# CA-IDMS®

## SQL Quick Reference
## 15.0

Computer Associates™

This documentation and related computer software program (hereinafter referred to as the “Documentation”) is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. (“CA”) at any time.

This documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of CA. This documentation is proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this documentation for their own internal use, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to return to CA the reproduced copies or to certify to CA that same have been destroyed.

To the extent permitted by applicable law, CA provides this documentation “as is” without warranty of any kind, including without limitation, any implied warranties of merchantability, fitness for a particular purpose or noninfringement. In no event will CA be liable to the end user or any third party for any loss or damage, direct or indirect, from the use of this documentation, including without limitation, lost profits, business interruption, goodwill, or lost data, even if CA is expressly advised of such loss or damage.

The use of any product referenced in this documentation and this documentation is governed by the end user's applicable license agreement.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with “Restricted Rights” as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227-7013(c)(1)(ii) or applicable successor provisions.

**Second Edition, October 2001**

© 2001 Computer Associates International, Inc.
All rights reserved.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

# Contents

**Chapter 1. Understanding Syntax Diagrams** . . . . . 1-1

**Chapter 2. Statements** . . . . . 2-1

**Chapter 3. Statement Syntax** . . . . . 3-1
3.1 ALLOCATE CURSOR . . . . . 3-3
3.2 ALTER ACCESS MODULE . . . . . 3-4
3.3 ALTER CATALOG . . . . . 3-5
3.4 ALTER INDEX . . . . . 3-6
3.5 ALTER PROCEDURE . . . . . 3-7
3.6 ALTER SCHEMA . . . . . 3-8
3.7 ALTER TABLE . . . . . 3-9
3.8 ALTER TABLE PROCEDURE . . . . . 3-10
3.9 BEGIN DECLARE SECTION . . . . . 3-11
3.10 CALL . . . . . 3-12
3.11 CLOSE (programmatic only) . . . . . 3-13
3.12 COMMIT . . . . . 3-14
3.13 CONNECT . . . . . 3-15
3.14 CREATE ACCESS MODULE . . . . . 3-16
3.15 CREATE CALC . . . . . 3-17
3.16 CREATE CONSTRAINT . . . . . 3-18
3.17 CREATE INDEX . . . . . 3-19
3.18 CREATE KEY . . . . . 3-20
3.19 CREATE PROCEDURE . . . . . 3-21
3.20 CREATE SCHEMA . . . . . 3-22
3.21 CREATE TABLE . . . . . 3-23
3.22 CREATE TABLE PROCEDURE . . . . . 3-24
3.23 CREATE TEMPORARY TABLE . . . . . 3-25
3.24 CREATE VIEW . . . . . 3-26
3.25 DEALLOCATE PREPARE (programmatic only) . . . . . 3-27
3.26 DECLARE CURSOR . . . . . 3-28
3.27 DECLARE EXTERNAL CURSOR (programmatic only) . . . . . 3-29
3.28 DELETE . . . . . 3-30
3.29 DESCRIBE (programmatic only) . . . . . 3-31
3.30 DROP ACCESS MODULE . . . . . 3-32
3.31 DROP CALC . . . . . 3-33
3.32 DROP CONSTRAINT . . . . . 3-34
3.33 DROP INDEX . . . . . 3-35
3.34 DROP KEY . . . . . 3-36
3.35 DROP PROCEDURE . . . . . 3-37
3.36 DROP SCHEMA . . . . . 3-38
3.37 DROP TABLE . . . . . 3-39
3.38 DROP TABLE PROCEDURE . . . . . 3-40
3.39 DROP VIEW . . . . . 3-41
3.40 END DECLARE SECTION (programmatic only) . . . . . 3-42
3.41 EXECUTE (programmatic only) . . . . . 3-43
3.42 EXECUTE IMMEDIATE (programmatic only) . . . . . 3-44

3.43 EXPLAIN . . . . . 3-45
3.44 FETCH (programmatic only) . . . . . 3-46
3.45 GRANT access module execution privilege . . . . . 3-47
3.46 GRANT definition privileges . . . . . 3-48
3.47 GRANT table access privileges . . . . . 3-49
3.48 INCLUDE (programmatic only) . . . . . 3-50
3.49 INSERT . . . . . 3-51
3.50 OPEN (programmatic only) . . . . . 3-52
3.51 PREPARE (programmatic only) . . . . . 3-53
3.52 RELEASE . . . . . 3-54
3.53 RESUME SESSION . . . . . 3-55
3.54 REVOKE all table privileges . . . . . 3-56
3.55 REVOKE SQL definition privileges . . . . . 3-57
3.56 REVOKE execution privilege . . . . . 3-58
3.57 REVOKE table access privileges . . . . . 3-59
3.58 ROLLBACK . . . . . 3-60
3.59 SELECT . . . . . 3-61
3.60 SET ACCESS MODULE (programmatic only) . . . . . 3-62
3.61 SET SESSION . . . . . 3-63
3.62 SET TRANSACTION . . . . . 3-64
3.63 SUSPEND SESSION . . . . . 3-65
3.64 TRANSFER OWNERSHIP . . . . . 3-66
3.65 UPDATE . . . . . 3-67
3.66 WHENEVER (programmatic only) . . . . . 3-68

**Chapter 4. Syntax Expansions** . . . . . 4-1
4.1 Access-module-specification . . . . . 4-3
4.2 Aggregate-function . . . . . 4-4
4.3 Authorization-identifier . . . . . 4-5
4.4 Between-predicate . . . . . 4-6
4.5 Bulk-options . . . . . 4-7
4.6 Column-definition . . . . . 4-8
4.7 Comparison-predicate . . . . . 4-9
4.8 Cursor-name . . . . . 4-10
4.9 Cursor-specification . . . . . 4-11
4.10 Data-type . . . . . 4-12
4.11 Describe-input-expression . . . . . 4-14
4.12 Describe-output-expression . . . . . 4-15
4.13 Dynamic-bulk-options1 . . . . . 4-16
4.14 Dynamic-bulk-options2 . . . . . 4-17
4.15 Dynamic-name-clause . . . . . 4-18
4.16 Dynamic-parameter-marker . . . . . 4-19
4.17 Exists-predicate . . . . . 4-20
4.18 Extended-cursor-name . . . . . 4-21
4.19 Extended-search-condition . . . . . 4-22
4.20 Extended-statement-name . . . . . 4-23
4.21 Host-variable . . . . . 4-24
4.22 Index-block-specifications . . . . . 4-25
4.23 In-predicate . . . . . 4-26
4.24 Labeled-duration . . . . . 4-27
4.25 Like-predicate . . . . . 4-28

