSTATEs:	42829

SQL0513	
Message Text:	Alias &1 in &2 cannot reference another alias.
Cause Text:	Alias &1 in &2 can only reference a table or a view. It cannot reference another alias.
Recovery Text:	Change the referenced name and try the request again.
SQLCODE or SQLCODEs:	-513
SQLSTATE or SQLSTATEs:	42924

SQL0514	
Message Text:	Prepared statement &2 not found.

SQL0514	
Cause Text:	An attempt was made to open cursor &1 which referred to prepared statement &2. Statement &2 has one of the following conditions: • The statement has never been prepared. • The statement was prepared in another program or another call of this program and the program was created with CLOSQLCSR(*ENDPGM). • The statement was prepared in another module or another call of this module and the module was created with CLOSQLCSR(*ENDMOD). • The statement was prepared in another call of this program and programs which have run SQL statements have ended and the program was created with CLOSQLCSR(*ENDSQL). • The statement was prepared in another call of this module and the activation group ended between calls. The module was created with CLOSQLCSR(*ENDACTGRP). • A COMMIT or ROLLBACK statement has destroyed all the prepared statements.
Recovery Text:	Do one of the following and precompile again: • Prepare the statement &2 (PREPARE statement) before attempting to open cursor &1. • Make certain that &2 has been prepared in the same program or module call prior to attempting to open cursor &1 or specify either CLOSQLCSR(*ENDSQL) CLOSQLCSR(*ENDJOB), or CLOSQLCSR(*ENDACTGRP) when precompiling the application. • If the statement was deleted by a COMMIT or ROLLBACK, specify HOLD on the COMMIT or ROLLBACK statement to preserve any open cursors, prepared statements, and locks on tables.
SQLCODE or SQLCODEs:	-514
SQLSTATE or SQLSTATEs:	26501

SQL0516	
Message Text:	Prepared statement &2 not found.
Cause Text:	&2 is not a valid prepared statement. The statement has one of the following conditions: • The statement has never been prepared. • The statement was prepared in another program or another call of this program and the program was created with CLOSQLCSR(*ENDPGM). • The statement was prepared in another module or another call of this module and the module was created with CLOSQLCSR(*ENDMOD). • The statement was prepared in another call of this program and programs which have run SQL statements have ended and the program was created with CLOSQLCSR(*ENDSQL). • The statement was prepared in another call of this module and the activation group ended between calls. The module was created with CLOSQLCSR(*ENDACTGRP). • A COMMIT or ROLLBACK statement has destroyed all the prepared statements.
Recovery Text:	Do one of the following and precompile again: • Make certain that &1 has been prepared in the same program or module call prior to using the DESCRIBE statement or specify either CLOSQLCSR(*ENDSQL), CLOSQLCSR(*ENDJOB), or CLOSQLCSR(*ENDACTGRP) when precompiling the application. • If the statement was deleted by a COMMIT or ROLLBACK, specify HOLD on the COMMIT or ROLLBACK statement to preserve any open cursors, prepared statements, and locks on tables.

SQL0516	
SQLCODE or SQLCODEs:	-516
SQLSTATE or SQLSTATEs:	26501

SQL0517	
Message Text:	Prepared statement &2 not SELECT statement.
Cause Text:	An attempt was made to open cursor &1, which refers to statement &2. Statement &2 is a valid prepared statement, but it is not a SELECT statement. OPEN can only refer to prepared SELECT statements.
Recovery Text:	Change the OPEN statement to refer to a prepared SELECT statement, or prepare statement &2 using a valid SELECT statement and then try the open.
SQLCODE or SQLCODEs:	-517
SQLSTATE or SQLSTATEs:	07005

SQL0518	
Message Text:	Prepared statement &1 not found.
Cause Text:	An EXECUTE statement referred to the statement &1. &1 is not a valid prepared statement. The statement has one of the following conditions: • The statement has never been prepared. • The statement identifies a prepared SELECT or DECLARE PROCEDURE statement. • The statement was in error at prepare or bind time. • The statement was prepared in another program or another call of this program and the program was created with CLOSQLCSR(*ENDPGM). • The statement was prepared in another module or another call of this module and the module was created with CLOSQLCSR(*ENDMOD). • The statement was prepared in another call of this program and programs which have run SQL statements have ended and the program was created with CLOSQLCSR(*ENDSQL). • The statement was prepared in another call of this module and the activation group ended between calls. The module was created with CLOSQLCSR(*ENDACTGRP). • The prepared statement was destroyed by a COMMIT or ROLLBACK statement. This only occurs when connected to a database other than DB2 UDB for iSeries.
Recovery Text:	Do one of the following: • If &1 identifies a prepared SELECT or DECLARE PROCEDURE statement, a different prepared statement must be named in the EXECUTE statement. • If the statement had errors, correct the errors and either prepare the statement again or precompile the program again. • Make certain that &1 has been prepared in the same program or module call prior to using the EXECUTE statement or specify either CLOSQLCSR(*ENDSQL), CLOSQLCSR(*ENDJOB), or CLOSQLCSR(*ENDACTGRP) when precompiling the application. • If the statement was deleted by a COMMIT or ROLLBACK, either specify PREPARE WITH HOLD or bind the program with the correct option to preserve prepared statements.
SQLCODE or SQLCODEs:	-518

SQL0518	
SQLSTATE or SQLSTATEs:	07003

SQL0519	
Message Text:	Prepared statement &2 in use.
Cause Text:	The application program has attempted to prepare statement &2. This statement is the SELECT statement for cursor &1 that is currently open.
Recovery Text:	Change the statement name in the PREPARE statement or correct the logic of the application program so that it closes cursor &1 before attempting the PREPARE statement again.
SQLCODE or SQLCODEs:	-519
SQLSTATE or SQLSTATEs:	24506

SQL0520	
Message Text:	Cannot UPDATE or DELETE on cursor &1.
Cause Text:	Cursor &1 is blocking records. An UPDATE or DELETE WHERE CURRENT OF cursor &1 was attempted but cannot be run because blocking was being used for the cursor.
Recovery Text:	Do not block records for cursor &1 if UPDATE or DELETE statements will be used against it.
SQLCODE or SQLCODEs:	-520
SQLSTATE or SQLSTATEs:	42828

SQL0525	
Message Text:	Statement not valid on application server.
Cause Text:	An attempt was made to run statement number &4 of package &2 in schema &3. Either the statement is in error, or the statement is not supported by the application server. The section number corresponding to the statement is &1.
Recovery Text:	Correct the statement which is in error and verify that it is a valid SQL statement on the application server. If the statement is a multiple-row FETCH, specify a single-row FETCH. If the statement is a multiple-row INSERT, either specify a VALUES clause with one or more rows to insert, or specify a subselect. If the statement is SET TRANSACTION, remove it from the application or prevent it from being issued when the Distributed Relational Database Architecture (DRDA) connection is used. Try the request again.
SQLCODE or SQLCODEs:	-525
SQLSTATE or SQLSTATEs:	51015

SQL0526	
Message Text:	Statement not valid for &1 TEMPORARY TABLE &2 in &3.

SQL0526	
Cause Text:	A &1 TEMPORARY TABLE &2 cannot be specified on ALTER TABLE, COMMENT ON, CREATE TRIGGER, GRANT, LABEL ON, LOCK TABLE, RENAME, or REVOKE statements. A constraint cannot be specified for a &1 TEMPORARY TABLE and a &1 TEMPORARY TABLE cannot be partitioned.
Recovery Text:	Specify a valid table on the statement or remove the constraint or partitioning clause. Try the request again.
SQLCODE or SQLCODEs:	-526
SQLSTATE or SQLSTATEs:	42995

SQL0527	
Message Text:	ALWCPYDTA(*NO) specified but temporary result required for &1.
Cause Text:	The Allow Copy Data (ALWCPYDTA) parameter was specified on the precompiler command or the STRSQL command with a value of *NO. This value indicates that the queries should always use data directly retrieved from the database, so that the data always reflects the current values. Certain types of SQL queries can not be resolved without making a copy of the data. Examples would be queries using the keywords DISTINCT or UNION. The query being run is a query which requires a copy of the data.
Recovery Text:	Specify a different value for the ALWCPYDTA parameter or change the SQL statement so that it can be evaluated without using a temporary result.
SQLCODE or SQLCODEs:	-527
SQLSTATE or SQLSTATEs:	42874

SQL0530	
Message Text:	Operation not allowed by referential constraint &1 in &2.
Cause Text:	If this is an INSERT or UPDATE statement, the value is not valid for the foreign key because it does not have a matching value in the parent key. If this is a DELETE statement affected by a SET DEFAULT delete rule, the default value is not valid for the same reason. If this is an ALTER TABLE statement, the result of the operation would violate the constraint &1. Constraint &1 in &2 for table &3 in &4 requires that any non-null value of the foreign key have a matching value in the parent key.
Recovery Text:	To conform to the constraint rule, you must either: • change the INSERT or UPDATE value to match a value in the parent key • insert a row in the parent file that matches the foreign key values being inserted or updated • insert a row in the parent file that matches the foreign key default values of the dependent rows. Otherwise, you must drop the referential constraint.
SQLCODE or SQLCODEs:	-530
SQLSTATE or SQLSTATEs:	23503

SQL0531	
Message Text:	Update prevented by referential constraint &1 in &2.

SQL0531	
Cause Text:	Constraint &1 in &2 identifies the table being updated as the parent table in a relationship with dependent table &3 in &4, with an update rule of RESTRICT or NO ACTION. The update of a parent key is prevented when there are rows in the dependent table with matching values.
Recovery Text:	In order to perform this update, you must either drop the constraint, or delete the rows in the dependent table that are dependent on this row.
SQLCODE or SQLCODEs:	-531
SQLSTATE or SQLSTATEs:	23504, 23001

SQL0532	
Message Text:	Delete prevented by referential constraint &1 in &2.
Cause Text:	Constraint &1 in &2 identifies the table being modified as the parent table in a relationship with dependent table &3 in &4, with a delete rule of RESTRICT or NO ACTION. The deletion of a row is prevented when there are rows in the dependent table with matching values.
Recovery Text:	In order to delete the row, you must either drop the constraint, or delete the rows in the dependent table that are dependent on this row.
SQLCODE or SQLCODEs:	-532
SQLSTATE or SQLSTATEs:	23001, 23504

SQL0536	
Message Text:	Delete not allowed because table &1 in &2 in subquery can be affected.
Cause Text:	The delete operation can not be performed because table &1 in &2, which is referenced in a subquery, may be affected by the operation. &1 in &2 is a dependent table in a referential constraint that has a delete rule of CASCADE, SET NULL, or SET DEFAULT.
Recovery Text:	Change the statement so that the subquery does not refer to a dependent table, or drop the constraint that defines the relationship between the two tables.
SQLCODE or SQLCODEs:	-536
SQLSTATE or SQLSTATEs:	42914

SQL0537	
Message Text:	Duplicate column name &1 in definition of key.
Cause Text:	Column &1 is specified more than once in the list of columns for a primary, unique, foreign, or partitioning key. Columns may only be specified once in the definition of a key.
Recovery Text:	Remove the duplicate column from the list of columns for the key. Try the request again.
SQLCODE or SQLCODEs:	-537
SQLSTATE or SQLSTATEs:	42709

SQL0538	
Message Text:	The FOREIGN key in constraint &1 in &2 not same as the parent key.
Cause Text:	The FOREIGN key in constraint &1 in &2 is not the same as the parent key of table &3 in &4. The FOREIGN key must have the same number of columns as the parent key and the data type and attributes of the FOREIGN key must be identical to the data type and attributes of the corresponding column of the parent key.
Recovery Text:	Correct the statement so that the description of the FOREIGN key conforms to that of the parent key of the specified table.
SQLCODE or SQLCODEs:	-538
SQLSTATE or SQLSTATEs:	42830

SQL0539	
Message Text:	Table &1 in &2 does not have a primary or unique key.
Cause Text:	Table &1 in &2 was specified either as the parent table in a referential constraint, or as the table from which to drop the primary or unique key in an ALTER TABLE statement. When no referencing column list is specified in a referential constraint, an attempt is made to use the primary key of the parent table. Table &1 has no primary key defined.
Recovery Text:	Correct the statement so that a referencing column list is specified in the FOREIGN KEY clause that matches the FOREIGN KEY column list, or define a primary key for the table being used as a parent. If this is an attempt to drop a primary or unique key, no recovery is necessary.
SQLCODE or SQLCODEs:	-539
SQLSTATE or SQLSTATEs:	42888

SQL0541	
Message Text:	Duplicate UNIQUE constraint exists for table &1 in &2.
Cause Text:	An attempt was made to add UNIQUE constraint &3 in &4. Table &1 in &2 already has a UNIQUE constraint that is a duplicate of the constraint being added. A UNIQUE constraint is a duplicate if the columns in the constraint are the same as the columns in another UNIQUE constraint, even if the columns are not in the same order. Constraint &1 cannot be added.
Recovery Text:	The constraint is already in effect. To change the name of the UNIQUE constraint, drop the duplicate constraint and try the request again.
SQLCODE or SQLCODEs:	-541
SQLSTATE or SQLSTATEs:	42891

SQL0543	
Message Text:	Constraint &1 conflicts with SET NULL or SET DEFAULT rule.
Cause Text:	Constraint &1 is a CHECK constraint that conflicts with an existing referential constraint that has either a SET NULL or a SET DEFAULT rule.
Recovery Text:	Change the CHECK constraint so it does not conflict with the referential constraint rule, or drop the referential constraint.

SQL0543	
SQLCODE or SQLCODEs:	-543
SQLSTATE or SQLSTATEs:	23511

SQL0544	
Message Text:	CHECK constraint &1 cannot be added.
Cause Text:	Existing data in the table violates the CHECK constraint rule in constraint &1. The constraint cannot be added.
Recovery Text:	Change the data in the table so that it follows the constraint specified in &1. Try the request again.
SQLCODE or SQLCODEs:	-544
SQLSTATE or SQLSTATEs:	23512

SQL0545	
Message Text:	INSERT or UPDATE not allowed by CHECK constraint.
Cause Text:	The value being inserted or updated does not meet the criteria of CHECK constraint &1. The operation is not allowed.
Recovery Text:	Change the values being inserted or updated so that the CHECK constraint is met. Otherwise, drop the CHECK constraint &1.
SQLCODE or SQLCODEs:	-545
SQLSTATE or SQLSTATEs:	23513

SQL0546	
Message Text:	CHECK condition of constraint &1 not valid.
Cause Text:	The CHECK condition of constraint &1 is not valid for one of the following reasons: • a column level CHECK condition refers to some other column in the table • the CHECK condition refers to a column that is not in this table • the CHECK condition refers to a special register • the CHECK condition uses a column function (such as AVG or COUNT) or a user-defined function • the CHECK condition contains a subselect • the CHECK condition uses the NODENAME, DBPARTITIONNAME, DATAPARTITIONNAME, or DATAPARTITIONNUM scalar function • the CHECK condition uses an expression involving LOBs.
Recovery Text:	Correct the error. Try the request again.
SQLCODE or SQLCODEs:	-546
SQLSTATE or SQLSTATEs:	42621

SQL0551	
Message Text:	Not authorized to object &1 in &2 type *&3.
Cause Text:	An operation was attempted on object &1 in &2 type *&3. This operation cannot be performed without the required authority.
Recovery Text:	Obtain the required authority from either the security officer or the object owner. If you are not authorized to a logical file, obtain the authority to the based-on files of the logical file. Try the operation again.
SQLCODE or SQLCODEs:	+551, -551
SQLSTATE or SQLSTATEs:	01548, 42501

SQL0552	
Message Text:	Not authorized to &1.