4.26 Linked-constraint-options . . . . . 4-29
4.27 Literal . . . . . 4-30
4.28 Null-predicate . . . . . 4-31
4.29 Order-by-specification . . . . . 4-32
4.30 Parameter-definition . . . . . 4-33
4.31 Procedure-reference . . . . . 4-34
4.32 Quantified-predicate . . . . . 4-35
4.33 Query-expression . . . . . 4-36
4.34 Query-specification . . . . . 4-37
4.35 Rcm-specification . . . . . 4-38
4.36 Ready-options . . . . . 4-39
4.37 Search-condition . . . . . 4-40
4.38 Session-attribute . . . . . 4-41
4.39 Special-register . . . . . 4-42
4.40 SQLDA-options . . . . . 4-43
4.41 Statement-name . . . . . 4-44
4.42 Static-cursor-name . . . . . 4-45
4.43 Static-statement-name . . . . . 4-46
4.44 Subquery . . . . . 4-47
4.45 Table-name . . . . . 4-48
4.46 Table-options . . . . . 4-49
4.47 Table-procedure-reference . . . . . 4-50
4.48 Value-expression . . . . . 4-51

**Chapter 5. DISPLAY and PUNCH Syntax** . . . . . 5-1
5.1 DISPLAY/PUNCH ALL statement . . . . . 5-4
5.2 DISPLAY/PUNCH ACCESS MODULE . . . . . 5-8
5.3 DISPLAY/PUNCH CALC KEY . . . . . 5-9
5.4 DISPLAY/PUNCH CONSTRAINT . . . . . 5-10
5.5 DISPLAY/PUNCH INDEX . . . . . 5-11
5.6 DISPLAY/PUNCH KEY . . . . . 5-12
5.7 DISPLAY/PUNCH PROCEDURE . . . . . 5-13
5.8 DISPLAY/PUNCH SCHEMA . . . . . 5-14
5.9 DISPLAY/PUNCH TABLE . . . . . 5-15
5.10 DISPLAY/PUNCH TABLE PROCEDURE . . . . . 5-16
5.11 DISPLAY/PUNCH VIEW . . . . . 5-17

**Chapter 6. Identifiers** . . . . . 6-1

**Chapter 7. Date/time values** . . . . . 7-1

**Chapter 8. Null values** . . . . . 8-1

**Chapter 9. CA-ADS** . . . . . 9-1

**Chapter 10. COBOL** . . . . . 10-1

**Chapter 11. PL/I** . . . . . 11-1

**Chapter 12. Precompiler Options and Directives** . . . . . 12-1

**Chapter 13. SQLCA** . . . . . 13-1
13.1 Structure . . . . . 13-3
13.2 SQLSTATE values . . . . . 13-6
13.3 SQLCODE error values . . . . . 13-9
13.4 SQLCODE and SQLCNRP values . . . . . 13-10

**Chapter 14. SQLDA** . . . . . 14-1
14.1 Structure . . . . . 14-3
14.2 SQLLEN . . . . . 14-5
14.3 SQLTYPE . . . . . 14-6
14.4 SQLSCALE . . . . . 14-7
14.5 SQLPRECISION . . . . . 14-8
14.6 SQLALN and SQLNALN . . . . . 14-9
14.7 SQLNULL . . . . . 14-10

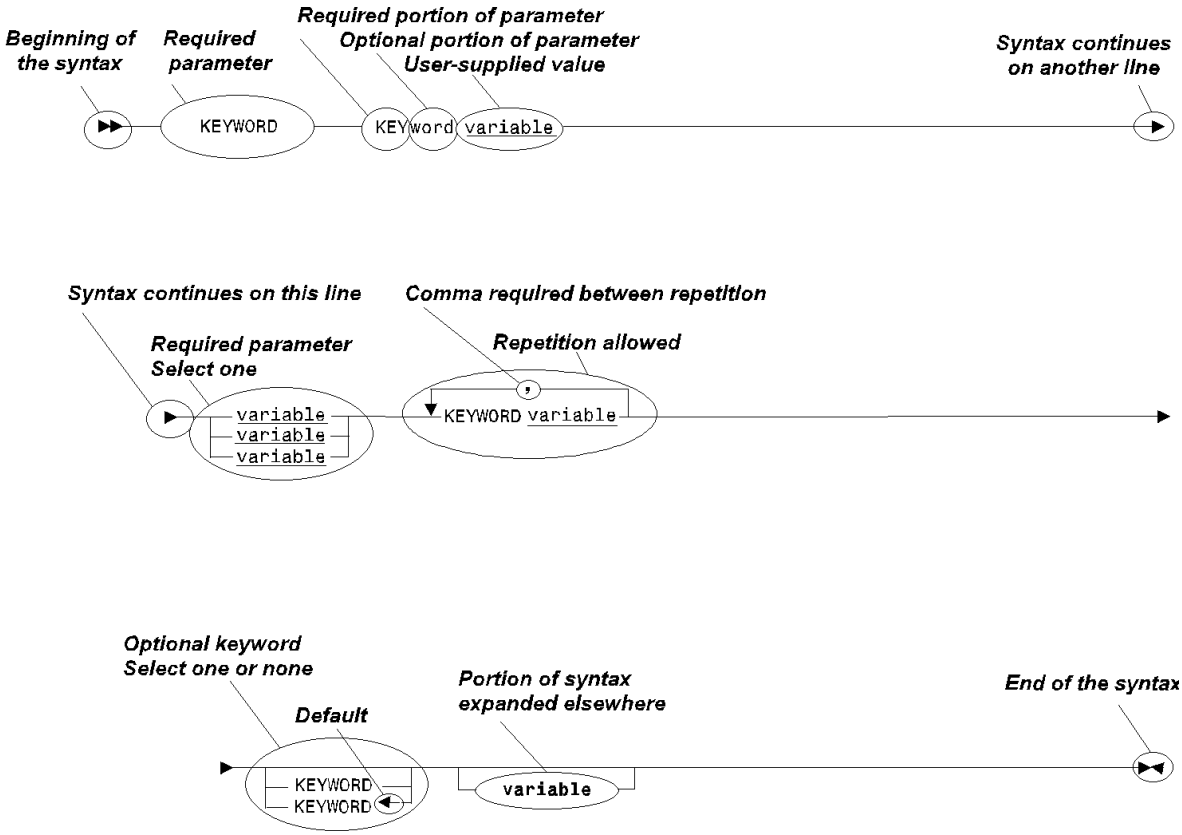
**Chapter 15. SYSTEM Tables** . . . . . 15-1
15.1 SYSTEM.AM . . . . . 15-3
15.2 SYSTEM.AMDEP . . . . . 15-4
15.3 SYSTEM.AREA . . . . . 15-5
15.4 SYSTEM.BUFFER . . . . . 15-6
15.5 SYSTEM.COLUMN . . . . . 15-7
15.6 SYSTEM.CONSTKEY . . . . . 15-8
15.7 SYSTEM.CONSTRAINT . . . . . 15-9
15.8 SYSTEM.DBNAME . . . . . 15-11
15.9 SYSTEM.DBSEGMENT . . . . . 15-12
15.10 SYSTEM.DBSSC . . . . . 15-13
15.11 SYSTEM.DBTABLE . . . . . 15-14
15.12 SYSTEM.DMCL . . . . . 15-15
15.13 SYSTEM.DMCLAREA . . . . . 15-16
15.14 SYSTEM.DMCLFILE . . . . . 15-17
15.15 SYSTEM.DMCLSEGMENT . . . . . 15-18
15.16 SYSTEM.FILE . . . . . 15-19
15.17 SYSTEM.FILEMAP . . . . . 15-20
15.18 SYSTEM.INDEX . . . . . 15-21
15.19 SYSTEM.INDEXKEY . . . . . 15-23
15.20 SYSTEM.JOURNAL . . . . . 15-24
15.21 SYSTEM.LOADHDR . . . . . 15-25
15.22 SYSTEM.ORDERKEY . . . . . 15-26
15.23 SYSTEM.SCHEMA . . . . . 15-27
15.24 SYSTEM.SECTION . . . . . 15-28
15.25 SYSTEM.SEGMENT . . . . . 15-29
15.26 SYSTEM.SYMBOL . . . . . 15-30
15.27 SYSTEM.SYNTAX . . . . . 15-31
15.28 SYSTEM.TABLE . . . . . 15-32
15.29 SYSTEM.VIEWDEP . . . . . 15-34
15.30 NOTES . . . . . 15-35