SQL0552	
Cause Text:	The operation cannot be performed without the required authority. See the following for the authority required: • CREATE TABLE requires the *USE authority to the Create Physical File (CRTPF) command. • CREATE VIEW or CREATE INDEX requires the *USE authority to the Create Logical File (CRTLF) command. • CREATE ALIAS requires the *USE authority to the Create DDM File (CRTDDMF) command. • CREATE SCHEMA requires the *USE authority to the Create Library (CRTLIB) command. If WITH DATA DICTIONARY is specified then the *USE authority to the Create Data Dictionary (CRTDTADCT) command is also required. • ALTER TABLE requires the *USE authority to the Add Physical File Constraint (ADDPFCST) command in order to add constraints, and the *USE authority to the Remove Physical File Constraint (RMVPFCST) command in order to drop constraints. • CREATE PROCEDURE or CREATE FUNCTION requires the *OBJOPR and *ADD authority to the catalog table SYSROUTINES in QSYS2. • ALTER PROCEDURE requires the *OBJOPR and *ALTER authority to the catalog table SYSROUTINES in QSYS2. • DROP PROCEDURE or DROP FUNCTION requires the *OBJOPR and *DLT authority to the catalog table SYSPARMS in QSYS2. • COMMENT ON PROCEDURE or COMMENT ON FUNCTION requires the *OBJOPR, *READ, and *UPD authority to the catalog table SYSROUTINES in QSYS2. • CREATE TYPE requires the *OBJOPR and *ADD authority to the catalog table SYSTYPES in QSYS2. • DROP TYPE requires the *OBJOPR and *DLT authority to the catalog table SYSTYPES in QSYS2. • COMMENT ON TYPE requires the *OBJOPR, *READ, and *UPD authority to the catalog table SYSTYPES in QSYS2. • CREATE TRIGGER requires the *USE authority to the Add Physical File Trigger (ADDPFTRG) command. • DROP TRIGGER requires the *USE authority to the Remove Physical File Trigger (RMVPFTRG) command. • COMMENT ON TRIGGER requires the *OBJOPR, *READ, and *UPD authority to the catalog table SYSTRIGGERS in QSYS2. • CREATE SEQUENCE requires the *USE authority to the Create Data Area (CRTDTAARA) command. • DROP SEQUENCE requires the *USE authority to the Delete Data Area (DLTDTAARA) command. • ALTER SEQUENCE requires the *USE authority to the Retrieve Data Area (RTVDTAARA) and Change Data Area (CRTDTAARA) commands. • COMMENT ON SEQUENCE requires the *USE authority to the Change Data Area (CHGDTAARA) command. • SET SESSION AUTHORIZATION requires that the authorization ID associated with the statement has *ALLOBJ special authority. • SET CURRENT DEGREE requires that the authorization ID associated with the statement has *JOBCTL special authority.
Recovery Text:	Obtain authority from the security officer and try the operation again.
SQLCODE or SQLCODEs:	+552, -552
SQLSTATE or SQLSTATEs:	01542, 42502

SQL0557	
Message Text:	Privilege not valid for table or view &1 in &2.
Cause Text:	The specified privilege is not valid for one of the following reasons: • An INDEX privilege is valid for tables and physical files but not for views. An index cannot be created on a view. • The specified privilege is not valid because table or view &1 in &2 does not have that capability. For example: DELETE, INSERT, and UPDATE privileges are not valid for a read-only view.
Recovery Text:	Specify a table or view that has the correct capability or remove the privilege that is not valid from the SQL statement.
SQLCODE or SQLCODEs:	-557
SQLSTATE or SQLSTATEs:	42852

SQL0567	
Message Text:	Authorization name &1 is not allowed.
Cause Text:	The authorization name cannot be a system-supplied user profile such as QSYS, QDFTOWN, or QSPL.
Recovery Text:	Change the name and try the request again.
SQLCODE or SQLCODEs:	-567
SQLSTATE or SQLSTATEs:	28000

SQL0569	
Message Text:	Not all requested privileges revoked from object &1 in &2 type *&3.
Cause Text:	A REVOKE operation was attempted on object &1 in &2 type *&3, but the privilege was not revoked. Either you do not have the specified privilege to object &1, you do not have *OBJMGT authority to object &1, or you tried to revoke the privilege from someone who does not currently have that privilege. All valid requested privileges were revoked.
Recovery Text:	If revoking a privilege from someone who does not currently have that privilege, then no action is required. If you do not have the privilege, change the REVOKE statement to specify valid privileges.
SQLCODE or SQLCODEs:	+569
SQLSTATE or SQLSTATEs:	01006

SQL0570	
Message Text:	Not all requested privileges to object &1 in &2 type *&3 granted.
Cause Text:	A GRANT operation was attempted on object &1 in &2 type *&3 but the privilege was not granted. Either you do not have all of the privileges to be granted or you are attempting to perform a GRANT statement using WITH GRANT OPTION but are not the object owner, do not have *OBJMGT authority to the object, or do not have *ALLOBJ special authority. All valid requested privileges were granted.
Recovery Text:	Obtain the required authority from either the security officer or the object owner. Try the operation again.

SQL0570	
SQLCODE or SQLCODEs:	+570
SQLSTATE or SQLSTATEs:	01007

SQL0573	
Message Text:	Table &1 in &2 does not have a matching parent key.
Cause Text:	A referencing column list was specified in the FOREIGN KEY clause for constraint &3 in &4. The parent table &1 in &2 does not have a matching PRIMARY or UNIQUE key. The constraint cannot be added.
Recovery Text:	Do one of the following and try the request again: • Specify a table in the FOREIGN KEY clause that has a PRIMARY or UNIQUE key that matches the referencing column list. • Change the referencing column list to match the definition of the PRIMARY or UNIQUE key defined on the parent table.
SQLCODE or SQLCODEs:	-573
SQLSTATE or SQLSTATEs:	42890

SQL0574	
Message Text:	Column or sequence attribute is not valid.
Cause Text:	Either the DEFAULT value or identity attribute value for column &3 in &1 in &2 is not valid, a sequence attribute value for sequence &1 in &2 is not valid, or an assigned value for cycle column &3 is not valid. For a DEFAULT value, the incorrect value may either be specified in this statement or it may already be defined for the column and is not compatible with the attributes specified on the ALTER TABLE statement. The values must conform to the following rules: • The DEFAULT value must be compatible with the data type of the column. A floating-point constant can only be a default value for a floating-point column. • The DEFAULT value must not be too long for the column. • If the column is defined as a date, time, or timestamp the DEFAULT value must be a valid string representation of that type. • If the DEFAULT value is defined as the value of the USER special register, the column must be defined as a CHAR or VARCHAR and the length attribute must be greater than or equal to 18. • The CCSID of the DEFAULT value must be compatible with the CCSID of the column. • The DEFAULT value for a column that is a user-defined type must either be promotable to the source type or must be cast to the user-defined type using the cast function for the type. • A DEFAULT value cannot be specified for a DataLink column. For an identity column or a sequence, the values specified for the START WITH, INCREMENT BY, MINVALUE, MAXVALUE, and RESTART WITH options must have a scale of zero. For a recursive common table expression cycle column name, the assignment value must be a character string with a length of one.
Recovery Text:	Change the value to one that is valid for the column or sequence. Try the request again.
SQLCODE or SQLCODEs:	-574
SQLSTATE or SQLSTATEs:	42894

SQL0577	
Message Text:	Modifying SQL data not permitted.
Cause Text:	One of the following errors has occurred: • A procedure was called or a function was invoked that was created with READS SQL DATA or CONTAINS SQL DATA specified as the data access attribute. A procedure or function created with READS SQL DATA or CONTAINS SQL DATA, or any procedure or function that is called by the procedure or function, cannot change data and cannot call a procedure or function that has the MODIFIES SQL DATA attribute. • READS SQL DATA and CONTAINS SQL DATA cannot be specified on the CREATE PROCEDURE, ALTER PROCEDURE, or CREATE FUNCTION statements for an SQL procedure or function if the routine body contains statements that change data. • A BEFORE trigger was activated that contains statements that change data. Statements that change data include INSERT, UPDATE, DELETE, REFRESH TABLE, GRANT, REVOKE, LABEL, COMMENT, and any CREATE, DROP or ALTER statements. If the error occurred in a procedure or function, the routine name is &1 and the specific name is &2.
Recovery Text:	Ensure procedures or functions created with READS SQL DATA or CONTAINS SQL DATA do not call procedures or functions that change SQL data. Specify MODIFIES SQL DATA when creating functions that change SQL data.
SQLCODE or SQLCODEs:	-577
SQLSTATE or SQLSTATEs:	2F002, 38002, 42985

SQL0578	
Message Text:	RETURN statement not executed for SQL function &1 in &2.
Cause Text:	During the execution of SQL function &1 in &2, the end of the routine body was reached without executing a RETURN statement.
Recovery Text:	Add a RETURN statement to the end of the function routine body. Try the request again.
SQLCODE or SQLCODEs:	-578
SQLSTATE or SQLSTATEs:	2F005

SQL0579	
Message Text:	Reading SQL data not permitted.
Cause Text:	One of the following errors has occurred: • A procedure was called or a function was invoked that was created with CONTAINS SQL DATA as the data access attribute. A procedure or function created with CONTAINS SQL DATA or any procedure or function that is called by the procedure or function, cannot read data and cannot call a procedure or function that has the READS SQL DATA attribute. • CONTAINS SQL DATA cannot be specified on the CREATE PROCEDURE, ALTER PROCEDURE, or CREATE FUNCTION statements for an SQL procedure or function if the routine body contains statements that read data. • A trigger containing SQL statements was activated. If the error occurred in a procedure or function, the routine name is &1 and the specific name is &2.
Recovery Text:	Ensure procedures and functions created with CONTAINS SQL DATA do not call procedures or functions that read SQL data.

SQL0579	
SQLCODE or SQLCODEs:	-579
SQLSTATE or SQLSTATEs:	2F004, 38004, 42985

SQL0580	
Message Text:	At least one result in CASE expression must be not NULL.
Cause Text:	The NULL value, a parameter marker, or the RAISE_ERROR scalar function is specified for all results in a CASE expression. At least one result in a CASE expression must be a value other than NULL, a parameter marker, or the RAISE_ERROR scalar function.
Recovery Text:	Change the CASE expression to have at least one result expression following a THEN or ELSE keyword to be some value other than NULL, a parameter marker, or the RAISE_ERROR scalar function. Try the request again.
SQLCODE or SQLCODEs:	-580
SQLSTATE or SQLSTATEs:	42625

SQL0581	
Message Text:	The results in a CASE expression are not compatible.
Cause Text:	The expressions specified as the result values of a CASE expression are not compatible.
Recovery Text:	Change the results in the CASE expression to values that are compatible. Try the request again.
SQLCODE or SQLCODEs:	-581
SQLSTATE or SQLSTATEs:	42804

SQL0583	
Message Text:	Use of function &1 in &2 not valid.
Cause Text:	Function &1 in &2 cannot be invoked where specified because it is defined to be not deterministic or contains an external action. Functions that are not deterministic cannot be specified in a GROUP BY clause or in a JOIN clause. Functions that are not deterministic or contain an external action cannot be specified in a PARTITION BY clause or an ORDER BY clause for an OLAP function. The RAISE_ERROR function cannot be specified in a GROUP BY or HAVING clause.
Recovery Text:	Remove the function. Try the request again.
SQLCODE or SQLCODEs:	-583
SQLSTATE or SQLSTATEs:	42845

SQL0585	
Message Text:	Schema &1 used incorrectly in the specified path.

SQL0585	
Cause Text:	An incorrect path was specified on the SET PATH or the SET OPTION SQLPATH statement. One of the following errors has occurred: • &1 is specified more than once in the path. • *LIBL is not the only value specified in the path. • If the schema name is *N, one of USER, CURRENT PATH, or SYSTEM PATH is specified more than one time in the path.
Recovery Text:	Specify a valid path. Try the request again.
SQLCODE or SQLCODEs:	-585
SQLSTATE or SQLSTATEs:	42732

SQL0590	
Message Text:	Name &1 specified in procedure or function &2 not unique.
Cause Text:	&1 is specified as a parameter, SQL variable, condition, or label in SQL procedure or function &2. The name is not unique.
Recovery Text:	Change the name so that it is unique.
SQLCODE or SQLCODEs:	-590
SQLSTATE or SQLSTATEs:	42734

SQL0595	
Message Text:	Commit level *&1 escalated to *&2 lock.
Cause Text:	*&1 was specified for the commit level, but *&1 was not used. The base tables were locked *&2 to satisfy the request for commitment level of *&1. If a ROLLBACK HOLD statement is requested, the cursor will remain in the same position. *&1 was not used for one of the following reasons: • A GROUP BY clause, HAVING clause, or a column function was specified in the statement. • A DISTINCT keyword was specified in the statement. • A UNION keyword was specified in the statement. • A join was specified in the statement, but not all of the files are journaled to the same journal. • The repeatable read commit level is implemented by DB2 UDB for iSeries by locking the table.
Recovery Text:	If escalation of commit level is not desired, change the statement or the requested commit level. If a lock level of share-no-update (*SHRNUP) was granted, but is not acceptable, specify *CHG or *NONE for the commit level.
SQLCODE or SQLCODEs:	+595
SQLSTATE or SQLSTATEs:	01526

SQL0596	
Message Text:	Error occurred during DISCONNECT of relational database &1.

SQL0596	
Cause Text:	An error occurred during DISCONNECT of relational database &1. However, this did not prevent the successful disconnect of relational database &1. Refer to the previous messages for a description of the error.
Recovery Text:	None required.
SQLCODE or SQLCODEs:	+596
SQLSTATE or SQLSTATEs:	01002

SQL0601	
Message Text:	&1 in &2 type &3 already exists.
Cause Text:	An attempt was made to create &1 in &2 or to rename a table, view, alias, or index to &1, but &1 already exists. All tables, views, aliases, indexes, SQL packages, sequences, constraints, triggers, and user-defined types in the same schema must have unique names. • If &1 is a temporary table, it cannot be replaced unless the WITH REPLACE clause is specified. • If the schema name is *N, this is a CREATE SCHEMA statement. If this is a CREATE TABLE or ALTER TABLE statement and the type is *N, &1 is a constraint.
Recovery Text:	Change &1 to a name that does not exist, or delete, move, or rename the existing object. If this is a temporary table, use the WITH REPLACE clause. If creating an SQL package, specify REPLACE(*YES) on CRTSQLPKG. Try the request again.
SQLCODE or SQLCODEs:	-601
SQLSTATE or SQLSTATEs:	42710

SQL0602	
Message Text:	More than 120 columns specified for CREATE INDEX.
Cause Text:	Only 120 columns are allowed in the CREATE INDEX statement.
Recovery Text:	Reduce the number of column names in the column list to the maximum of 120 names. Try the request again.
SQLCODE or SQLCODEs:	-602
SQLSTATE or SQLSTATEs:	54008

SQL0603	
Message Text:	Unique index cannot be created because of duplicate keys.
Cause Text:	An attempt was made to create unique index &1 in &2 or add unique constraint &1 in &2. The operation cannot be performed because the rows in table &3 in &4 contain one or more duplicate values in the columns used to create the index.

SQL0603	
Recovery Text:	Do one of the following and try the request again: • Remove the UNIQUE attribute from the CREATE INDEX statement. • Remove the UNIQUE constraint from the ALTER TABLE statement. • Change the data in the related table so that all key values are unique. • Specify UNIQUE WHERE NOT NULL on the CREATE INDEX statement if the duplicate keys contain nulls. The uniqueness restriction would not apply when the key value contains nulls. For information on what rows contain the duplicate key values, see the previously listed messages in the job log.
SQLCODE or SQLCODEs:	-603
SQLSTATE or SQLSTATEs:	23515

SQL0604	
Message Text:	Attributes not valid.
Cause Text:	One of the following contains a length, precision, scale, or an ALLOCATE attribute that is not valid. • A column in the CREATE TABLE, ALTER TABLE, or DECLARE GLOBAL TEMPORARY TABLE statement. • A parameter or an SQL variable in the DECLARE PROCEDURE, CREATE PROCEDURE, ALTER PROCEDURE, CREATE FUNCTION, or CREATE TRIGGER statement. • The CAST scalar function. • A CREATE TYPE source data type. The definition is not valid for one of the following reasons: • If a DECIMAL or NUMERIC data type is specified, precision must be from 1 through 63 and the scale must be between 0 and the precision. • If CHARACTER or BINARY is specified, the length must be from 1 through 32766 for a parameter or for a column that does not allow null values or from 1 through 32765 for a column that allows null values. • If VARCHAR or VARBINARY is specified, the length must be from 1 through 32740 for a parameter or for a column that does not allow null values or from 1 through 32739 for a column that allows null values. • If the FOR MIXED DATA clause or a mixed CCSID is specified, the length cannot be less than 4. • If GRAPHIC is specified, the length must be from 1 through 16383 for a parameter or for a column that does not allow null values and from 1 through 16382 for a column that allows null values. The length specified is the number of DBCS characters. • If VARGRAPHIC is specified, the length must be from 1 through 16370 for a parameter or for a column that does not allow null values or from 1 through 16369 for a column that allows null values. The length specified is the number of DBCS characters. • If BLOB or CLOB is specified, the length must be from 1 through 2G (2048 M or 2,097,152 K or 2,147,483,647 bytes). If DBCLOB is specified, the length must be from 1 through 1073741823. • If DATALINK is specified, the length must be from 1 through 32717. • If VARCHAR, VARBINARY, CLOB, BLOB, DBCLOB, DATALINK, or VARGRAPHIC is specified the ALLOCATE attribute must be less than or equal to both the length attribute specified and 32766.
Recovery Text:	Correct the length, precision, scale, or ALLOCATE attribute. Try the request again.
SQLCODE or SQLCODEs:	-604

SQL0604	
SQLSTATE or SQLSTATEs:	42611

SQL0607	
Message Text:	Operation not allowed on system table &1 in &2.
Cause Text:	The table or view &1 in &2 is a catalog or system table. Catalogs, system tables and system triggers cannot be changed or locked by the user.
Recovery Text:	Change the SQL statement to refer to a table that is not a system table. Try the request again.
SQLCODE or SQLCODEs:	-607
SQLSTATE or SQLSTATEs:	42832

SQL0612	
Message Text:	&1 is a duplicate column name.
Cause Text:	One of the following errors has occurred: • Column &1 is specified more than once on a CREATE TABLE or CREATE VIEW statement. Column names and system column names must be unique in a table or view. • Column &1 is specified in the ADD clause of an ALTER TABLE statement. Column &1 already exists in the table. • Column &1 is specified more than once in the ALTER, DROP, or ADD clauses of an ALTER TABLE statement. • Column &1 is specified more than once in the column list of a common table expression or in the correlation clause for a table or derived table. • Column &1 is specified more than once in the column list of an UPDATE trigger. • Column &1 is specified more than once in the RETURNS TABLE clause of a CREATE FUNCTION statement. The return column names for a user defined table function must be unique. • Column &1 is specified more than once in the CYCLE column list of the recursive common table expression.
Recovery Text:	Do one of the following and try the request again: • Specify unique names for each of the columns. • Remove the column from all but one clause of a single ALTER TABLE statement. Multiple statements can be specified, if required.
SQLCODE or SQLCODEs:	-612
SQLSTATE or SQLSTATEs:	42711

SQL0613	
Message Text:	Key is too long or contains too many columns.

SQL0613	
Cause Text:	One of the following has occurred: • The PRIMARY or UNIQUE key cannot be created for constraint &1 in &2. • The partitioning key for the table is too long. Either more than 120 columns were specified in a key or the sum of the lengths of the columns specified in the key exceeds the maximum of 32768 bytes. If the list contains null capable columns then an additional byte is required for the length of each null capable column. If the list contains variable length columns, then the 2-byte length of the columns is included in the total length. The key length may be too long if an ICU (UNICODE) table is being used. The internal expansion of the key can be up to six times the user specified key length due to translation. See previous messages for details.
Recovery Text:	Remove some of the columns from the key or change the length attributes of the columns so that the total length of the key does not exceed the maximum. Try the request again.
SQLCODE or SQLCODEs:	-613
SQLSTATE or SQLSTATEs:	54008

SQL0614	
Message Text:	Length of columns for CREATE INDEX too long.
Cause Text:	The sum of the lengths of the columns specified in a CREATE INDEX exceeds the maximum of 32768 bytes. If the list contains null capable columns then an additional byte is required for each null capable column. If the index contains variable length columns, then the 2-byte length of the columns is included in the total length. The key length may be too long if an ICU (UNICODE) table is being used. The internal expansion of the key can be up to six times the user specified key length due to translation. See previous messages for details.
Recovery Text:	Reduce the length by deleting some of the columns from the CREATE INDEX column list. Try the request again.
SQLCODE or SQLCODEs:	-614
SQLSTATE or SQLSTATEs:	54008

SQL0615	
Message Text:	Object &1 in &2 type *&3 not dropped. It is in use.
Cause Text:	Object &1 in &2 type *&3 was not dropped because it is already being used by the same application process. If the object is a table, it may be in use by an open cursor. If the object is an SQL package, the package may currently be running.
Recovery Text:	If the object is a table, the cursor must be closed. If the object is an SQL package, the SQL package cannot drop itself. Try the drop request again.
SQLCODE or SQLCODEs:	-615
SQLSTATE or SQLSTATEs:	55006

SQL0616	
Message Text:	&1 in &2 type &3 cannot be dropped with RESTRICT.
Cause Text:	An attempt was made to drop &1 in &2 with the RESTRICT option. &1 cannot be dropped because a view, a constraint, or an index is dependent on it.

SQL0616	
Recovery Text:	Specify CASCADE on the ALTER TABLE statement to drop &1 and the views, constraints, or indexes that are dependent on it. Try the request again.
SQLCODE or SQLCODEs:	-616
SQLSTATE or SQLSTATEs:	42893

SQL0624	
Message Text:	Table &1 in &2 already has a primary key.
Cause Text:	An attempt was made to add a primary or unique key to table &1 in &2. Either a primary key is already defined for this table, or the table has an access path which does not match the key being added. A table can only have one primary key. The constraint cannot be added.
Recovery Text:	Drop the primary key currently defined on the table or add the constraint as a UNIQUE constraint. If the table's access path does not match, make sure that the number of columns in the constraint match the number of columns in the access path. Try the request again.
SQLCODE or SQLCODEs:	-624
SQLSTATE or SQLSTATEs:	42889

SQL0628	
Message Text:	Clauses not valid in same definition.
Cause Text:	Clauses specified to define the attributes of a column, a sourced function, or a trigger are not valid. One of the following has occurred: • More than one of the clauses FOR BIT DATA, FOR SBCS DATA, FOR MIXED DATA, or CCSID was specified for a column definition. • READ PERMISSION FS and WRITE PERMISSION BLOCKED were specified for a DataLink column. • READ PERMISSION DB and WRITE PERMISSION FS were specified for a DataLink column. • WRITE PERMISSION FS and the ON UNLINK clause were specified for a DataLink column. • INCLUDING or EXCLUDING COLUMN DEFAULTS and USING TYPE DEFAULTS were specified for a table. • A clause was specified that is not valid when creating a sourced function. • FOR EACH STATEMENT is specified for a BEFORE or INSTEAD OF trigger or is specified with MODE DB2ROW. • MODE DB2SQL is specified for a BEFORE trigger and the trigger contains a reference to the trigger table. • A WHEN clause or an UPDATE column list is specified for an INSTEAD OF trigger. • A correlation name with OLD ROW or NEW ROW and a SET OPTION for date or time option were specified. • The option specified on SET OPTION DBGVIEW is not consistent with the value specified for the DEBUG MODE.
Recovery Text:	Change or remove a clause so that the definition is valid. Try the request again.
SQLCODE or SQLCODEs:	-628

SQL0628	
SQLSTATE or SQLSTATEs:	42613

SQL0629	
Message Text:	SET NULL not allowed for referential constraint &1 in &2.
Cause Text:	SET NULL was specified in the REFERENCES clause for referential constraint &1. None of the columns in the foreign key allows null values.
Recovery Text:	Create the table so that at least one of the columns in the foreign key allows null values or specify a different default action on the ON DELETE clause. Try the request again.
SQLCODE or SQLCODEs:	-629
SQLSTATE or SQLSTATEs:	42834

SQL0631	
Message Text:	Foreign key for referential constraint &1 in &2 too long.
Cause Text:	The FOREIGN key cannot be created for constraint &1 in &2. Either more than 120 columns were specified in a FOREIGN KEY clause or the sum of the lengths of the columns specified in the key exceeds the maximum of 32768 bytes. If the list contains null capable columns then an additional byte is required for the length of each null capable column. If the list contains variable length columns, then the 2-byte length of the columns is included in the total length.
Recovery Text:	Remove some of the columns from the FOREIGN KEY clause. Try the request again.
SQLCODE or SQLCODEs:	-631
SQLSTATE or SQLSTATEs:	54008

SQL0636	
Message Text:	Ranges specified for partition &1 not valid.
Cause Text:	The ranges specified for the partitioning keys are not valid for one of the following reasons: • Partitions are not specified in the correct order. • ENDING value for a partition is less than the STARTING value. • Values specified in the EVERY clause are not valid for the range. • MINVALUE or MAXVALUE was specified for a start or end value and the statement contained an EVERY clause. • Partition keys overlap. • MINVALUE or MAXVALUE was specified for a start or end value of a partition key boundary and a subsequent key value for that range is not the same value.
Recovery Text:	Specify valid ranges for the STARTING, ENDING, or EVERY clause. Try the request again.
SQLCODE or SQLCODEs:	-636
SQLSTATE or SQLSTATEs:	56016

SQL0637

Message Text:	Keyword or clause starting with &1 not valid.
Cause Text:	Keyword or clause &1 is not valid for one of the following reasons: • DEFAULT, UNIQUE, and PRIMARY, each DataLink option, and each identity option can only be specified once in a column definition on a CREATE TABLE statement. • UNIQUE and PRIMARY cannot both be specified for the same column definition. • PRIMARY can only be specified once on a CREATE TABLE statement. • The IDENTITY COLUMN ATTRIBUTES, COLUMN DEFAULTS, and USING TYPE DEFAULTS options can only be specified once. • For ALTER TABLE ALTER COLUMN, each SET or DROP option can only be specified once for a column. • The attribute string for the PREPARE statement can only specify an option once and cannot contain conflicting options.
Recovery Text:	Remove all but one specification for each keyword or clause. Try the request again.
SQLCODE or SQLCODEs:	-637
SQLSTATE or SQLSTATEs:	42614

SQL0642	
Message Text:	Maximum number of constraints exceeded for &1 in &2.
Cause Text:	A constraint cannot be added to table &1 in &2 because the table is already associated with 300 constraints. This limit includes all constraints defined on the table and all referential constraints where the table is defined as a parent.
Recovery Text:	Drop one of the other 300 constraints, if possible, and try the request again.
SQLCODE or SQLCODEs:	-642
SQLSTATE or SQLSTATEs:	54021

SQL0644	
Message Text:	ATTRIBUTES value &1 not valid.
Cause Text:	The ATTRIBUTES string specified for the PREPARE statement contains &1, which is not correct. The type of SQL statement being prepared is &2.
Recovery Text:	Remove or correct the invalid attribute and try again.
SQLCODE or SQLCODEs:	-644
SQLSTATE or SQLSTATEs:	42615

SQL0645	
Message Text:	WHERE NOT NULL clause ignored for index &1 in &2.
Cause Text:	UNIQUE WHERE NOT NULL was specified when creating index &1 in &2. However, none of the columns in the index allow null values. The index is created as a unique index.
Recovery Text:	Remove the WHERE NOT NULL clause from the CREATE INDEX statement.
SQLCODE or SQLCODEs:	+645

SQL0645	
SQLSTATE or SQLSTATEs:	01528

SQL0658	
Message Text:	Function &1 in &2 cannot be dropped.
Cause Text:	Function &1 in schema &2 cannot be dropped because it was implicitly generated by the CREATE DISTINCT TYPE statement.
Recovery Text:	To drop the function, you must drop the distinct type with which this function is associated.
SQLCODE or SQLCODEs:	-658
SQLSTATE or SQLSTATEs:	42917

SQL0663	
Message Text:	Number of partitioning values is not correct.
Cause Text:	Wrong number of values were specified in the STARTING or ENDING clause for partition &1. The number of values specified must match the number of key columns for the partitioning key. Only one column can be specified in the partitioning key if the EVERY clause is used.
Recovery Text:	Change the number of key limit values specified so that it matches the number of partitioning key columns or change the number of columns in the partitioning key. Try the request again.
SQLCODE or SQLCODEs:	-663
SQLSTATE or SQLSTATEs:	53038

SQL0665	
Message Text:	The partition name or number &1 is not valid.
Cause Text:	Partition name or partition number &1 was specified in a DROP PARTITION clause but does not refer to a partition in the table. The partition name or number specified must identify an existing partition from the table.
Recovery Text:	Specify the partition name or number for an existing partition. Try the request again.
SQLCODE or SQLCODEs:	-665
SQLSTATE or SQLSTATEs:	53039

SQL0666	
Message Text:	SQL query exceeds specified time limit or storage limit.
Cause Text:	A database query was about to be started whose estimated run time of &1 exceeds the specified limit of &2 or whose estimated temporary storage usage of &3 exceeds the specified limit of &4. The query time limit and temporary storage limit are specified on the CHGQRYA CL command.

SQL0666	
Recovery Text:	The following changes to the query could reduce the estimated elapsed time or estimated amount of temporary storage usage. • Change the query so that fewer records are returned by adding more restrictive record selection specifications. • Change the record selection of the query so that an existing access path can be used to process the records more quickly. • Create an access path with keys that match the record selection of the query using the SQL CREATE INDEX statement. • Change the ordering specification so neither a temporary access path needs to be built nor a sort needs to be performed. • Change the grouping field specification to match the left-most key fields of an existing access path. This allows an existing access path to be used. • Specify the OPTIMIZE FOR NN ROWS clause where NN represents the actual number of records which are to be retrieved before the query is ended. This clause will cause the query optimizer to assume that the query will not be run to completion and consequently reduce the estimates. • Change the output type of your query to produce the results on the display. • Specify a new value for the query time limit with the QRYTIMLMT parameter of the CHGQRYA CL command. • Specify a new value for the query temporary storage limit with the QRYSTGLMT parameter of the CHGQRYA CL command. • Examine the query debug messages contained in the job log for performance information and suggestions. These messages will precede this message in the job log and will provide additional information and suggestions.
SQLCODE or SQLCODEs:	-666
SQLSTATE or SQLSTATEs:	57005

SQL0667	
Message Text:	FOREIGN key value does not match a value in the parent key of constraint &1 in &2.
Cause Text:	Every value in the FOREIGN key of the dependent table must have a matching value in the parent key of the parent table. For constraint &1 in &2, there is an existing value in FOREIGN key of table &3 in &4 that does not have a matching value in the parent table. The constraint cannot be added.
Recovery Text:	Update the rows in either the dependent table or parent table so that all values of the FOREIGN key have matching values in the parent key, or modify the definition of the keys in the referential constraint. Try the request again.
SQLCODE or SQLCODEs:	-667
SQLSTATE or SQLSTATEs:	23520

SQL0675	
Message Text:	Specified delete rule not allowed with trigger on table &1 in &2.
Cause Text:	The delete rule specified in referential constraint &3 in &4 on table &1 in &2 is not allowed for the specified trigger. Constraint rule DELETE CASCADE is not allowed with a delete trigger. Constraint rules DELETE SET NULL and DELETE SET DEFAULT are not allowed with an update trigger.