# Chapter 1. Understanding Syntax Diagrams

Look at the list of notation conventions below to see how syntax is presented in this manual. The example following the list shows how the conventions are used.

| | |
|---|---|
| `UPPERCASE`<br>`OR`<br>`SPECIAL CHARACTERS` | Represents a required keyword, partial keyword, character, or symbol that must be entered completely as shown. |
| `lowercase` | Represents an optional keyword or partial keyword that, if used, must be entered completely as shown. |
| `underlined lowercase` | Represents a value that you supply. |
| ◄ | Points to the default in a list of choices. |
| **`lowercase bold`** | Represents a portion of the syntax shown in greater detail at the end of the syntax or elsewhere in the document. |
| ►►──────────── | Shows the beginning of a complete piece of syntax. |
| ────────────►◄ | Shows the end of a complete piece of syntax. |
| ────────────► | Shows that the syntax continues on the next line. |
| ►──────────── | Shows that the syntax continues on this line. |
| ────────────►─ | Shows that the parameter continues on the next line. |
| ─►──────────── | Shows that a parameter continues on this line. |
| ►── parameter ──► | Shows a required parameter. |
| ►┬─ parameter ─┬──►<br>└─ parameter ─┘ | Shows a choice of required parameters. You must select one. |
| ►┬──────────┬──►<br>└─ parameter ─┘ | Shows an optional parameter. |
| ►┬──────────┬──►<br>├─ parameter ─┤<br>└─ parameter ─┘ | Shows a choice of optional parameters. Select one or none. |
| ►─┬─────────┬─↓─ parameter ─┴──► | Shows that you can repeat the parameter or specify more than one parameter. |
| ►─┬──── , ────┬─↓─ parameter ─┴──► | Shows that you must enter a comma between repetitions of the parameter. |

## Sample Syntax Diagram



# Chapter 2. Statements

### Categories

| Category | Description |
|---|---|
| Access module management | Control the creation and characteristics of access modules |
| Authorization | Control access to and ownership of database entities |
| Data manipulation | Retrieve and update data in the database |
| Dynamic compilation | Control the runtime compilation and execution of SQL statements |
| Data description | Control the creation and characteristics of logical database entities |
| Precompiler directives | Instruct the precompiler to include specified data structures and to generate specified error-processing code |
| Session management | Establish and control the characteristics of SQL sessions |
| Transaction management | Establish and control the characteristics of CA-IDMS/DB database transactions |

### Access module management statements

| Statement | Purpose |
|---|---|
| ALTER ACCESS MODULE | Modifies an access module in the dictionary |
| CREATE ACCESS MODULE | Creates an access module from one or more SQL statement modules (RCMs) |
| DROP ACCESS MODULE | Deletes an access module and its definition from the dictionary |
| EXPLAIN | Describes the strategy used to access data for a DELETE, INSERT, SELECT, or UPDATE statement |

### Authorization statements

| Statement | Purpose |
|---|---|
| GRANT definition privileges | Gives one or more users the privilege of performing selected actions on a specified schema, access module, table, or view |
| GRANT execution privilege | Gives one or more users the privilege of executing a specified access module |
| GRANT all privileges | Gives one or more users all definition and access privileges on a specified table or view |
| GRANT table access privileges | Gives one or more users the privilege of performing selected actions on a specified table or view |
| REVOKE definition privileges | Removes from one or more users the privilege of performing selected actions on a specified schema, access module, table, or view |
| REVOKE execution privilege | Removes from one or more users the privilege of executing a specified access module |
| REVOKE all privileges | Removes from one or more users all definition and access privileges on a specified table or view |
| REVOKE table access privileges | Removes from one or more users the privilege of performing selected actions on a specified table or view |
| TRANSFER OWNERSHIP | Passes ownership of a schema from one user or group of users to another |

### Data manipulation statements

| Statement | Purpose |
|---|---|
| CALL | Invokes a procedure or a table procedure |
| CLOSE* | Places a specified cursor in the closed state |
| DECLARE CURSOR* | Defines a cursor for a specified result table |
| DECLARE EXTERNAL CURSOR* | Identifies an externally defined global cursor to be used by the application program |
| DELETE | Deletes one or more rows from a table |
| FETCH* | Retrieves values from the result table associated with a cursor |
| INSERT | Adds one or more new rows to a table |
| OPEN* | Places a specified cursor in the open state |
| SELECT | Retrieves values from one or more tables and views |
| UPDATE | Modifies the values in one or more rows of a table |