SQL0675	
Recovery Text:	Either use the RMVPFTRG command to remove the trigger, use the RMVPFCST command to remove the constraint, define the constraint with a valid delete rule, or define the trigger with a different event.
SQLCODE or SQLCODEs:	-675
SQLSTATE or SQLSTATEs:	42892

SQL0678	
Message Text:	Data type of literal &1 not compatible with column &3.
Cause Text:	The literal specified is not compatible. If literal &1 is specified for the STARTING, ENDING, or EVERY clause, it is not compatible with column &3. The data type of column &3 is &2. If the literal is specified in a comparison with a numeric value, the literal is not a valid numeric value.
Recovery Text:	Specify a valid literal. Try the request again.
SQLCODE or SQLCODEs:	-678
SQLSTATE or SQLSTATEs:	53045

SQL0679	
Message Text:	Object &1 in &2 type *&3 not created due to pending operation.
Cause Text:	The object &1 has an outstanding DROP or CREATE under commitment control which is preventing the create. This could have occurred in one of the following ways: • This application process has performed a DROP under commitment control which has not been committed and is now trying to create the same object using commitment control level *NONE. • A different application process has performed a DROP under commitment control which has not been committed. • This application process has performed a DROP under commitment control using a different commit definition and the DROP has not been committed. • This application process has performed a CREATE under commitment control which has not been committed and is now trying to use the object on a subsequent CREATE under commitment control level *NONE.
Recovery Text:	Do one of the following and try the request again: • If it was your application process which issued the uncommitted DROP or CREATE statement then issue a COMMIT before attempting the creation of the object or issue the CREATE statement from a program using a commitment control level other than *NONE. • If the application process that issued the DROP on this object is not your application, then that application process must perform a COMMIT or a ROLLBACK of the DROP statement. • If your application process issued the uncommitted DROP or CREATE statement under a different commit definition, issue a COMMIT or ROLLBACK for that commit definition.
SQLCODE or SQLCODEs:	-679
SQLSTATE or SQLSTATEs:	57006

SQL0680	
Message Text:	Too many columns specified.
Cause Text:	Too many columns were specified in the definition of a user defined table function. A maximum of 125 columns can be specified for the input parameters and the return columns combined. This maximum is reduced by one if both the SCRATCHPAD and DBINFO structures are requested.
Recovery Text:	Reduce the number of parameters or return columns specified for the user defined table function. Try the request again.
SQLCODE or SQLCODEs:	-680
SQLSTATE or SQLSTATEs:	54011

SQL0683	
Message Text:	Clause not valid for specified type.
Cause Text:	One of the following has occurred: • The FOR BIT DATA, the FOR MIXED DATA, the FOR SBCS DATA, or the CCSID clause was specified for the CAST scalar function, for a CREATE DISTINCT TYPE source data type, or for column or parameter &1. These clauses are not valid if the type is numeric, date, time, timestamp, binary, or a user defined type. The FOR BIT DATA, FOR MIXED DATA, or FOR SBCS DATA clause is not valid if the type is graphic or DataLink. • The LINKTYPE option was specified for a column that is not a DataLink.
Recovery Text:	If the CCSID clause is specified, change the specified type to be character, varying-length character, CLOB, graphic, varying-length graphic, DBCLOB, or DATALINK. If the FOR BIT DATA, FOR MIXED DATA, FOR SBCS DATA clause is specified, change the type to be character, varying-length character, or CLOB. If the type is correct, remove the clause. If the LINKTYPE option is specified, change the specified type to be DataLink or remove the clause.
SQLCODE or SQLCODEs:	-683
SQLSTATE or SQLSTATEs:	42842

SQL0696	
Message Text:	Correlation name or table &3 not valid.
Cause Text:	The correlation name or table &3 is not valid for trigger &1 in &2. The reason code is &4. One of the following reason codes indicates the error: 1 - NEW correlation name or NEW_TABLE &3 was specified in a DELETE trigger. 2 - OLD correlation name or OLD_TABLE &3 was specified in an INSERT trigger. 3 - OLD_TABLE or NEW_TABLE was specified in a BEFORE trigger or was specified with DB2ROW.
Recovery Text:	Change the referencing clause or the type of trigger so that the statement is valid. Try the request again.
SQLCODE or SQLCODEs:	-696
SQLSTATE or SQLSTATEs:	42898

SQL0697	
Message Text:	REFERENCING OLD or NEW not valid for statement trigger.

SQL0697	
Cause Text:	REFERENCING OLD or NEW was specified for an SQL trigger. Old and new correlation variables are not valid for statement triggers.
Recovery Text:	Remove the REFERENCING clause or specify FOR EACH ROW on the CREATE TRIGGER statement. Try the request again.
SQLCODE or SQLCODEs:	-697
SQLSTATE or SQLSTATEs:	42899

SQL0707	
Message Text:	Name &1 in &2 not allowed.
Cause Text:	Name &1 is not allowed. • Distinct type or sequence &1 in &2 cannot be created. Either the name is a reserved word or the schema is specified as QSYS, QSYS2, SYSIBM, or QTEMP. Distinct types and sequences cannot be created in QSYS, QSYS2, SYSIBM, or QTEMP. • Savepoint &1 cannot have a name that starts with SYS.
Recovery Text:	Change the name to a non-reserved word, or create the user-defined data type or sequence in a schema other than QSYS, QSYS2, SYSIBM, or QTEMP.
SQLCODE or SQLCODEs:	-707
SQLSTATE or SQLSTATEs:	42939

SQL0713	
Message Text:	Value for &2 is not correct.
Cause Text:	The value specified in the SET &2 statement is not correct. The NULL value cannot be used to set the &2 special register. &2 can only be set using a character or UCS-2 or UTF-16 graphic string. If this is SET ENCRYPTION PASSWORD WITH HINT, the hint value cannot be longer than 32 characters. If this is SET SCHEMA, all letters in the schema name must be uppercase or the schema name must be delimited. If this is SET CURRENT DEGREE, the value cannot be longer than 5 characters. If this is a SET SESSION AUTHORIZATION, the authorization name cannot be longer than 10 characters. An authorization name specified as a string literal or with a host variable cannot contain the special register strings USER, SYSTEM_USER, or SESSION_USER and cannot contain lowercase letters or special characters.
Recovery Text:	Specify a valid value for the special register. Make sure it does not have the NULL value. Try the request again.
SQLCODE or SQLCODEs:	-713
SQLSTATE or SQLSTATEs:	3F000, 42815

SQL0723	
Message Text:	SQL trigger &1 in &2 failed with SQLCODE &3 SQLSTATE &4.
Cause Text:	An error has occurred in a triggered SQL statement in trigger &1 in schema &2. The SQLCODE is &3, the SQLSTATE is &4, and the message is &5.
Recovery Text:	Refer to the joblog for more information regarding the detected error. Correct the error and try the request again.

SQL0723	
SQLCODE or SQLCODEs:	-723
SQLSTATE or SQLSTATEs:	09000

SQL0724	
Message Text:	Too many cascaded trigger programs.
Cause Text:	The maximum depth of 200 cascaded triggers has been exceeded.
Recovery Text:	Remove any trigger that is causing repeated trigger programs to be called for the same table.
SQLCODE or SQLCODEs:	-724
SQLSTATE or SQLSTATEs:	54038

SQL0751	
Message Text:	SQL statement &1 not allowed.
Cause Text:	The statement &1 is not allowed in a stored procedure, user-defined function, or trigger. • Statements not allowed in a trigger program are CONNECT, SET CONNECTION, RELEASE, DISCONNECT, and SET RESULT SETS. • RUNSQLSTM is not allowed in a trigger program. • COMMIT and ROLLBACK are not allowed in a trigger program if the trigger program is running in the same activation group as the triggering program. COMMIT and ROLLBACK are not allowed in an SQL trigger. • ALTER TABLE is not allowed in a trigger program when commitment control is active. • Statements not allowed in a stored procedure or user-defined function that is running on a remote application server are CONNECT, SET CONNECTION, RELEASE, DISCONNECT, COMMIT, ROLLBACK, and SET TRANSACTION. • Statements not allowed in an SQL BEFORE trigger are INSERT, UPDATE, DELETE, ALTER TABLE, COMMENT ON, CREATE, DROP, GRANT, LABEL ON, RENAME, REVOKE, REFRESH TABLE, and ALTER PROCEDURE. • The RETURN statement is not allowed in an SQL trigger. • Statements not allowed in a secondary thread are CREATE TRIGGER, CREATE FUNCTION (SQL), CREATE PROCEDURE (SQL), and ALTER PROCEDURE (SQL).
Recovery Text:	Remove the statement &1 from your trigger program, user-defined function, or stored procedure. Try the request again.
SQLCODE or SQLCODEs:	-751
SQLSTATE or SQLSTATEs:	0W000, 2F003, 38003, 42985, 42987

SQL0752	
Message Text:	Connection cannot be changed. Reason code is &1.

SQL0752	
Cause Text:	Connection cannot be made because the application process is not in a connectable state. The reason code is &1. Reason codes and their meanings are: • 1 - SQL is not in a connectable state. SQL enters the connectable state after a COMMIT or ROLLBACK. SQL leaves the connectable state when any SQL statement except a COMMIT, ROLLBACK, or CONNECT statement is run. • 2 - There are pending changes or open files under commitment control at the current server and the relational database (RDB) specified on the CONNECT request is not the current server. • 3 - A create SQL package request is being processed when not on a commit boundary. • 4 - The connection is locked by another invocation of Interactive SQL, or there is a level mismatch between the Interactive SQL product and the DB2 UDB for iSeries product. • 5 - The connection cannot be changed due to restrictions with remote connections and the job level commitment definition. • 6 - The connection cannot be changed to a remote system due to a SET TRANSACTION statement. • 7 - The connection cannot be changed using *RUW connection management because a previous connection is protected. • 8 - CONNECT RESET cannot start a local connection to RDB &2 because the RDB for the auxiliary storage pool (ASP) group of the thread is &3. • 9 - There is a mismatch between the active connection and the current ASP group of the thread.
Recovery Text:	Do one of the following based on the reason code: • 1 - Issue a COMMIT or ROLLBACK statement to enter the connectable state. • 2 - Close all files open under commitment control and issue a COMMIT or ROLLBACK statement. • 3 - Issue a COMMIT or ROLLBACK statement. • 4 - Exit Interactive SQL and try the request again. If Interactive SQL is active, the current server can only be changed using Interactive SQL. • 5 - All activation groups associated with the job level commitment definition must be local; or only one remote connection and no local connections can be associated with the job level commitment definition. • 6 - Issue a COMMIT or ROLLBACK statement. • 7 - Release the protected conversation followed by a COMMIT. • 8 - Change the ASP group of the thread to the ASP group for relational database &2. • 9 - Either change the ASP group of the thread to the ASP group for relational database &2 or use a CONNECT or SET CONNECTION statement to change the active connection.
SQLCODE or SQLCODEs:	-752
SQLSTATE or SQLSTATEs:	0A001

SQL0771	
Message Text:	Table with ROWID column not allowed in QTEMP.
Cause Text:	A table with a ROWID column cannot be created in QTEMP.
Recovery Text:	Remove the ROWID column or create the table in a schema other than QTEMP. Try the request again.
SQLCODE or SQLCODEs:	-771

SQL0771	
SQLSTATE or SQLSTATEs:	428C7

SQL0773	
Message Text:	Case not found for CASE statement.
Cause Text:	A CASE statement without an ELSE clause was specified in the routine body of an SQL procedure. None of the conditions specified in the CASE statement were met.
Recovery Text:	Change the CASE statement to handle all conditions that can occur.
SQLCODE or SQLCODEs:	-773
SQLSTATE or SQLSTATEs:	20000

SQL0774	
Message Text:	Statement cannot be executed within a compound SQL statement.
Cause Text:	
Recovery Text:	
SQLCODE or SQLCODEs:	-774
SQLSTATE or SQLSTATEs:	2D522

SQL0775	
Message Text:	Statement not allowed in specified SQL routine.
Cause Text:	A statement specified in the routine body of an SQL procedure or function is not allowed. A list of restrictions follows: • A COMMIT or ROLLBACK statement cannot be specified in an atomic compound statement in an SQL procedure. • An ATOMIC compound statement cannot be specified in an SQL function. • COMMIT, ROLLBACK, CONNECT, DISCONNECT, SET CONNECTION, SET RESULT SETS, and SET TRANSACTION statements cannot be specified in an SQL function. • The SET RESULT SETS statement cannot be specified in an SQL routine body unless RESULT SET is specified for the procedure.
Recovery Text:	Remove the statement from the SQL function or procedure.
SQLCODE or SQLCODEs:	-775
SQLSTATE or SQLSTATEs:	42910

SQL0776	
Message Text:	Cursor &1 specified in FOR statement not allowed.
Cause Text:	Cursor &1 is specified as the cursor name on a FOR statement in an SQL procedure. The cursor cannot be specified on a FETCH, OPEN, or CLOSE statement within the FOR statement.
Recovery Text:	Remove the OPEN, CLOSE, or FETCH statement.

SQL0776	
SQLCODE or SQLCODEs:	-766
SQLSTATE or SQLSTATEs:	428D4

SQL0777	
Message Text:	Nested compound statements not allowed.
Cause Text:	Compound statements in the routine body of an SQL procedure or function cannot be nested.
Recovery Text:	
SQLCODE or SQLCODEs:	-777
SQLSTATE or SQLSTATEs:	42919

SQL0778	
Message Text:	End label &1 not same as begin label.
Cause Text:	Label &1 specified at the end of a compound, FOR, WHILE, REPEAT, or LOOP statement in an SQL procedure or function is not the same as the label at the beginning of the statement. The end label cannot be specified if the begin label is not specified.
Recovery Text:	Ensure the end label is the same as the begin label for compound, FOR, WHILE, REPEAT, and LOOP statements.
SQLCODE or SQLCODEs:	-778
SQLSTATE or SQLSTATEs:	428D5

SQL0779	
Message Text:	Label &1 specified not valid.
Cause Text:	Label &1 is specified on a LEAVE or a GOTO statement in an SQL procedure or function. The label is not a valid label or is not in the same scope as the current statement.
Recovery Text:	Specify a valid label that is within the same scope. Try the request again.
SQLCODE or SQLCODEs:	-779
SQLSTATE or SQLSTATEs:	42736

SQL0780	
Message Text:	UNDO specified for a handler not valid.
Cause Text:	UNDO is specified for a handler in a compound statement in an SQL procedure, function, or trigger. UNDO cannot be specified unless the compound statement is ATOMIC. UNDO cannot be specified in a trigger.
Recovery Text:	Either specify an ATOMIC compound statement or specify EXIT or CONTINUE on the handler.
SQLCODE or SQLCODEs:	-780

SQL0780	
SQLSTATE or SQLSTATEs:	428D6

SQL0781	
Message Text:	Condition &1 specified in handler not defined.
Cause Text:	Condition &1 specified in a handler in an SQL procedure or function is not defined.
Recovery Text:	Define the condition using the DECLARE CONDITION statement or remove the condition from the handler.
SQLCODE or SQLCODEs:	-781
SQLSTATE or SQLSTATEs:	42737

SQL0782	
Message Text:	Condition value &1 specified in handler not valid.
Cause Text:	Condition &1 specified in a handler in an SQL procedure or function is not valid for one of the following reasons: • The condition value has already been specified by another handler in the same scope. • The condition or SQLSTATE was specified in the same handler as SQLEXCEPTION, SQLWARNING, or NOT FOUND.
Recovery Text:	Remove the condition from the handler.
SQLCODE or SQLCODEs:	-782
SQLSTATE or SQLSTATEs:	428D7

SQL0783	
Message Text:	Select list for cursor &1 in FOR statement not valid.
Cause Text:	The select list in the FOR statement must contain unique column names. The select list specified either contains duplicate column names or unnamed expressions. If two column names are the same, the column name is &2.
Recovery Text:	Specify unique column names in the select list specified in the FOR statement.
SQLCODE or SQLCODEs:	-783
SQLSTATE or SQLSTATEs:	42738

SQL0784	
Message Text:	Constraint &1 cannot be dropped.
Cause Text:	Constraint &1 is a CHECK constraint or a UNIQUE constraint. It cannot be dropped because it is enforcing a primary key to be not null or a ROWID to be unique.