**Programmatic only*

### Dynamic compilation statements

| Statement | Purpose |
|---|---|
| ALLOCATE CURSOR* | Defines a cursor for a dynamically-prepared statement |
| DEALLOCATE PREPARE* | Destroys a dynamically-compiled statement and all other dynamically-compiled statements that directly or indirectly reference it |
| DESCRIBE* | Directs CA-IDMS/DB to return information about a dynamically compiled SQL statement in an SQL descriptor area |
| EXECUTE* | Executes a dynamically compiled SQL statement |
| EXECUTE IMMEDIATE* | Dynamically compiles and executes an SQL statement |
| PREPARE* | Dynamically compiles an SQL statement for later execution in the application program |

**Programmatic only*

### Data description statements

| Statement | Purpose |
|---|---|
| ALTER CATALOG | Supports the correct sorting of additional national characters for specific languages (used by the alternate character set feature) |
| ALTER INDEX | Modifies the definition of an index in the dictionary |
| ALTER PROCEDURE | Modifies the definition of a procedure in the dictionary |
| ALTER SCHEMA | Modifies the definition of a schema in the dictionary |
| ALTER TABLE | Modifies the definition of a base table in the dictionary |
| ALTER TABLE PROCEDURE | Modifies the definition of a table procedure in the dictionary |
| CREATE CALC | Defines a CALC key on a base table |
| CREATE CONSTRAINT | Defines a constraint in the dictionary |
| CREATE INDEX | Defines an index on a base table |
| CREATE PROCEDURE | Defines a procedure in the dictionary |
| CREATE SCHEMA | Defines a schema in the dictionary |
| CREATE TABLE | Defines a table in the dictionary |
| CREATE TABLE PROCEDURE | Defines a table procedure in the dictionary |
| CREATE TEMPORARY TABLE | Defines a temporary table |
| CREATE VIEW | Defines a view in the dictionary |
| DROP CONSTRAINT | Deletes the definition of a constraint from the dictionary |
| DROP CALC | Deletes the definition of a CALC key from the dictionary |
| DROP INDEX | Deletes the definition of an index from the dictionary |
| DROP PROCEDURE | Deletes the definition of a procedure in the dictionary |
| DROP SCHEMA | Deletes the definition of a schema from the dictionary |

| Statement | Purpose |
| --- | --- |
| DROP TABLE | Deletes the definition of a base table from the dictionary |
| DROP TABLE PROCEDURE | Deletes the definition of a table procedure in the dictionary |
| DROP VIEW | Deletes the definition of a view from the dictionary |

### Precompiler-directive statements

| Statement | Purpose |
|---|---|
| BEGIN DECLARE SECTION* | Notifies the precompiler that a host variable definition is beginning |
| END DECLARE SECTION* | Notifies the precompiler that a host variable definition has ended |
| INCLUDE* | Directs the precompiler to create host variable definitions for a specified structure or table in the application program |
| WHENEVER* | Specifies an action to be taken when the execution of an SQL statement results in a nonzero SQLCODE value |

**Programmatic only*

### Session management statements

| Statement | Purpose |
|---|---|
| CONNECT | Establishes a connection to a CA-IDMS/DB dictionary and begins an SQL session |
| RELEASE | Releases a connection to a CA-IDMS/DB dictionary and ends the SQL session |
| RESUME SESSION | Resumes a suspended SQL session |
| SET SESSION | Establishes SQL session characteristics |
| SUSPEND SESSION | Suspends an SQL session and any transaction currently active within the session |

### Transaction management statements

| Statement | Purpose |
|---|---|
| COMMIT | Makes permanent the changes to the database made during the current transaction and optionally ends the transaction |
| ROLLBACK | Cancels changes made to the database during the current transaction and ends the transaction |
| SET ACCESS MODULE* | Identifies the access module to be used by a transaction |

| Statement | Purpose |
| --- | --- |
| SET TRANSACTION | Overrides access module defaults for conditions under which a transaction executes |

**Programmatic only*

### Teleprocessing statements

| Statement | Effect |
| --- | --- |
| COMMIT TASK | COMMIT CONTINUE on all nonsuspended SQL sessions and database transactions |
| ROLLBACK TASK | ROLLBACK RELEASE on all nonsuspended SQL sessions and database transactions |
| COMMIT TASK ALL | COMMIT on all nonsuspended SQL sessions and database transactions |
| FINISH TASK | COMMIT RELEASE on all nonsuspended SQL sessions and database transactions |
| ROLLBACK TASK CONTINUE | ROLLBACK on all nonsuspended SQL sessions and database transactions |
| Task termination with pending next task code (for example, DISPLAY in CA-ADS; DC RETURN NEXT TASK CODE in COBOL) | ROLLBACK RELEASE on all nonsuspended SQL sessions |
| Task termination without pending next task code (for example, LEAVE ADS; DC RETURN in COBOL) | ROLLBACK RELEASE on all SQL sessions |
| TASK ABEND | ROLLBACK RELEASE on all SQL sessions |