SQL0784	
Recovery Text:	Do one of the following and try the request again: • Drop the primary key which this CHECK constraint is enforcing to be not null. If the primary key is needed, change the attributes of the columns that make up the primary key to be NOT NULL, and then add the primary key again. • Drop the ROWID column. An identity column could be used instead of the ROWID column.
SQLCODE or SQLCODEs:	-784
SQLSTATE or SQLSTATEs:	42860

SQL0785	
Message Text:	Use of SQLCODE or SQLSTATE not valid.
Cause Text:	SQLCODE or SQLSTATE was used as a variable in the routine body of an SQL procedure, but is not valid for one of the following reasons: • SQLCODE is not declared as INT. • SQLSTATE is not declared as CHAR(5). • The variable is set to NULL.
Recovery Text:	Declare the SQLCODE variable as INT and the SQLSTATE variable as CHAR(5). Set the variable to a valid value.
SQLCODE or SQLCODEs:	-785
SQLSTATE or SQLSTATEs:	428D8

SQL0787	
Message Text:	RESIGNAL statement not within a handler.
Cause Text:	The RESIGNAL statement, specified in an SQL routine, must be specified inside a handler.
Recovery Text:	Remove the RESIGNAL statement or use a SIGNAL statement. Try the request again.
SQLCODE or SQLCODEs:	-787
SQLSTATE or SQLSTATEs:	0K000

SQL0798	
Message Text:	Value cannot be specified for GENERATED ALWAYS column &1.
Cause Text:	A value cannot be specified for column &1 because it is defined as GENERATED ALWAYS.
Recovery Text:	Do one of the following and try the request again: • Remove the value from the INSERT or UPDATE statement, or specify DEFAULT to have a system value generated for the column. • Specify the OVERRIDING USER VALUE clause on the INSERT or UPDATE statement to override the GENERATED ALWAYS attribute and assign the value to the column • Remove the column from the SET statement.
SQLCODE or SQLCODEs:	-798

SQL0798	
SQLSTATE or SQLSTATEs:	428C9

SQL0799	
Message Text:	Special register &1 does not exist at the application server.
Cause Text:	A SET statement references special register &1. This special register does not exist on the application server so it cannot be referenced in the statement.
Recovery Text:	Remove the reference to the unsupported special register from the statement. Try the request again.
SQLCODE or SQLCODEs:	+799
SQLSTATE or SQLSTATEs:	01527

SQL0802	
Message Text:	Data conversion or data mapping error.
Cause Text:	Error type &3 has occurred. Error types and their meanings are: • 1 - Arithmetic overflow. • 2 - Floating point overflow. • 3 - Floating point underflow. • 4 - Floating point conversion error. • 5 - Not an exact result. • 6 - Numeric data that is not valid. • 7 - Double-byte character set (DBCS) data that is not valid. • 8 - Division by zero. • 9 - Hash value cannot be computed for the requested query. • 10 - User-defined function returned a mapping error. • 11 - Not valid length found in a varying-length column returned from an array result set. • 12 - Result of a concatenation operation on a varying-length field exceeded the maximum allowed length of the result type. If the error occurred when assigning a value to a host variable of a FETCH, embedded SELECT, SET, or VALUES INTO statement, the host variable name is &2 and the relative position of the host variable in the INTO clause is &1. If the host variable name is *N, the error occurred when attempting to resolve a search condition. If more than one data mapping error occurred, this is a description of the first error that occurred. For a description of any other data mapping errors, see the previously listed messages in the job log.
Recovery Text:	The error was caused by data that was not valid or that was too large. Look at the previously listed messages in the job log (DSPJOBLOG command) or press F10 (Display messages in job log) on this display to determine what row and columns were involved in the error. Correct the data and then try the request again.
SQLCODE or SQLCODEs:	+802, -802
SQLSTATE or SQLSTATEs:	01004, 01519, 01547, 01564, 01565, 22001, 22003, 22012, 22023, 22504

SQL0803	
Message Text:	Duplicate key value specified.

SQL0803	
Cause Text:	An INSERT, UPDATE or ALTER TABLE statement was issued. Unique index or unique constraint &1 in &2 exists over one or more columns of table &3 in &4. The operation cannot be performed because one or more values would have produced a duplicate key in the unique index or constraint.
Recovery Text:	Change the statement so that duplicate keys are not produced. For information on what rows contain the duplicate key values, look at the previously listed messages in the job log (DSPJOBLOG command) or press F10 (Display messages in job log) on this display.
SQLCODE or SQLCODEs:	-803
SQLSTATE or SQLSTATEs:	23505

SQL0804	
Message Text:	SQLDA or descriptor area not valid.
Cause Text:	If the error type is 2, 3, 9, 12, or 13, the entry in error is &2, the value of SQLTYPE or TYPE is &3, and the value of SQLLEN, SQLLONGLEN, or LENGTH is &4. If the error type is 13, the &5 was being set when the inconsistency was found. The specified SQLDA or descriptor area is not valid because of error type &1. Error types and their meanings are: • 1 - The value of SQLN is less than zero, the value of SQLD is not between 0 and 8000, the value of SQLD is greater than the value of SQLN, or that the value of SQLD has not been initialized in REXX. • 2 - The value of SQLTYPE is either not valid or not supported or has not been initialized in REXX. The types that are not supported in REXX are NUL-terminated graphic string, NUL-terminated character string, PASCAL L-string, sign leading separate, and binary with precision and scale. • 3 - The value of SQLLEN or SQLLONGLEN is not valid or the value of SQLLEN, SQLPRECISION, or SQLSCALE has not been initialized in REXX. If REXX and SQLTYPE is decimal or numeric, then either SQLPRECISION or SQLSCALE has not been initialized. Otherwise, SQLLEN has not been initialized. If SQLTYPE is a LOB variable, then SQLLONGLEN is not valid. • 5 - The SQLDA area was not on a 16-byte boundary. • 6 - The value specified for SQLDABC is not valid. The value is either not large enough for the number of entries specified in SQLN or the value is greater than the maximum allowed. • 7 - The value of SQLN was not at least twice the size of SQLD and LOB host variables were found in the SQLDA. • 8 - The seventh byte of SQLDAID was not a '2', '3' or '4' and LOB host variables were found in the SQLDA. • 9 - The SQLDATAL pointer was not null for a DBCLOB host variable, but the length value referenced by the SQLDATAL pointer had an odd value. • 10 - The SQLTYPE for a LOB locator did not match the type associated with LOB locator. • 11 - The row length is greater than the maximum allowed row length. • 12 - The TYPE for an item in the descriptor area has not been set. • 13 - The attributes of the item in the descriptor area are not consistent. TYPE, LENGTH, PRECISION, SCALE, DB2_CCSID, and DATETIME_INTERVAL_CODE need to be consistent with one another for each item set with the SET DESCRIPTOR statement.
Recovery Text:	Correct the error in the SQLDA or descriptor area and try the request again.
SQLCODE or SQLCODEs:	-804
SQLSTATE or SQLSTATEs:	07002

SQL0805	
Message Text:	SQL package &1 in &2 not found at DRDA Server.
Cause Text:	A remote request was attempted to &4 for SQL package &1 in &2. The SQL package was not found. If you are using Interactive SQL or Query Manager, an attempt to create a package on the remote system failed (see common cause below) and the package requested does not exist.
Recovery Text:	The most common cause of this problem in an Interactive SQL session to a non-iSeries server is that the connection is not updateable. In that case the package cannot be automatically created. To ensure the connection is updateable, do a RELEASE ALL command followed by a COMMIT before connecting to the relational database. In other cases, the SQL package can be created by using the CRTSQLPKG command. Also, precompiling an SQL program with the RDB parameter specified will create an SQL package on the system. Create or restore the SQL package. Run the application again. If you are using Interactive SQL or Query Manager or SQL Call Level Interface, exit the product and enter a CL COMMIT or ROLLBACK command. This will enable you to continue processing at the local system. Determine why the package creation failed by examining the job log. Correct the problem and attempt the Interactive SQL or Query Manager session again.
SQLCODE or SQLCODEs:	-805
SQLSTATE or SQLSTATEs:	51002

SQL0809	
Message Text:	Row length exceeds 3.5 gigabytes.
Cause Text:	The lengths of the values used as input in the INSERT or UPDATE statement would cause the row length to exceed the maximum length of 3.5 gigabytes.
Recovery Text:	Change the statement so that the sum of all field lengths result in a row length of less than 3.5 gigabytes.
SQLCODE or SQLCODEs:	-809
SQLSTATE or SQLSTATEs:	54018

SQL0811	
Message Text:	Result of SELECT more than one row.
Cause Text:	The result table of a SELECT INTO statement, a subquery, or a subselect of a SET statement contains more than one row. The error type is &1. If the error type is 1 then a SELECT INTO statement attempted to return more than one row. If the error type is 2 then a subselect of a basic predicate has produced more than one row. Only one row is allowed.
Recovery Text:	Change the selection so that only one result row is returned and then try the request again. The DECLARE CURSOR, OPEN, and FETCH statements must be used to process more than one result row. For a subquery the IN, EXISTS, ANY or ALL predicates can be used to process more than one result row. If one row was expected, there may be data errors, such as duplicate rows, that are causing more than one row to be returned.
SQLCODE or SQLCODEs:	-811
SQLSTATE or SQLSTATEs:	21000

SQL0817	
Message Text:	Update operation not allowed.
Cause Text:	An attempt has been made to run an SQL statement that would change the contents of a table or create or drop a database object after SET TRANSACTION READ ONLY has been performed.
Recovery Text:	Specify SET TRANSACTION READ WRITE, or remove the update operation from the program and try again.
SQLCODE or SQLCODEs:	-817
SQLSTATE or SQLSTATEs:	25006

SQL0818	
Message Text:	Consistency tokens do not match.
Cause Text:	Package &3 in &4 on application server &5 cannot be run with the current application program because either the application program has been recompiled without rebuilding the package or the package has been restored from a back level version.
Recovery Text:	Rebuild the package by using the CRTSQLPKG command or by using a CRTSQLxxx command specifying the correct relational database. Otherwise, restore the package from a version which matches the application program being run.
SQLCODE or SQLCODEs:	-818
SQLSTATE or SQLSTATEs:	51003

SQL0822	
Message Text:	Address in the SQLDA or descriptor area not valid.
Cause Text:	The SQLDA or descriptor area contains an address, SQLDATA or DATA value, or SQLIND or INDICATOR value in entry number &1 that is not valid. The incorrect address or value is type &2. • Type 1 indicates that the SQLDATA or DATA address is not valid. • Type 2 indicates that the SQLIND or INDICATOR address is not valid. • Type 3 indicates that the SQLDA address is not valid. • Type 4 indicates that the row storage area is not large enough. • Type 5 indicates that the indicator area for a blocked FETCH statement is not large enough. • Type 6 indicates that the SQLDATA field was not initialized to a value in a REXX procedure. • Type 7 indicates that the SQLIND field was not initialized to a value in a REXX procedure. • Type 8 indicates that the SQLDATAL address is not valid.
Recovery Text:	For types 1, 2, 3, or 8, change the address in entry &1 to a valid address. For types 4 and 5, allocate enough area for all of the rows being requested. For types 6 and 7, initialize the SQLDATA or SQLIND fields to a valid value. Try the request again.
SQLCODE or SQLCODEs:	-822
SQLSTATE or SQLSTATEs:	51004

SQL0827	
Message Text:	&1 in &2 type *SQLPKG cannot be accessed.
Cause Text:	SQL Package &1 in &2 was not created using the QSQPRCED API and cannot be accessed by the QSQPRCED API. *SQLPKG objects created using CRTSQLPKG or the CRTSQLxxx commands cannot be used by the QSQPRCED API.
Recovery Text:	Use the QSQPRCED API to create a new *SQLPKG object. Change your request to use the package created by the API.
SQLCODE or SQLCODEs:	-827
SQLSTATE or SQLSTATEs:	42862

SQL0840	
Message Text:	Number of selected items exceeds 8000.
Cause Text:	The number of items returned in a select list or presented in the insert list exceeds the maximum of 8000.
Recovery Text:	Reduce the number of selected items and try the request again.
SQLCODE or SQLCODEs:	-840
SQLSTATE or SQLSTATEs:	54004

SQL0842	
Message Text:	Connection to relational database &1 already exists.
Cause Text:	An attempt was made to do one of the following: • CONNECT to a relational database when the connection is active. • CONNECT to a relational database that has the same communication information as a connection to a relational database that is active. The active relational database is &1.
Recovery Text:	If CONNECT was specified, either use the SET CONNECTION statement to make relational database &1 the current connection or change the RDB directory entry (CHGRDBDIRE) for the relational database you are connecting to so that at least part of the communication information is different from what is specified in the entry for &1. For APPC connections, the communication information is the remote location, device description, local location, remote network identifier, mode, and transaction program. For TCP/IP connections, the communication information is the remote location and port identification.
SQLCODE or SQLCODEs:	-842
SQLSTATE or SQLSTATEs:	08002

SQL0843	
Message Text:	Connection to relational database &1 does not exist.
Cause Text:	A SET CONNECTION, RELEASE, or DISCONNECT statement specified relational database name &1 which is not active.
Recovery Text:	Specify the name of a relational database which has an active connection.
SQLCODE or SQLCODEs:	-843

SQL0843	
SQLSTATE or SQLSTATEs:	08003

SQL0845	
Message Text:	PREVIOUS VALUE for sequence &1 cannot be used.
Cause Text:	A PREVIOUS VALUE expression specified sequence &1 in &2, but a value is not available in this application process. • A NEXT VALUE expression must be evaluated before a PREVIOUS VALUE expression can be used. • A NEXT VALUE expression must be evaluated after a sequence is altered or dropped.
Recovery Text:	Evaluate the NEXT VALUE expression for sequence &1 in &2 in the same application process before using PREVIOUS VALUE.
SQLCODE or SQLCODEs:	-845
SQLSTATE or SQLSTATEs:	51035

SQL0846	
Message Text:	Attributes not valid for IDENTITY column or sequence.
Cause Text:	Error &6 occurred for an IDENTITY column or sequence. For an IDENTITY column, &2 in schema &3 is the name of the table. For a sequence, &2 in &3 is the name of the sequence. • Code 1 - The data type of an IDENTITY column or sequence is not INTEGER, BIGINT, SMALLINT, or DECIMAL or NUMERIC with a scale of zero. An IDENTITY column that is DECIMAL or NUMERIC must have a precision less than or equal to 31. • Code 2 - The value specified for START WITH, INCREMENT BY, MINVALUE, or MAXVALUE is outside the range for the data type specified. • Code 3 - The value specified for MINVALUE is larger than the value specified for MAXVALUE. • Code 4 - The value specified for CACHE is not valid. The minimum value for CACHE is 2. • Code 7 - A sequence can only be altered to a distinct type if the current data type is a built-in data type and it is promotable to the source data type of the distinct type. If the current data type is a distinct type, it can only be changed to its source data type.
Recovery Text:	Specify valid attributes for the IDENTITY column or sequence. Try the request again.
SQLCODE or SQLCODEs:	-846
SQLSTATE or SQLSTATEs:	42815

SQL0858	
Message Text:	Cannot disconnect relational database &1 due to LU6.2 protected conversation.
Cause Text:	The DISCONNECT statement cannot be used to disconnect relational database &1 because the conversation uses an LU6.2 protected conversation.
Recovery Text:	Use the RELEASE statement followed by a COMMIT statement to end LU6.2 protected conversations.
SQLCODE or SQLCODEs:	-858

SQL0858	
SQLSTATE or SQLSTATEs:	08501

SQL0862	
Message Text:	Local program attempted to connect to a remote relational database.
Cause Text:	Local program &1 in &2 attempted to connect to a remote relational database. Either the CONNECT statement or the SET CONNECTION statement was specified and the relational database specified was a remote relational database.
Recovery Text:	Specify the RDB parameter on the SQL precompile command.
SQLCODE or SQLCODEs:	-862
SQLSTATE or SQLSTATEs:	55029

SQL0863	
Message Text:	Mixed or DBCS CCSID not supported by relational database &1.
Cause Text:	The connection was completed, but remote relational database &1 does not support either the mixed or DBCS CCSID. SBCS data can be used. The product identification is &2.
Recovery Text:	No recovery needed.
SQLCODE or SQLCODEs:	+863
SQLSTATE or SQLSTATEs:	01539

SQL0871	
Message Text:	Too many CCSID values specified.
Cause Text:	More than 80 unique combinations of character data type and Coded Character Set Identifier (CCSID) were used. When accessing remote data, there is a limit of 80 different CCSID values.
Recovery Text:	Change that request to only access 80 different combinations of character data type and CCSID.
SQLCODE or SQLCODEs:	-871
SQLSTATE or SQLSTATEs:	54019

SQL0880	
Message Text:	Savepoint &1 does not exist or is not valid in this context.
Cause Text:	The RELEASE TO SAVEPOINT or ROLLBACK TO SAVEPOINT statement does not identify a savepoint that exists at the current savepoint level.
Recovery Text:	Correct the statement to use a valid savepoint name, then try the statement again.
SQLCODE or SQLCODEs:	-880
SQLSTATE or SQLSTATEs:	3B001

SQL0881	
Message Text:	Savepoint &1 already exists.
Cause Text:	The savepoint name was previously defined at the current savepoint level and either an existing savepoint or the new savepoint is defined with the UNIQUE keyword.
Recovery Text:	Either use a different savepoint name or omit the UNIQUE clause if the existing savepoint was created without the UNIQUE clause and the savepoint name is intended to be reused. An existing savepoint can be released using the RELEASE TO SAVEPOINT statement.
SQLCODE or SQLCODEs:	-881
SQLSTATE or SQLSTATEs:	3B501

SQL0882	
Message Text:	Savepoint does not exist.
Cause Text:	A ROLLBACK TO SAVEPOINT without a savepoint name was attempted but no savepoint exists at the current savepoint level.
Recovery Text:	Correct the application logic to either set a savepoint or to not attempt to rollback to a savepoint.
SQLCODE or SQLCODEs:	-882
SQLSTATE or SQLSTATEs:	3B502

SQL0900	
Message Text:	Application process not in a connected state.
Cause Text:	One of the following occurred: • The current connection was disconnected using the DISCONNECT statement. • The current connection was released and a commit occurred • A previous error has left the application process in an unconnected state. Use the Display Job Log (DSPJOBLOG) command to see previous errors.
Recovery Text:	Issue CONNECT statement with the TO or RESET clause or the SET CONNECTION statement to enter the connected state.
SQLCODE or SQLCODEs:	-900
SQLSTATE or SQLSTATEs:	08003

SQL0901	
Message Text:	SQL system error.
Cause Text:	An SQL system error has occurred. The current SQL statement cannot be completed successfully. The error will not prevent other SQL statements from being processed. Previous messages may indicate that there is a problem with the SQL statement and SQL did not correctly diagnose the error. The previous message identifier was &1. Internal error type &2 has occurred. If precompiling, processing will not continue beyond this statement.
Recovery Text:	See the previous messages to determine if there is a problem with the SQL statement. To view the messages, use the DSPJOBLOG command if running interactively, or the WRKJOB command to view the output of a precompile. An application program receiving this return code may attempt further SQL statements. Correct any errors and try the request again.