# Chapter 3. Statement Syntax


3.1 ALLOCATE CURSOR . . . . . 3-3
3.2 ALTER ACCESS MODULE . . . . . 3-4
3.3 ALTER CATALOG . . . . . 3-5
3.4 ALTER INDEX . . . . . 3-6
3.5 ALTER PROCEDURE . . . . . 3-7
3.6 ALTER SCHEMA . . . . . 3-8
3.7 ALTER TABLE . . . . . 3-9
3.8 ALTER TABLE PROCEDURE . . . . . 3-10
3.9 BEGIN DECLARE SECTION . . . . . 3-11
3.10 CALL . . . . . 3-12
3.11 CLOSE (programmatic only) . . . . . 3-13
3.12 COMMIT . . . . . 3-14
3.13 CONNECT . . . . . 3-15
3.14 CREATE ACCESS MODULE . . . . . 3-16
3.15 CREATE CALC . . . . . 3-17
3.16 CREATE CONSTRAINT . . . . . 3-18
3.17 CREATE INDEX . . . . . 3-19
3.18 CREATE KEY . . . . . 3-20
3.19 CREATE PROCEDURE . . . . . 3-21
3.20 CREATE SCHEMA . . . . . 3-22
3.21 CREATE TABLE . . . . . 3-23
3.22 CREATE TABLE PROCEDURE . . . . . 3-24
3.23 CREATE TEMPORARY TABLE . . . . . 3-25
3.24 CREATE VIEW . . . . . 3-26
3.25 DEALLOCATE PREPARE (programmatic only) . . . . . 3-27
3.26 DECLARE CURSOR . . . . . 3-28
3.27 DECLARE EXTERNAL CURSOR (programmatic only) . . . . . 3-29
3.28 DELETE . . . . . 3-30
3.29 DESCRIBE (programmatic only) . . . . . 3-31
3.30 DROP ACCESS MODULE . . . . . 3-32
3.31 DROP CALC . . . . . 3-33
3.32 DROP CONSTRAINT . . . . . 3-34
3.33 DROP INDEX . . . . . 3-35
3.34 DROP KEY . . . . . 3-36
3.35 DROP PROCEDURE . . . . . 3-37
3.36 DROP SCHEMA . . . . . 3-38
3.37 DROP TABLE . . . . . 3-39
3.38 DROP TABLE PROCEDURE . . . . . 3-40
3.39 DROP VIEW . . . . . 3-41
3.40 END DECLARE SECTION (programmatic only) . . . . . 3-42
3.41 EXECUTE (programmatic only) . . . . . 3-43
3.42 EXECUTE IMMEDIATE (programmatic only) . . . . . 3-44
3.43 EXPLAIN . . . . . 3-45
3.44 FETCH (programmatic only) . . . . . 3-46
3.45 GRANT access module execution privilege . . . . . 3-47
3.46 GRANT definition privileges . . . . . 3-48
3.47 GRANT table access privileges . . . . . 3-49

3.48 INCLUDE (programmatic only) . . . . . 3-50
3.49 INSERT . . . . . 3-51
3.50 OPEN (programmatic only) . . . . . 3-52
3.51 PREPARE (programmatic only) . . . . . 3-53
3.52 RELEASE . . . . . 3-54
3.53 RESUME SESSION . . . . . 3-55
3.54 REVOKE all table privileges . . . . . 3-56
3.55 REVOKE SQL definition privileges . . . . . 3-57
3.56 REVOKE execution privilege . . . . . 3-58
3.57 REVOKE table access privileges . . . . . 3-59
3.58 ROLLBACK . . . . . 3-60
3.59 SELECT . . . . . 3-61
3.60 SET ACCESS MODULE (programmatic only) . . . . . 3-62
3.61 SET SESSION . . . . . 3-63
3.62 SET TRANSACTION . . . . . 3-64
3.63 SUSPEND SESSION . . . . . 3-65
3.64 TRANSFER OWNERSHIP . . . . . 3-66
3.65 UPDATE . . . . . 3-67
3.66 WHENEVER (programmatic only) . . . . . 3-68

# 3.1 ALLOCATE CURSOR

```
►►── ALLOCATE extended-cursor-name CURSOR ──────────────────────────►
 ►── FOR extended-statement-name ──────────────────────────────────►◄
```

# 3.2 ALTER ACCESS MODULE

```
►►── ALTER ACCESS MODULE ┬──────────────┬─ access-module-name ──────────►
                        └─ schema-name.┘

►┬──────────────────────────────────────┬──────────────────────────────►
 └─ VERSION am-version-number ──────────┘

►┬──────────────────────────────────────────────────────────────────────►
 │                  ┌──────────,──────────┐
 └─ ADD -↓- rcm-specification ─────────────┘

►┬──────────────────────────────────────────────────────────────────────►
 │                  ┌──────,──────┐
 └─ DROP -↓- rcm-name -────────────┘

►┬──────────────────────────────────────────────────────────────────────►
 │                  ┌──────────,──────────┐
 │                  ↓- rcm-specification ──┴──────────────────────────┐
 │ REPLACE ─┬─ CHANGED ──────────────────────────────────────────────┤
 │          ├─ ALL ──┬──────────────────────────────────────────────┤
 │          │        │                  ┌──────────────,──────────────┐
 │          │        └─ MAP -↓─┬─ schema-name-1 ─┬─ TO - schema-name-2 ┴──┘
 │          │                  └─ NULL ──────────┘
 │          └──────────────────────────────────────────────────────┘

►┬──────────────────────────────────────────────────────────────────────►
 └─ AUTO RECREATE ─┬─ ON -┬──────────┘
                   └─ OFF ┘

►┬──────────────────────────────────────────────────────────────────────►
 └─ VALIDATE ─┬─ BY STATEMENT ─┬──────────┘
              ├─ BY MODULE ────┤
              └─ ALL ──────────┘

►┬──────────────────────────────────────────────────────────────────────►
 ├─ READ ONLY -┬──────────┘
 └─ READ WRITE ┘

►┬──────────────────────────────────────────────────────────────────────►
 └─ DEFAULT ISOLATION ─┬─ CURSOR STABILITY ─┬──────────┘
                       └─ TRANSIENT READ ───┘

►┬──────────────────────────────────────────────────────────────────────►◄
 │                  ┌──────────────────,──────────────────┐
 │                  ↓- segment-name.area-name ready-options -┴──────┐
 │ READ ─┬──────────────────────────────────────────────────────┤
 │       └─ ALL ready-options ──────────────────────────────────┘
```

# 3.3 ALTER CATALOG

```
►►── ALTER CATALOG DEFAULT CHARACTER SET alternate-character-set-name──►◄
```

# 3.4 ALTER INDEX

```
►►─ ALTER INDEX index-name ON ┬──────────────────┬─ table-identifier ──────►
                              └─ schema-name. ───┘

 ►─ INDEX BLOCK CONTAINS key-count KEYS ──────────────────────────────────►◄
```

# 3.5 ALTER PROCEDURE

```
►►── ALTER PROCEDURE ┬──────────────────────┬─ procedure-identifier ──────►
                     └─ schema-name. ───────┘

 ►┬─ ADD parameter-definition ──────────────────────────────────────────►
  │                  ┌──────────┐
  │                  ↓          │
  ├─ ADD ( ─┬─ parameter-definition ─┬─ ) ──────────────────────────────┤
  ├─ EXTERNAL NAME external-routine-name ──────────────────────────────┤
  ├──ESTIMATED ROWS row-count ─────────────────────────────────────────┤
  ├──ESTIMATED IOS io-count ───────────────────────────────────────────┤
  ├──LOCAL WORK AREA local-stge-size ──────────────────────────────────┤
  ├──GLOBAL WORK AREA global-stge-size ┬──────────────────────────────┤
  │                                    └─ KEY ┬─ key-ID ─┬─┤
  │                                           └─ NULL ─┘  │
  ├──USER MODE ────────────────────────────────────────────────────────┤
  └──SYSTEM MODE ──────────────────────────────────────────────────────┘
```

# 3.6 ALTER SCHEMA

```
►►── ALTER SCHEMA schema-name ──────────────────────────────────►
►┬──────────────────────────────────────────────────────────────►◄
 ├─ DEFAULT AREA ─┬─ segment-name.area-name ─┬──────────────────┤
 │                └─ NULL ───────────────────┘                  │
 ├─ DBNAME ─┬─ database-name ─┬──────────────────────────────── │
 │          └─ NULL ──────────┘                                 │
 └─ FOR NONSQL SCHEMA nonsql-schema-specification ──────────────┘
```

**Expansion of nonsql-schema-specification**

```
►►┬──────────────────────┬─ nonsql-schema-name ─┬──────────────────────────┬──►
  └─ dictionary-name. ───┘                      └─ VERSION version-number ─┘
►┬──────────────────────────────────────────────────────────────────────────►◄
 └─ DBNAME ─┬─ database-name ─┬──┘
            └─ NULL ──────────┘
```

# 3.7 ALTER TABLE

```
►►── ALTER TABLE ┬──────────────────┬─ table-identifier ──────────────►
                └─ schema-name. ───┘

 ►─┬─ ADD column-definition ──────────────┬──────────────────────────►◄
   ├─ ADD CHECK ( search-condition ) ─────┤
   ├─ DROP CHECK ─────────────────────────┤
   ├─ DROP DEFAULT INDEX ─────────────────┤
   └─ ESTIMATED ROWS estimated-row-count ─┘
```

# 3.8 ALTER TABLE PROCEDURE

```
►►── ALTER TABLE PROCEDURE ┬──────────────┬─ table-procedure-identifier ──►
                          └─ schema-name. ┘

 ►┬─ ADD parameter-definition ──────────────────────────────────────────►
  │                  ┌──────────,──────────┐
  ├─ ADD ( ─┬─↓─ parameter-definition ─┴─ ) ──────────────────────────┤
  ├─ EXTERNAL NAME external-routine-name ──────────────────────────────┤
  ├──ESTIMATED ROWS row-count ─────────────────────────────────────────┤
  ├──ESTIMATED IOS io-count ───────────────────────────────────────────┤
  ├──LOCAL WORK AREA local-stge-size ──────────────────────────────────┤
  ├──GLOBAL WORK AREA global-stge-size ┬──────────────────────────────┤
  │                                    └─ KEY ┬─ key-ID ─┬─┤
  │                                           └─ NULL .  ┘
  ├──USER MODE ────────────────────────────────────────────────────────┤
  └──SYSTEM MODE ──────────────────────────────────────────────────────┘
```

# 3.9 BEGIN DECLARE SECTION

```
►►── BEGIN DECLARE SECTION ──────────────────────────────────────►◄
```

# 3.10 CALL

```
►►── CALL ──┬─ procedure-reference ─────────┬──────────────────────────────►◄
          └─ table-procedure-reference ─┘
```

# 3.11 CLOSE (programmatic only)

```
►►── CLOSE cursor-name ──────────────────────────────────────────►◄
```

# 3.12 COMMIT

```
►►── COMMIT work ─┬──────────┬──────────────────────────────────────►◄
                  ├─ CONTINUE ─┤
                  └─ RELEASE ──┘
```

# 3.13 CONNECT

```
►►── CONNECT TO ─┬─ dictionary-name ──────────┬──────────────────────────►◄
                 └─ :dictionary-variable-name ─┘
```

# 3.14 CREATE ACCESS MODULE

```
►►── CREATE ACCESS MODULE ┬──────────────────┬ access-module-name ──────────►
                         └ schema-name. ────┘

►┬──────────────────────────────────────────────────────────────────────►
 └ VERSION am-version-number -┘

►── FROM -┬─┬──────────────────┬ rcm-name ┬──────────────────────────┬──►
        ↓ │ └ dictionary-name. ┘          │ └ VERSION rcm-version-number ┘
        ,─┘                               └──────────────────────────┘

►┬──────────────────────────────────────────────────────────────────────►
 │ ┬──────────────────────────────────────────────────────────────────┬
 │ │                                  ,                               │
 └ MAP -┬─┬─┬ schema-name-1 ┬─ TO ── schema-name-2 ──┬──────────────────┘
      ↓ │ │ └ NULL ─────────┘                        │
      ,─┘ └──────────────────────────────────────────┘

►┬──────────────────────────────────────────────────────────────────────►
 └ AUTO RECREATE ┬─ ON ◄ ─┬
                 └─ OFF -─┘

►┬──────────────────────────────────────────────────────────────────────►
 └ VALIDATE ┬─ BY STATEMENT ─┬
            ├─ BY MODULE ────┤
            └─ ALL ──────────┘

►┬──────────────────────────────────────────────────────────────────────►
 ├─ READ ONLY ──┤
 └─ READ WRITE ◄ ─┘

►┬──────────────────────────────────────────────────────────────────────►
 └ DEFAULT ISOLATION ┬─ CURSOR STABILITY ─┬
                     └─ TRANSIENT READ ───┘

►┬──────────────────────────────────────────────────────────────────────►◄
 │ ┬──────────────────────────────────────────────────────────────────┬
 │ │                                  ,                               │
 └ READ ┬─┬─┬- segment-name.area-name ready-options ─┬──────────────────┘
      ↓ │ │ └ ALL ready-options ─────────────────────┘
      ,─┘ └──────────────────────────────────────────┘
```

# 3.15 CREATE CALC

```
►►── CREATE ┬──────────┬─ CALC key ──────────────────────────────►
          └─ UNIQUE ─┘

 ►── ON ┬──────────────┬─ table-identifier ──────────────────────►
        └─ schema-name. ─┘

 ►── ( -┌──────── , ────────┐- column-name ─┘ ) ──────────────────────────►◄
```