SQL0901	
SQLCODE or SQLCODEs:	-901
SQLSTATE or SQLSTATEs:	58004

SQL0904	
Message Text:	Resource limit exceeded.
Cause Text:	Resource limit type &1 exceeded with reason code &2. A list of the limit types follows: • Type 1 indicates that the user profile storage limit or the machine storage limit was exceeded. • Type 2 indicates that the machine lock limit was exceeded. • Type 3 indicates that the query resource limit was exceeded. For more information see the previously listed message CPD4365. • Type 4 indicates that a journal error has occurred. • Type 5 indicates that the commit lock limit was exceeded. • Type 6 indicates that the maximum size of the table has been reached. • Type 7 indicates that the maximum size of the prepared statement area has been reached. • Type 8 indicates that the maximum number of cursors have been opened for this job. • Type 9 indicates that the maximum number of entries in the lock table have been used for this job. • Type 12 indicates that the maximum DRDA communications buffer size was exceeded. • Type 13 indicates that the maximum amount of blocked data was exceeded. • Type 14 indicates that the maximum amount of descriptor space has been allocated.
Recovery Text:	Do one of the following: • If this is error type 1, contact the security officer to increase the user profile storage limit, or delete some objects to free up storage and then try the request again. • If this is error type 2, then try the operation when the number of machine locks held has decreased. • If this is error types 3, 4, or 5, see previously listed messages in the job log for recovery information. • If this is error type 6, Some of the rows from this table must be moved to another table. • If this is error type 7, issue a COMMIT or ROLLBACK without the HOLD clause before issuing anymore PREPARE statements. • If this is error type 8, issue a CLOSE before issuing any more OPEN statements. • If this is error type 9, issue a COMMIT or ROLLBACK without the HOLD clause. • If this is error type 12, reduce the total size of column data supplied with the SQL request. • If this is error type 13, reduce the number of rows in the block. • If this is error type 14, reduce the number of allocated descriptors with the DEALLOCATE DESCRIPTOR statement.
SQLCODE or SQLCODEs:	-904
SQLSTATE or SQLSTATEs:	57011

SQL0906	
Message Text:	Operation not performed because of previous error.

SQL0906	
Cause Text:	A previous error has made cursor &1 not usable.
Recovery Text:	The cursor is not usable. Perform the following steps: 1) Close the cursor. 2) Open the cursor. 3) Try the operation again.
SQLCODE or SQLCODEs:	-906
SQLSTATE or SQLSTATEs:	24514

SQL0907	
Message Text:	Data change violation occurred.
Cause Text:	The row referenced by the statement which caused a trigger program to be invoked was referenced again in the trigger program. The reference in the trigger program attempted to update or delete the row. This is called a destructive data change and is not allowed.
Recovery Text:	Remove the statement which caused the error from your trigger program and attempt the request again.
SQLCODE or SQLCODEs:	-907
SQLSTATE or SQLSTATEs:	27000

SQL0910	
Message Text:	Object &1 in &2 type *&3 has a pending change.
Cause Text:	Object &1 has an outstanding change made under commitment control that is preventing this operation. One of the following may have occurred: • This application process performed an operation on this object under commitment control. The operation has not been committed. The application process is now attempting to change the same object using commitment control level *NONE. • A different application process has performed an operation on this object under commitment control. The operation has not been committed. • This application process has performed an operation on this object under commitment control using a different commit definition. The operation has not been committed. • This application process has performed an operation on this object under commitment control. The operation has not been committed. The table cannot be altered until the changes are committed or rolled back.
Recovery Text:	Do one of the following and try the request again: • If your application process issued the uncommitted operation, either issue a COMMIT or ROLLBACK before attempting any other operations on this object, or issue the statement from a program using a commitment control level other than *NONE. • If the application process that issued the uncommitted operation on this object is not your application process, then that application process must perform a COMMIT or a ROLLBACK. • If your application process issued the uncommitted operation using a different commit definition, issue a COMMIT or ROLLBACK for that commit definition. • Issue either a COMMIT or ROLLBACK before attempting an ALTER TABLE statement on this object.
SQLCODE or SQLCODEs:	-910
SQLSTATE or SQLSTATEs:	57007

SQL0913	
Message Text:	Row or object &1 in &2 type *&3 in use.
Cause Text:	The requested object &1 in &2 type *&3 is either in use by another application process or a row in the object is in use by either another application process or another cursor in this application process.
Recovery Text:	Look at the previously listed messages in the job log (DSPJOBLOG command) or from interactive SQL press F10 (Display messages in job log) on this display to determine if this is an object or record lock wait time out. Do one of the following: • If the object is locked by another application process, try the SQL statement again when the object is not in use. Use the Work with Object Locks (WRKOBJLCK) command to determine who is currently using the object. • If the object is a schema and an attempt was made to create a table, view, or index into this schema under commitment control, a save-while-active operation may be in progress on the same schema by another job in the system. Try the request again when the save-while-active processing is complete. • If a record is locked by another application process, try the SQL statement again when the record is not in use. The Display Record Locks (DSPRCDLCK) command will determine who is currently using the record. • If this is a record lock held by another cursor in the same application process, you must issue a COMMIT, ROLLBACK, or another FETCH statement on the cursor that is holding the lock before issuing this SQL statement. If this error occurs frequently, use the Change Physical File (CHGPF), Change Logical File (CHGLF), or Override Data Base File (OVRDBF) command to change the object or record wait time out.
SQLCODE or SQLCODEs:	-913
SQLSTATE or SQLSTATEs:	57033

SQL0918	
Message Text:	ROLLBACK is required.
Cause Text:	The activation group requires a ROLLBACK to be performed prior to running any other SQL statements.
Recovery Text:	Issue a ROLLBACK CL command or an SQL ROLLBACK statement and then continue.
SQLCODE or SQLCODEs:	-918
SQLSTATE or SQLSTATEs:	51021

SQL0950	
Message Text:	Relational database &1 not in relational database directory.
Cause Text:	A request for relational database &1 was made. However the relational database name was not found in the relational database directory.
Recovery Text:	Do one of the following: • Change the name of the relational database specified on the CONNECT, SET CONNECTION, RELEASE, or DISCONNECT statement or the RDB parameter of the SQL precompile commands. • Add the relational database name to the relational database directory using the Add Relational Database Directory Entry (ADDRDBDIRE) command.

SQL0950	
SQLCODE or SQLCODEs:	-950
SQLSTATE or SQLSTATEs:	42705, 55006

SQL0951	
Message Text:	Object &1 in &2 not altered.
Cause Text:	Object &1 in &2 was not altered because it, or a related object, is being used by the same application process. A table may be related in a referential constraint relationship with another table being used by the same application process.
Recovery Text:	Close the cursor and try the alter request again.
SQLCODE or SQLCODEs:	-951
SQLSTATE or SQLSTATEs:	55007

SQL0952	
Message Text:	Processing of the SQL statement ended. Reason code &1.
Cause Text:	The SQL operation was ended before normal completion. The reason code is &1. Reason codes and their meanings are: • 1 - An SQLCancel API request has been processed, for example from ODBC. • 2 - SQL processing was ended by sending an exception. • 3 - Abnormal termination. • 4 - Activation group termination. • 5 - Reclaim activation group or reclaim resources. • 6 - Process termination. • 7 - An EXIT function was called. • 8 - Unhandled exception. • 9 - A Long Jump was processed. • 10 - A cancel reply to an inquiry message was received. • 11 - Open Database File Exit Program (QIBM_QDB_OPEN). • 0 - Unknown cause.
Recovery Text:	If the reason code is 1, a client request was made to cancel SQL processing. For all other reason codes, see previous messages to determine why SQL processing was ended.
SQLCODE or SQLCODEs:	-952
SQLSTATE or SQLSTATEs:	57014

SQL0969	
Message Text:	Error occurred while passing request to application requester driver program.
Cause Text:	An unexpected error occurred while passing the SQL request to the application requester driver program for relational database &1. See previously listed messages in the job log for the cause of the failure.
Recovery Text:	Correct any problems and try the request again.

SQL0969	
SQLCODE or SQLCODEs:	-969
SQLSTATE or SQLSTATEs:	58033

SQL0971	
Message Text:	Constraint &4 in check pending state.
Cause Text:	The operation being performed on table &2 in &3 failed. Constraint &4 in &5 could not be enforced because of reason code &1. The reason codes and their meanings are: • 1 - The dependent file is in check pending status due to a referential constraint violation. • 2 - The dependent or parent file's access path is not valid. • 3 - The file is in check pending status due to a check constraint violation.
Recovery Text:	For reason codes 1 and 3, use the CHGPFCST command to disable the constraint. Then use the DSPCPCST command to see the records causing the check pending status. Correct the data in the file and then use the CHGPFCST command to enable the constraint. For reason code 2, use the EDTRBDAP command or Manage Index Rebuilds in iSeries Navigator to rebuild the file's access path.
SQLCODE or SQLCODEs:	-971
SQLSTATE or SQLSTATEs:	57011

SQL0990	
Message Text:	Outcome unknown for the unit of work.
Cause Text:	The unit of work completed but the outcome is not fully known at all sites. Either a conversation failure occurred and resynchronization is occurring to correct the problem, or a ROLLBACK occurred at one of the resources.
Recovery Text:	No user action is necessary.
SQLCODE or SQLCODEs:	+990
SQLSTATE or SQLSTATEs:	01587

SQL1530	
Message Text:	SET CURRENT DEGREE statement not fully enabled.
Cause Text:	Parallel processing is not enabled on this machine because the system feature DB2 UDB Symmetric Multiprocessing is not installed on the system.
Recovery Text:	
SQLCODE or SQLCODEs:	+1530
SQLSTATE or SQLSTATEs:	01623

SQL1583	
Message Text:	PAGESIZE value &1 not correct.

SQL1583	
Cause Text:	The PAGESIZE value of &1 was not correct for one of the following reasons: • It cannot be specified for an ENCODED VECTOR index. • The valid values for PAGESIZE are 8, 16, 32, 64, 128, 256, and 512. • The PAGESIZE value may conflict with the key length of the index.
Recovery Text:	Change the PAGESIZE value to one of the allowed values. Make sure this is not an ENCODED VECTOR index. Try the request again.
SQLCODE or SQLCODEs:	-1583
SQLSTATE or SQLSTATEs:	428DE

SQL1596	
Message Text:	WITH EMPTY TABLE not valid for &1 in &2.
Cause Text:	The ALTER TABLE cannot have the WITH EMPTY TABLE clause specified for table &1 in &2. The table is either a materialized query table or the parent table in a referential constraint which conflicts with the WITH EMPTY TABLE clause.
Recovery Text:	Change the name of the table and try the request again.
SQLCODE or SQLCODEs:	-1596
SQLSTATE or SQLSTATEs:	42928

SQL4300	
Message Text:	Java support is not installed or properly configured on this platform.
Cause Text:	Support for Java stored procedures and user-defined functions is not installed and configured on this server.
Recovery Text:	Ensure that a compatible Java Development Kit is installed.
SQLCODE or SQLCODEs:	-4300
SQLSTATE or SQLSTATEs:	42724

SQL4301	
Message Text:	Java interpreter startup or communication failed for reason code &1.
Cause Text:	An error occurred while attempting to start or communicate with a Java interpreter. The reason codes and their meanings follow: • 1 - Java environment variables or Java database configuration parameters are invalid. • 2 - A Java Native Interface call to the Java interpreter failed. • 4 - The Java interpreter has terminated itself and cannot be restarted.
Recovery Text:	Ensure that Java environment variables or Java database configuration parameters are valid. Ensure that a Java method called by the Java interpreter does not use System.out. Ensure that internal DB2 classes (com.ibm.db2) are not overridden by user classes.
SQLCODE or SQLCODEs:	-4301
SQLSTATE or SQLSTATEs:	58004

SQL4302	
Message Text:	Java stored procedure or user-defined function &1, specific name &2 aborted with an exception ″&3″.
Cause Text:	The Java stored procedure or user-defined function aborted with a Java exception. If SQJAVA component trace is on, then the component trace for the job contains a Java stack traceback for the aborted method.
Recovery Text:	Debug the Java method to eliminate the exception.
SQLCODE or SQLCODEs:	-4302
SQLSTATE or SQLSTATEs:	38501

SQL4303	
Message Text:	Java stored procedure or user-defined function &1, specific name &2, could not be identified from external name &3.
Cause Text:	The CREATE PROCEDURE or CREATE FUNCTION statement that declared this stored procedure or user-defined function had a badly formatted EXTERNAL NAME clause. The external name must be formatted as follows: ″package.subpackage.class.method″.
Recovery Text:	Submit a corrected CREATE PROCEDURE or CREATE FUNCTION statement.
SQLCODE or SQLCODEs:	-4303
SQLSTATE or SQLSTATEs:	42724

SQL4304	
Message Text:	Java stored procedure or user-defined function &1, specific name &2 could not load Java class &3 for reason code &4.
Cause Text:	The Java class given by the EXTERNAL NAME clause of a CREATE PROCEDURE or CREATE FUNCTION statement could not be loaded. The reason codes and their meanings are: • 1 - The class was not found on the CLASSPATH. • 2 - The class did not implement the required interface (″com.ibm.db2.app.StoredProc″ or ″com.ibm.db2.app.UDF″) or lacked the Java ″public″ access flag. • 3 - The default constructor failed or was unavailable.
Recovery Text:	Ensure that the compiled ″.class″ file is installed in the CLASSPATH, for example under ″/QIBM/UserData/OS400/SQLLib/Function″. Ensure it implements the required Java interfaces and is ″public″.
SQLCODE or SQLCODEs:	-4304
SQLSTATE or SQLSTATEs:	42724

SQL4306	
Message Text:	Java stored procedure or user-defined function &1, specific name &2 could not call Java method &3, signature &4.
Cause Text:	The Java method given by the EXTERNAL NAME clause of a CREATE PROCEDURE or CREATE FUNCTION statement could not be found. Its declared argument list may not match what the database expects, or it may not be a ″public″ instance method.

SQL4306	
Recovery Text:	Ensure that a Java instance method exists with the ″public″ flag and the expected argument list for this call.
SQLCODE or SQLCODEs:	-4306
SQLSTATE or SQLSTATEs:	42724

SQL4701	
Message Text:	Too many partitions specified.
Cause Text:	One of the following has occurred: • There are too many partitions specified on the CREATE TABLE statement. • There was an attempt to add a partition to a existing table with an ALTER TABLE statement but the table already has the maximum number of partitions. There is a maximum of 256 partitions allowed in a partitioned table.
Recovery Text:	For a CREATE TABLE statement, correct the number of partitions so that the number does not exceed the maximum. For an ALTER TABLE statement, remove the ADD PART clause. Try the request again.
SQLCODE or SQLCODEs:	-4701
SQLSTATE or SQLSTATEs:	54054

SQL5001	
Message Text:	Column qualifier or table &2 undefined.
Cause Text:	Name &2 was used to qualify a column name or was specified as the operand of the RRN, HASHED_VALUE, PARTITION, NODENAME, NODENUMBER, DBPARTITIONNAME, DBPARTITIONNUM, DATAPARTITIONNAME, or DATAPARTITIONNUM scalar function. The name is not defined to be a table designator in this SQL statement or the table designator cannot be referenced where it is specified in the SQL statement. If a correlation name is specified following the table name in a FROM clause, the correlation name is considered to be the table designator. If a correlation name is not specified, the table name is considered to be the table designator. If using SQL naming and the table is qualified with authorization name, then the table designator is authorization-name.table-name. If the authorization name is not specified, the table designator is the implicit authorization name followed by the table name. Correlation from a nested table expression to a higher level table is only allowed if the TABLE keyword is used for the definition of the nested table expression. If the name is *N, a lateral correlation reference from a nested table expression is not allowed. You can not correlate to a table at a higher level than the nested table expression for one of the following reasons: • The nested table expression contains a UNION, EXCEPT, or INTERSECT. • The nested table expression uses the DISTINCT keyword in the SELECT clause. • The nested table expression contains an ORDER BY clause. • The correlated provider is in the same FROM clause as the nested table expression but is part of a RIGHT OUTER JOIN or RIGHT EXCEPTION JOIN. • The nested table expression is in the FROM clause of another nested table expression that contains one of these restrictions. In an OLAP function, the ORDER OF table designator must refer to a table designator in the FROM clause of the subselect.
Recovery Text:	Ensure all column names are qualified with a valid table designator. Make sure a table designator is specified as the argument to the function. Use the TABLE keyword to allow correlated columns within a nested table expression. Try the request again.

SQL5001	
SQLCODE or SQLCODEs:	-5001
SQLSTATE or SQLSTATEs:	42703

SQL5003	
Message Text:	Cannot perform operation under commitment control.
Cause Text:	The following operations cannot be performed under commitment control with COMMIT(*CHG), COMMIT(*CS), or COMMIT(*ALL) specified: • DROP SCHEMA statement. • GRANT or REVOKE statement to an object that has an authority holder. • CREATE statement in SQL naming mode of an object that has an authority holder. These operations cannot be committed or rolled back.
Recovery Text:	Specify COMMIT(*NONE), and try the statement again.
SQLCODE or SQLCODEs:	-5003
SQLSTATE or SQLSTATEs:	42922

SQL5005	
Message Text:	Operator &4 not consistent with operands.
Cause Text:	The operator specified is not consistent with the previous operands. The arithmetic operators (*, /, and **) are not valid with concatenation operators or with the DIGITS and SUBSTR scalar functions. The concatenation operator is not valid with other operations or functions that result in a numeric value, such as the arithmetic operators (* and /) or the LENGTH, DECIMAL, FLOAT, or INTEGER scalar functions.
Recovery Text:	Change the SQL statement so all expressions are valid numeric expressions, string expressions, or date/time expressions.
SQLCODE or SQLCODEs:	-5005
SQLSTATE or SQLSTATEs:	42815

SQL5012	
Message Text:	Host variable &1 not numeric with zero scale.
Cause Text:	Host variable &1 was specified in a position where it is not allowed. The host variable was not usable for one of the following reasons: • It is not numeric. • The scale is not zero.
Recovery Text:	Change the host variable to a numeric type with zero scale.
SQLCODE or SQLCODEs:	-5012
SQLSTATE or SQLSTATEs:	42618

SQL5016	
Message Text:	Qualified object name &1 not valid.
Cause Text:	One of the following has occurred: • The syntax used for the qualified object name is not valid for the naming option specified. With system naming, the qualified form of an object name is schema-name/object-name. With SQL naming the qualified form of an object name is authorization-name.object-name. • The syntax used for the qualified object name is not allowed. User-defined types cannot be qualified with the schema in the system naming convention on parameters and SQL variables of an SQL procedure or function.
Recovery Text:	Do one of the following and try the request again: • If you want to use the SQL naming convention, verify the SQL naming option in the appropriate SQL command and qualify the object names in the form authorization-id.object-name. • If you want to use the system naming convention, specify the system naming option in the appropriate SQL command and qualify the object names in the form schema-name/object-name. • With the system naming convention, ensure the user-defined types specified for parameters and variables in an SQL routine can be found in the current path.
SQLCODE or SQLCODEs:	-5016
SQLSTATE or SQLSTATEs:	42833

SQL5017	
Message Text:	Too many users specified for GRANT or REVOKE.
Cause Text:	More than the maximum of 50 users are specified on the GRANT or REVOKE statement.
Recovery Text:	Change the GRANT or REVOKE statement to specify a maximum 50 users. Try the request again.
SQLCODE or SQLCODEs:	-5017
SQLSTATE or SQLSTATEs:	54009

SQL5021	
Message Text:	FOR UPDATE column &1 not valid.
Cause Text:	One of the following has occurred: • A column in the FOR UPDATE clause is specified in the ORDER BY clause. If the column name is *N, a list of columns was not specified in the FOR UPDATE clause. This is the same as listing all columns. • A DATALINK column in the FOR UPDATE clause is specified in the SELECT list.
Recovery Text:	Remove the duplicate column from one of the clauses. If no columns were specified in the FOR UPDATE clause, remove either the FOR UPDATE clause or the ORDER BY clause. Try the request again.
SQLCODE or SQLCODEs:	-5021
SQLSTATE or SQLSTATEs:	42930

SQL5023	
Message Text:	Statement name &1 previously referred to.
Cause Text:	The statement name &1 referred to in this DECLARE CURSOR statement has already been referred to in a previous DECLARE CURSOR. A statement name can only be associated with one cursor.
Recovery Text:	Check the statement names specified on all DECLARE CURSOR statements in an application program or REXX procedure to make sure they are unique. Try the request again.
SQLCODE or SQLCODEs:	-5023
SQLSTATE or SQLSTATEs:	26510

SQL5024	
Message Text:	Host variable &1 not character, UCS-2 graphic, or UTF-16 graphic.
Cause Text:	Host variable &1 is not defined as character, UCS-2 graphic, or UTF-16 graphic. Host variables in a precompiled program or REXX procedure must be character, UCS-2 graphic, or UTF-16 graphic if used as: • The statement string in a PREPARE or EXECUTE IMMEDIATE statement. • The table name in a DESCRIBE TABLE statement. • The procedure name in a CALL statement. • The server name, authorization name, or password in a CONNECT, SET CONNECTION, RELEASE, DISCONNECT, SET ENCRYPTION PASSWORD, or SET SESSION AUTHORIZATION statement. • The special register value in a SET special register statement such as SET PATH or SET SCHEMA. • The SQLSTATE value or signal information value in a SIGNAL statement. • A descriptor name.
Recovery Text:	Specify a host variable that is character, UCS-2 graphic, or UTF-16 graphic. Try the request again.
SQLCODE or SQLCODEs:	-5024
SQLSTATE or SQLSTATEs:	42618

SQL5027	
Message Text:	OPTION(*SYS) valid only if relational database &1 (product identification &2) is an iSeries.
Cause Text:	SQL naming is required when the create of a SQL package is to a relational database that is not another iSeries.
Recovery Text:	Change the program to use SQL naming, then retry the CRTSQLxxx command with the parameter OPTION(*SQL).
SQLCODE or SQLCODEs:	-5027
SQLSTATE or SQLSTATEs:	560C4

SQL5028	
Message Text:	COMMIT(*NONE) valid only if relational database &1 (product identification &2) is an iSeries.
Cause Text:	Commit level of *CHG, *CS or *ALL required when relational database &1 is not another iSeries.
Recovery Text:	Change the program to use commitment control, then retry the CRTSQLxxx command with a commitment control level of *CHG, *CS, or *ALL specified.
SQLCODE or SQLCODEs:	-5028
SQLSTATE or SQLSTATEs:	560C4

SQL5047	
Message Text:	Error processing SRTSEQ or LANGID parameter. Message is &3, &4.
Cause Text:	An error occurred during an attempt to retrieve the sort sequence table for the SRTSEQ parameter &1 and LANGID parameter &2. Message &3 was returned.
Recovery Text:	Correct the errors as indicated by message &3 and issue the request again. If a sort sequence table is not required, specify *HEX as the SRTSEQ parameter.
SQLCODE or SQLCODEs:	-5047
SQLSTATE or SQLSTATEs:	42616

SQL5051	
Message Text:	Qualifier &1 not same as name &2.
Cause Text:	One of the following has occurred: • An object created in a CREATE SCHEMA statement is qualified by a name other than the schema name. All objects created in a CREATE SCHEMA statement must be either qualified by the schema name &2 or not qualified. Unqualified objects are implicitly qualified by the schema name. • A constraint name was qualified by a name that is not the same as the qualifier for the table. A constraint for a table must be qualified by the same schema as the table. If not explicitly qualified, a constraint name is implicitly qualified by the default schema, if one is specified. Otherwise, the constraint name is implicitly qualified by the authorization ID for SQL names and by the qualifier of the table name for system names.
Recovery Text:	Do one of the following and try the request again: • Explicitly qualify the object in the schema with &2 or remove qualifier &1 from the object name. • Use the same qualification for constraint names and table names.
SQLCODE or SQLCODEs:	-5051
SQLSTATE or SQLSTATEs:	42875

SQL7001	
Message Text:	Table &1 in &2 not database file.
Cause Text:	SQL processing is only valid for a database file. All other file types are not allowed.
Recovery Text:	Make certain that the table and schema names are correct.

SQL7001	
SQLCODE or SQLCODEs:	-7001
SQLSTATE or SQLSTATEs:	42858

SQL7002	
Message Text:	Override parameter not valid.
Cause Text:	An Override Data Base File (OVRDBF) command was issued for one of the files referenced in the SQL statement. A parameter on the OVRDBF command is not valid for SQL. See message CPF4276 in the job log for information about which parameter is not valid.
Recovery Text:	Delete the override (DLTOVR command). Use the OVRDBF command again without the parameter that is not valid, if necessary, and then try the operation again.
SQLCODE or SQLCODEs:	-7002
SQLSTATE or SQLSTATEs:	42847

SQL7003	
Message Text:	File &1 in &2 has more than one format.
Cause Text:	SQL cannot process a file unless it has only one format.
Recovery Text:	Make certain that the correct filename was specified. Try the request again.
SQLCODE or SQLCODEs:	-7003
SQLSTATE or SQLSTATEs:	42857

SQL7006	
Message Text:	Cannot drop schema &1.
Cause Text:	&1 is a schema that is in the library list.
Recovery Text:	Remove &1 from the library list before attempting the DROP. Try the request again.
SQLCODE or SQLCODEs:	-7006
SQLSTATE or SQLSTATEs:	55018

SQL7007	
Message Text:	COMMIT, ROLLBACK, or SAVEPOINT not valid.
Cause Text:	A COMMIT, ROLLBACK, or SAVEPOINT statement was issued, but commitment control is not active.
Recovery Text:	Change the commitment control level *NONE to *CHG, *CS, or *ALL. The SET TRANSACTION statement can be used to change the isolation level to something other than *NONE. Try the request again.
SQLCODE or SQLCODEs:	-7007
SQLSTATE or SQLSTATEs:	51009

SQL7008	
Message Text:	&1 in &2 not valid for operation.
Cause Text:	The reason code is &3. Reason codes are: • 1 - &1 has no members. • 2 - &1 has been saved with storage free. • 3 - &1 not journaled, or no authority to the journal. Files with an RI constraint action of CASCADE, SET NULL, or SET DEFAULT must be journaled to the same journal. • 4 and 5 - &1 is in or being created into production library but the user has debug mode UPDPROD(*NO). • 6 - Schema being created, but user in debug mode with UPDPROD(*NO). • 7 - A based-on table used in creation of a view is not valid. Either the table is program described table or it is in a temporary schema. • 8 - Based-on table resides in a different ASP than ASP of object being created. • 9 - Index is currently held or is not valid. • 10 - A constraint or trigger is being added to an invalid type of table, or the maximum number of triggers has been reached, or all nodes of the distributed table are not at the same release level. • 11 - Distributed table is being created in schema QTEMP, or a view is being created over more than one distributed table. • 12 - Table could not be created in QTEMP, QSYS, QSYS2, or SYSIBM because it contains a column of type DATALINK having the FILE LINK CONTROL option. • 13 - The table contains a DATALINK column or a LOB column that conflicts with the data dictionary. • 14 - A DATALINK, LOB, or IDENTITY column cannot be added to a non SQL table. • 15 - Attempted to create or change an object using a commitment definition in a different ASP. • 16 - Sequence &1 in &2 was incorrectly modified with a CL command. • 17 - The table is not usable because it contains partial transactions.
Recovery Text:	Do one of the following based on the reason code: • 1 - Add a member to &1 (ADDPFM). • 2 - Restore &1 (RSTOBJ). • 3 - Start journaling on &1 (STRJRNPF), or get access to the journal. • 4, 5, or 6 - Perform a CHGDBG command with UPDPROD(*YES). • 7 - Remove table names which identify files in QTEMP or program described files. • 8 - Use tables in the same ASP. • 9 - Use the EDTRBDAP command to change the sequence of the access path from HELD to 1-99 or *OPN, or rebuild or delete the unique index or constraint. • 10 - Specify tables that are valid for constraints or triggers. • 11 - Specify a schema other than QTEMP, or create the view over only one distributed table. • 12 - Specify a schema other than QTEMP, QSYS, QSYS2, or SYSIBM. • 13 - Specify a schema that does not contain a data dictionary or remove all DATALINK and LOB columns. • 14 - Specify an SQL table. • 15 - Specify an object in the same ASP as the current commitment definition or end the current commitment definition. • 16 - Specify a different sequence, or delete the data area associated with sequence &1 in &2 and re-create the sequence. • 17 - See previous message in the job log.

SQL7008	
SQLCODE or SQLCODEs:	-7008
SQLSTATE or SQLSTATEs:	55019

SQL7010	
Message Text:	Logical file &1 in &2 not valid for CREATE VIEW.
Cause Text:	Logical file &1 in &2 is specified in the subselect clause of a CREATE VIEW. Views cannot be created over logical files.
Recovery Text:	Remove logical file &1 from the CREATE VIEW statement and try the request again.
SQLCODE or SQLCODEs:	-7010
SQLSTATE or SQLSTATEs:	42850

SQL7011	
Message Text:	&1 in &2 not table, view, or physical file.
Cause Text:	The SQL statement &3 cannot be performed on a file which is not a table, view, single format logical file, or physical file.
Recovery Text:	Do one of the following: • Use a control language (CL) command to do the function. • Select the correct table, view, logical, or physical file.
SQLCODE or SQLCODEs:	-7011
SQLSTATE or SQLSTATEs:	42851

SQL7017	
Message Text:	Unable to run statement with specified commit level.
Cause Text:	SQL is unable to run the statement with the specified commit level because SQL cannot register a resource with commitment control.
Recovery Text:	See previous messages for more information.
SQLCODE or SQLCODEs:	-7017
SQLSTATE or SQLSTATEs:	42971

SQL7018	
Message Text:	COMMIT HOLD or ROLLBACK HOLD not allowed.
Cause Text:	COMMIT HOLD or ROLLBACK HOLD was attempted to an application server or from an application requester that is not an iSeries. HOLD is only allowed when the application requester and the application server are both iSeries.
Recovery Text:	Remove HOLD and submit the statement again.
SQLCODE or SQLCODEs:	-7018

SQL7018	
SQLSTATE or SQLSTATEs:	42970

SQL7020	
Message Text:	SQL package creation failed.
Cause Text:	An SQL package must exist on the application server to run SQL statements. Creation of a package for this purpose failed on the application server and returned SQLCODE &1, SQLSTATE &2. The name of the package being created was &3 in schema &4 on relational database &5.
Recovery Text:	Do the response as indicated for the SQLCODE &1 and SQLSTATE &2. If you are running with commitment control, exit Interactive SQL or Query Manager and issue a CL COMMIT or ROLLBACK command. This will enable you to continue processing at the local system.
SQLCODE or SQLCODEs:	-7020
SQLSTATE or SQLSTATEs:	42969

SQL7021	
Message Text:	Local program attempting to run on application server.
Cause Text:	An attempt was made to run an SQL program in a process that is an application server.
Recovery Text:	Initiate another job and run the SQL program in that job.
SQLCODE or SQLCODEs:	-7021
SQLSTATE or SQLSTATEs:	57043

SQL7022	
Message Text:	User &1 not the same as current user &2.
Cause Text:	One of the following occurred: • User &1 was specified in a CONNECT statement that specified the local relational database name. The user specified is not the same as the current job user &2. • User &1 was specified in a CONNECT statement and a connection using &2 already exists to the specified relational database using connection method *DUW.
Recovery Text:	If connecting to the local relational database, change the statement so the user specified on the CONNECT is the same as the current job user ID. If connecting to a remote relational database, either use the SET CONNECTION statement to use the existing connection or end the current connection and issue the CONNECT statement with the new user id.
SQLCODE or SQLCODEs:	-7022
SQLSTATE or SQLSTATEs:	42977

SQL7024	
Message Text:	CCSIDs are not compatible.