# 3.16 CREATE CONSTRAINT

```
►►── CREATE CONSTRAINT constraint-name ──────────────────────────────►
►┬──────────────────┬ referencing-table ( -┌──────────┬↓- foreign-key-column ─┴─ ) ──►
 └ schema-name. ────┘                     └──────────,──────────┘
►── REFERENCES ┬──────────────────┬ referenced-table ──────────────────►
               └ schema-name. ────┘
►── ( -┌──────────┬↓- referenced-column ─┴─ ) ──────────────────────────►
          └──────────,──────────┘
►┬──────────────────────────────────────────────────────────────────┬►◄
 ├─ LINKED linked-constraint-options ─┤
 └─ UNLINKED ◄ ┬──────────────┬──────┘
               └ CLUSTERED ───┘
```

# 3.17 CREATE INDEX

```
►►── CREATE ┬──────────┬─ INDEX index-name ──────────────────────────►
          └─ UNIQUE ─┘

►── ON ┬──────────────┬─ table-identifier ──────────────────────────►
      └─ schema-name. ─┘

►── ( ┌──────────────,──────────────┐ ) ──────────────────────────►
    -↓- column-name ┬──────────┬──┘
                    ├─ ASC ◄ ─┤
                    └─ DESC -─┘

►┬──────────────────────────────────────────────────────────────►
 ├─ COMPRESSED ──────┤
 └─ UNCOMPRESSED ◄ ─┘

►┬──────────────────────────────┬┬──────────────────────────────┬►
 └─ IN segment-name.area-name ─┘└─ index-block-specification ─┘

►┬──────────────┬──────────────────────────────────────────────►◄
 └─ CLUSTERED ─┘
```

# 3.18 CREATE KEY

```
►►── CREATE ┬──────────┬── KEY key-name ──────────────────────────►
          ├─ UNIQUE ─┤
          └─ PRIMARY ─┘

 ►── ON ──┬──────────────┬──┬─ procedure-identifier ──────┬──────────►
          └─ schema-name. ─┘  └─ table-procedure-identifier ─┘

          ┌──────── , ────────┐
 ►── ( ─↓─ parameter-name. ─┘── ) ──┬──────────────────────────┬──►
                                   └─ ESTIMATED ROWS row-count ─┘
 ►──┬──────────────────────────────────────────────────────────►◄
    └─ ESTIMATED IOS io-count ─┘
```

# 3.19 CREATE PROCEDURE

```
►►── CREATE PROCEDURE ┬──────────────────────────┬─ procedure-identifier ──────►
                      └─ schema-name. ─┘

   ┌──────────────── , ──────────┐
►── ( ─┬─ parameter-definition - ) EXTERNAL NAME external-routine-name PROTOCOL IDMS ──►

►──┬──────────────────────────────────────────┬──┬──────────────────────────────────┬──►
   └─ ESTIMATED ROWS row-count ─┘ └─ ESTIMATED IOS io-count ─┘

►──┬──────────────┬──────────────────────────────────────────────────────────────────►
   ├─ USER MODE ←─┤
   └─ SYSTEM MODE ─┘

►──┬──────────────────────────────────────────────────────────────────────────────────►
   └── LOCAL WORK AREA  local-stge-size ─┘

►──┬──────────────────────────────────────────────────────────────────────────────────►
   └─ GLOBAL WORK AREA ── global-stge-size ──┬──────────────┬──┘
                                             └─ KEY key-id ─┘
```

# 3.20 CREATE SCHEMA

```
►►── CREATE SCHEMA schema-name ──────────────────────────────────►
►┬──────────────────────────────────────────────────────────────►◄
 ├─ DEFAULT AREA segment-name.area-name ──────────┤
 └─ FOR NONSQL SCHEMA nonsql-schema-specification ─┘
```

**Expansion of nonsql-schema-specification**

```
►►┬──────────────────────┬─ nonsql-schema-name ─┬──────────────────────────┬──►
  └─ dictionary-name. ─┘                        └─ VERSION version-number ─┘
►┬──────────────────────┬──────────────────────────────────────────────►◄
 └─ DBNAME database-name ─┘
```

# 3.21 CREATE TABLE

```
►►── CREATE TABLE ┬──────────────┬─ table-identifier ──────────────►
                 └─ schema-name. ┘

 ►── ( ┌──────────,──────────┐
       └─↓─ column-definition ┴──┬──────────────────────────┬── ) ──►
                                 └─ ,CHECK ( search-condition ) ┘

 ►──┬──────────────────────────┬──────────────────────────────────►
    └─ IN segment-name.area-name ┘

 ►──┬──────────────────────────────────────────────────────────┬──►
    └─ COMPRESS ┬──────────────────────────────────────────┬────┘
                └─ USING ┬─ BUILTIN ──────────────────────┬─┘
                         └─ data-characteristic-table-name ┘

 ►──┬──────────────────────────────────┬──────────────────────────►◄
    └─ ESTIMATED ROWS estimated-row-count ┘
```

# 3.22 CREATE TABLE PROCEDURE

```
►►── CREATE TABLE PROCEDURE ──┬──────────────┬── table-procedure-identifier ──►
                             └─ schema-name. ─┘

                   ┌──────────────,──────────┐
►── ( ─┬─ parameter-definition - ─┘ ) EXTERNAL NAME external-routine-name ──►

►──┬──────────────────────────────┬──┬──────────────────────────────┬──────►
   └─ ESTIMATED ROWS row-count ─┘  └─ ESTIMATED IOS io-count ─┘

►──┬──────────┬──────────────────────────────────────────────────────────►
   ├─ USER MODE ◄─┤
   └─ SYSTEM MODE ─┘

►──┬──────────────────────────────┬──────────────────────────────────────►
   └── LOCAL WORK AREA  local-stge-size ─┘

►──┬──────────────────────────────────────────────────┬──────────────────►◄
   └─ GLOBAL WORK AREA ── global-stge-size ──┬──────────┬┘
                                             └─ KEY key-id ─┘
```

# 3.23 CREATE TEMPORARY TABLE

```
►►── CREATE TEMPORARY TABLE table-identifier ──────────────────────────►
                              ┌──────────,──────────┐
►── ( -↓- column-definition ──┴──────────────────────────────────────────►◄
```

# 3.24 CREATE VIEW

```
►►── CREATE VIEW ┬──────────────────┬─ view-identifier ──────────────►
                └─ schema-name. ───┘

►┬──────────────────────────────────────────────────────────────────►
 │   ┌──────────── , ────────────┐
 └─ ( -↓- view-column-name ────┴── ) ┘

►── AS query-expression ──────────────────────────────────────────►

►┬──────────────────────────┬──────────────────────────────────────►
 └─ order-by-specification ─┘

►┬──────────────────────┬──────────────────────────────────────────►◄
 └─ WITH CHECK OPTION ─┘
```