SQL7024	
Cause Text:	An attempt was made to create an index or to group columns, but the columns have incompatible CCSIDs. The sort sequence is not *HEX. If the statement is CREATE INDEX, index &1 in schema &2 was not created. The CCSIDs of character key columns or character columns in a GROUP BY clause must be associated CCSIDs when the sort sequence is not *HEX. Associated CCSIDs all have the same single-byte code page.
Recovery Text:	Do one of the following: • Change the sort sequence to *HEX and try the statement again. • Change the CCSIDs of the columns so that they are either 65535 or associated CCSIDs and try the statement again.
SQLCODE or SQLCODEs:	-7024
SQLSTATE or SQLSTATEs:	42876

SQL7026	
Message Text:	Auxiliary storage pool &4 or &5 not found.
Cause Text:	Object &1 in &2 type *&3 was not created because auxiliary storage pool (ASP) &4 or device name &5 does not exist on the system or in the ASP group of the thread.
Recovery Text:	Specify a correct ASP number or ASP device name and try the request again.
SQLCODE or SQLCODEs:	-7026
SQLSTATE or SQLSTATEs:	42896

SQL7027	
Message Text:	Cannot GRANT specified privileges on object &1 in &2 type *&3.
Cause Text:	A GRANT operation was attempted on view &1 in &2 type *&3. This operation cannot be performed because it would give the user specified additional privileges to the underlying file &4 in &5. The user has *OBJOPR or *OBJMGT system privileges to the underlying file.
Recovery Text:	One of the following may be done: • Grant the specified privileges to &4 • Obtain the required authority from either the security officer or the object owner • Delete the current authority to &4. Try the operation again.
SQLCODE or SQLCODEs:	-7027
SQLSTATE or SQLSTATEs:	42984

SQL7028	
Message Text:	Owner and primary group cannot be the same.
Cause Text:	While attempting to create an object, SQL attempted to change the owner of the object. The new owner was the same as the primary group for that object. This is not allowed.
Recovery Text:	Change the primary group for the user profile and try the request again.
SQLCODE or SQLCODEs:	-7028
SQLSTATE or SQLSTATEs:	42944

SQL7029	
Message Text:	New name &3 is not valid.
Cause Text:	An attempt was made to rename &1 in &2, but the new name is not valid. Both the new name and the new system name are valid system names. If both names are specified in the RENAME statement, only the name following SYSTEM NAME can be a valid system name.
Recovery Text:	Do one of the following and try the request again: • Specify either the new name or the new system name. • Change the first name to be a name that is not valid as a system name.
SQLCODE or SQLCODEs:	-7029
SQLSTATE or SQLSTATEs:	428B8

SQL7030	
Message Text:	Alias &1 for table &2 in &3 not valid for statement.
Cause Text:	The SQL statement cannot be performed on alias &1 because the alias refers to a member of table &2 in schema &3.
Recovery Text:	Specify a valid table or an alias that does not refer to a member. Try the request again.
SQLCODE or SQLCODEs:	-7030
SQLSTATE or SQLSTATEs:	55042

SQL7031	
Message Text:	Sort sequence table &1 too long.
Cause Text:	Sort sequence table &1 in &2 is a UCS-2 sort sequence table that is greater than 31560 bytes long. It cannot be used with Distributed Relational Database Architecture (DRDA).
Recovery Text:	Specify a different sort sequence table to be used with Distributed Relational Database Architecture (DRDA).
SQLCODE or SQLCODEs:	-7031
SQLSTATE or SQLSTATEs:	54044

SQL7032	
Message Text:	SQL procedure, function, or trigger &1 in &2 not created.
Cause Text:	SQL procedure, function, or trigger &1 in &2 was not created. The compile was not successful. SQL creates an SQL procedure, function, or trigger as a C program that contains embedded SQL. Errors not found during the initial parsing of the CREATE PROCEDURE, ALTER PROCEDURE, CREATE FUNCTION, or CREATE TRIGGER statement can be found during the precompile.
Recovery Text:	If a compile error occurred, see the appropriate listing in QSYSPRT. If the SQL precompile failed, there is always a listing with the error. If the C compile failed, the listing is only created if requested. Specify SET OPTION OUTPUT=*PRINT prior to the routine body in the CREATE statement if listings are required.
SQLCODE or SQLCODEs:	-7032

SQL7032	
SQLSTATE or SQLSTATEs:	42904

SQL7033	
Message Text:	Alias name &1 in &2 not allowed.
Cause Text:	Alias name &1 cannot be used. This program was compiled on a release before alias names were supported. The table name it is using has now been defined as an alias name.
Recovery Text:	Recompile the program on a release that supports alias names or remove the alias from the system.
SQLCODE or SQLCODEs:	-7033
SQLSTATE or SQLSTATEs:	42923

SQL7034	
Message Text:	LOB locators are not allowed with COMMIT(*NONE).
Cause Text:	A LOB locator cannot be used with commitment control level of *NONE or *NC.
Recovery Text:	Use a commitment control level of *CHG, *UR, *CS, *ALL, *RS, or *RR.
SQLCODE or SQLCODEs:	-7034
SQLSTATE or SQLSTATEs:	42926

SQL7036	
Message Text:	System User-Defined Type name used in SQLDA.
Cause Text:	A User-Defined Type name returned in the extended SQLVAR entry of the SQLDA is longer than 19 characters. Since there is not room in the SQLNAME entry, the system name is substituted instead. The system name is 10 characters.
Recovery Text:	
SQLCODE or SQLCODEs:	+7036
SQLSTATE or SQLSTATEs:	01634

SQL7037	
Message Text:	Data in a distributed file &1 in &2 cannot be redistributed.
Cause Text:	An attempt was made to change the node group, partitioning file, partitioning key, or an attribute of a partitioning key. These changes can cause data to be redistributed, but data in file &1 in &2 cannot be redistributed because it contains a DataLink with FILE LINK CONTROL.
Recovery Text:	Change the request so that data will not be redistributed, and try the function again.
SQLCODE or SQLCODEs:	-7037
SQLSTATE or SQLSTATEs:	429B6

SQL7038	
Message Text:	Delete cascade not valid for &1 in &2.
Cause Text:	A delete cascade rule can not be added to &1 in &2 as it contains a DataLink column.
Recovery Text:	Either remove the DataLink column or remove the specified delete cascade rule.
SQLCODE or SQLCODEs:	-7038
SQLSTATE or SQLSTATEs:	429B7

SQL7048	
Message Text:	Operation not allowed because trigger is invalid.
Cause Text:	An open of an SQL table or view failed as a result of inoperative trigger &1 in schema &2. The open could be due to an insert, update, delete, or open cursor statement.
Recovery Text:	Drop and re-create trigger &1 in &2. See the previous CPF418A escape message for the name of the file that failed to open. Preceding the CPF418A will be CPD502B messages for each invalid trigger found. The SYSTRIGGERS catalog can also be used to determine the SQL table an invalid trigger is attached to.
SQLCODE or SQLCODEs:	-7048
SQLSTATE or SQLSTATEs:	51037

SQL7049	
Message Text:	An internal object limit has been exceeded.
Cause Text:	The SQL statement cannot be run successfully because an internal object limitation has been encountered. The reason code is &1. The operation failed for one of the following reasons: 1 - The maximum number of stored procedures with open result sets has been exceeded. 2 - An internal space limitation has been reached while processing result sets. 3 - A limit has been reached creating the C program for an SQL procedure, function, or trigger. 4 - An internal limit on SQL program or package size has been reached. &2 in &3 type *&4 reached the limit.
Recovery Text:	The reason codes and their recovery follow: 1 - Either fetch a result set or close an open result set for a stored procedure. 2 - Fetch or close one or more open result sets, or reduce the size of the array result set. 3 - Modify the SQL routine by reducing the number of SQL statements. 4 - Reduce the number of SQL statements in the program or package or move some SQL statements to a different module, and recompile the program or package.
SQLCODE or SQLCODEs:	-7049
SQLSTATE or SQLSTATEs:	54035

SQL7050	
Message Text:	Result sets are not available from procedure &1 in &2.
Cause Text:	An SQL CALL statement was performed for procedure &1 in &2. It opened &3 result sets, but all were closed before they could be processed. This can be caused by the ending of an activation group, or by some other function that closes SQL cursors, such as a Reclaim Resources (RCLRSC) CL command. This can also happen if the procedure contains a SET RESULT SETS statement, and an error occurred on that statement.

SQL7050	
Recovery Text:	If the procedure was created with *NEW as the activation group, change it to *CALLER or a named activation group. Remove any functions that might be closing SQL cursors. If the procedure contains the SET RESULT SETS statement, make sure it completes successfully and does not identify cursors that are not open.
SQLCODE or SQLCODEs:	+7050
SQLSTATE or SQLSTATEs:	01646

SQL7051	
Message Text:	MODE DB2SQL before trigger converted to MODE DB2ROW.
Cause Text:	MODE DB2SQL before triggers are not supported. The SQL trigger &1 in &2 will be converted from MODE DB2SQL to MODE DB2ROW.
Recovery Text:	MODE DB2ROW should be specified for all BEFORE triggers. Change the statement and try the request again.
SQLCODE or SQLCODEs:	+7051
SQLSTATE or SQLSTATEs:	01647

SQL7052	
Message Text:	Object &1 type *&3 cannot be created in &2.
Cause Text:	An attempt to create object &1 type *&3 in &2 failed because the library is not valid for user objects.
Recovery Text:	Perform the operation using a different library.
SQLCODE or SQLCODEs:	-7052
SQLSTATE or SQLSTATEs:	55050

SQL7053	
Message Text:	Relational database directory not available.
Cause Text:	An attempt to connect to relational database (RDB) &1 failed because the RDB directory is not available. A vary on operation may be in progress for the Auxiliary Storage Pool (ASP) devices.
Recovery Text:	If a vary on operation is in progress for any ASP devices, wait until the operation is complete. If repeated attempts fail, you may need to run the command RCLSTG SELECT(*DBXREF).
SQLCODE or SQLCODEs:	-7053
SQLSTATE or SQLSTATEs:	57011

SQL7054	
Message Text:	The open failed due to exit program &1.

SQL7054	
Cause Text:	An exit program was installed to examine all file opens using the ADDEXITPGM EXITPNT(QIBM_QDB_OPEN) FORMAT(DBOP0100) command. The exit program ended this open request.
Recovery Text:	
SQLCODE or SQLCODEs:	-7054
SQLSTATE or SQLSTATEs:	58002

SQL7055	
Message Text:	Cursor &1 is ambiguous.
Cause Text:	Cursor &1 will become ambiguous because there is another cursor with the same name already opened from a different invocation of the procedure.
Recovery Text:	Close all other cursors with the same name opened in other invocations of the procedure or include the cursor in the set of result sets specified on this SET RESULT SETS statement. Try the request again.
SQLCODE or SQLCODEs:	-7055
SQLSTATE or SQLSTATEs:	24502

SQL7905	
Message Text:	Table &1 in &2 created but was not journaled.
Cause Text:	The table &1 was created in &2, but the table was not journaled. Either a QDFTJRN data area indicated that journaling should not be started, the default journal did not exist, or the default journal did exist but the journaling could not be started. The default journal may be damaged, unable to accept journal entries, or may not exist. The SQL name for the table is &1 and the system name for the table is &3. For a distributed table, journaling could not be started on all of the systems. The default journal must exist, and be able to accept journal entries, on each of the systems in the node group.
Recovery Text:	The table was created, but until the table is journaled, COMMIT(*CHG), COMMIT(*CS), COMMIT(*RS), COMMIT(*RR), COMMIT(*UR), and COMMIT(*ALL) will not be allowed for table &1. If the default journal does not exist, create the journal (CRTJRN command) and start journaling (STRJRNPF command). If the journal is in error, correct the problem to the journal If the table is distributed, correct the problems on all the systems in the node group and start journaling (STRJRNPF command).
SQLCODE or SQLCODEs:	+7905
SQLSTATE or SQLSTATEs:	01567

SQL7909	
Message Text:	Routine &1 was created, but cannot be saved and restored.

SQL7909	
Cause Text:	The routine &1 was created successfully in &2 with a specific name of &3, but the routine's attributes could not be saved in the associated program or service program object. If the *PGM or *SRVPGM object is saved and then restored, the SQL catalogs will not be updated with the attributes for this routine. Reason code is &4. Reason codes and their meanings are: • 1 - The external program did not exist when the CREATE statement was issued. • 2 - The external program schema is QSYS. • 3 - The external program was not an ILE *PGM or *SRVPGM. • 4 - The external program was in use by another job. • 5 - The SQL associated space in the external program was in use by another job. • 6 - The SQL associated space in the external program could not be expanded. • 7 - The external program was compiled in a release prior to V4R4M0. • 8 - The SQL associated space in the external program already contains the maximum number of routine definitions.
Recovery Text:	Do one of the following based on the reason code: • 1 - Ensure that the external program exists when the CREATE statement is issued. • 2 - Ensure that the external program schema is not QSYS. • 3 - Ensure that the external program is an ILE *PGM or *SRVPGM. • 4 - Use WRKOBJLCK to ensure that the external program is available when the routine is created. • 5 - Ensure that the external program is available when the routine is created. • 6 - Try recompiling the external program to rebuild the program's associated space. • 7 - Recompile the external program in a more recent release. • 8 - Drop one of the routines currently defined for the external program.
SQLCODE or SQLCODEs:	+7909
SQLSTATE or SQLSTATEs:	01660

SQL7941	
Message Text:	Application process not at a commit boundary.
Cause Text:	A commitment control level other than *NONE was specified on the RUNSQLSTM command, but the application process is not at a commit boundary.
Recovery Text:	Issue a COMMIT or ROLLBACK to get to a commit boundary, or specify COMMIT(*NONE) on the RUNSQLSTM command.
SQLCODE or SQLCODEs:	-7941
SQLSTATE or SQLSTATEs:	42981

Related concepts

"SQL message concepts" on page 23
SQL messages are displayed when a DB2 Universal Database for iSeries returns an error or warning code to the application that uses it.

Code license and disclaimer information

IBM grants you a nonexclusive copyright license to use all programming code examples from which you can generate similar function tailored to your own specific needs.

| SUBJECT TO ANY STATUTORY WARRANTIES WHICH CANNOT BE EXCLUDED, IBM, ITS
| PROGRAM DEVELOPERS AND SUPPLIERS MAKE NO WARRANTIES OR CONDITIONS EITHER
| EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OR
| CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND
| NON-INFRINGEMENT, REGARDING THE PROGRAM OR TECHNICAL SUPPORT, IF ANY.

| UNDER NO CIRCUMSTANCES IS IBM, ITS PROGRAM DEVELOPERS OR SUPPLIERS LIABLE FOR
| ANY OF THE FOLLOWING, EVEN IF INFORMED OF THEIR POSSIBILITY:

| 1. LOSS OF, OR DAMAGE TO, DATA;
| 2. DIRECT, SPECIAL, INCIDENTAL, OR INDIRECT DAMAGES, OR FOR ANY ECONOMIC
| CONSEQUENTIAL DAMAGES; OR
| 3. LOST PROFITS, BUSINESS, REVENUE, GOODWILL, OR ANTICIPATED SAVINGS.

| SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF DIRECT,
| INCIDENTAL, OR CONSEQUENTIAL DAMAGES, SO SOME OR ALL OF THE ABOVE LIMITATIONS
| OR EXCLUSIONS MAY NOT APPLY TO YOU.

Appendix. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, IBM License Agreement for Machine Code, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming Interface Information

This SQL messages and codes publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of IBM i5/OS.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

| DB2
| DB2 Universal Database
| Distributed Relational Database Architecture
| DRDA
| IBM
| IBM (logo)
| e(logo)server
| iSeries
| OS/390
| z/OS

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Terms and conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

Personal Use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

Commercial Use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED ″AS-IS″ AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Printed in USA