# 3.25 DEALLOCATE PREPARE (programmatic only)

```
►►── DEALLOCATE PREPARE statement-name ──────────────────────────────►◄
```

# 3.26 DECLARE CURSOR

```
►►── DECLARE static-cursor-name ┬──────────┬─ CURSOR ──────────────────►
                               └─ GLOBAL ─┘

 ►── FOR ┬─ cursor-specification ─┬──────────────────────────────────►◄
         └─ static-statement-name ─┘
```

# 3.27 DECLARE EXTERNAL CURSOR (programmatic only)

```
►►── DECLARE static-cursor-name EXTERNAL CURSOR ──────────────────────────►◄
```

# 3.28 DELETE

```
►►── DELETE FROM table-name ─┬──────────┬──────────────────────────────────►
                             └─ alias ─┘

 ►─┬──────────────────────────────────────────────────────────────────────►◄
   └─ WHERE ─┬─ search-condition ──────────────────────────┬
             └─ CURRENT OF ─┬─ cursor-name ──────────────┬
                            └─ dynamic-name-clause ──────┘
```

# 3.29 DESCRIBE (programmatic only)

```
►►─ DESCRIBE ─┬─ OUTPUT ◄ ─┬─ statement-name ──────────────────────────►
              └─ INPUT ────┘

 ►─ USING sql DESCRIPTOR descriptor-area-name1 ──────────────────────────►

 ►─┬──────────────────────────────────────────────────────────────────┬─►◄
   ├─ INPUT -─┬─ USING sql DESCRIPTOR descriptor-area-name2 ─┘
   └─ OUTPUT ─┘
```

**Note 1:** If DESCRIBE OUTPUT is specified or implied, you may only specify the INPUT USING parameter; similarly, if DESCRIBE INPUT is specified, you may only specify the OUTPUT USING parameter.

**Note 2:** For compatibility with earlier releases, you can specify "INTO sql descriptor" in place of "USING sql DESCRIPTOR"; however, this is an extension to ANSI standard SQL.

# 3.30 DROP ACCESS MODULE

```
►►── DROP ACCESS MODULE ──────────────────────────────────────►
►┬──────────────┬─ access-module-name ──────────────────────►
 └─ schema-name. ┘
►┬──────────────────────────────┬──────────────────────────►
 └─ VERSION am-version-number ──┘
►┬──────────────┬──────────────────────────────────────────►◄
 └─ PRESERVE ───┘
```

# 3.31 DROP CALC

```
►►── DROP CALC key FROM ┬──────────────┬─ table-identifier ──────────►◄
                       └─ schema-name. ┘
```

# 3.32 DROP CONSTRAINT

```
►►── DROP CONSTRAINT constraint-name ──────────────────────────────►
 ►── FROM -┬──────────────────┬── table-identifier ──────────────────►◄
        └─ schema-name. ─┘
```

# 3.33 DROP INDEX

```
►►── DROP INDEX index-name ──────────────────────────────────────►
 ►── FROM ┬──────────────┬─ table-identifier ──────────────────►◄
       └─ schema-name. ─┘
```

# 3.34 DROP KEY

```
►►── DROP KEY key-name ──────────────────────────────────────────►
 ►── FROM ┬──────────────────┬┬─ procedure-identifier ──────────┬──►◄
        └─ schema-name. ─┘└─ table-procedure-identifier ─┘
```

# 3.35 DROP PROCEDURE

```
►►── DROP PROCEDURE ──┬──────────────────┬── procedure-identifier ──────────►◄
                      └── schema-name . ─┘
►──┬──────────┬──────────────────────────────────────────────────────────►◄
   └── CASCADE ─┘
```

# 3.36 DROP SCHEMA

```
►►── DROP SCHEMA schema-name ─┬──────────────────────────────────────►◄
                               └─ CASCADE ─┘
```

# 3.37 DROP TABLE

```
►►── DROP TABLE ┬──────────────┬─ table-identifier ──────────────►
                └─ schema-name. ┘

►┬──────────┬──────────────────────────────────────────────────►◄
 └─ CASCADE ┘
```

# 3.38 DROP TABLE PROCEDURE

```
►►── DROP TABLE PROCEDURE ──┬──────────────┬── table-procedure-identifier ──────►◄
                            └── schema-name .┘
►──┬──────────┬──────────────────────────────────────────────────────────────►◄
   └── CASCADE ┘
```

# 3.39 DROP VIEW

```
►►── DROP VIEW ──┬──────────────────┬── view-identifier ──────────────────►
                  └─ schema-name. ─┘

►──┬──────────┬──────────────────────────────────────────────────────────►◄
   └─ CASCADE ─┘
```

# 3.40 END DECLARE SECTION (programmatic only)

```
►►── END DECLARE SECTION ──────────────────────────────────────────►◄
```

# 3.41 EXECUTE (programmatic only)

```
►►─ EXECUTE statement-name ──────────────────────────────────────►
►─┬──────────────────────────────────────────────────────────────┬──►◄
  │                   ┌──────── , ────────┐                      │
  └─ USING ──┬─┬─↓─ host-variable ──┴──────────────────────────┴┬┘
             │ │                                                 │
             │ ├─ :dyn-buff sql DESCRIPTOR descriptor-area-name ─┤
             │ ├─ BULK :bulk-buffer bulk-options ────────────────┤
             │ └─ BULK :dyn-buff dynamic-bulk-options1 ──────────┘
```

# 3.42 EXECUTE IMMEDIATE (programmatic only)

```
►►── EXECUTE IMMEDIATE ─┬─ 'sql-statement' ──────────────────────────►◄
                        └─ :sql-statement-variable-name ─┘
```

# 3.43 EXPLAIN

```
►►── EXPLAIN ──────────────────────────────────────────────────►
►┬─ access-module-specification ──────────────────────────────┬►
 └─ STATEMENT 'sql-statement' ┬──────────────────────────────┬─┘
                              └─ STATEMENT NUMBER statement-number ─┘

►┬──────────────────────────────────────────────────────────┬►
 └─ INTO TABLE ┬──────────────────┬── table-identifier ─┘
               └─ schema-name. ─┘

►┬──────────────────────────────┬──────────────────────────►◄
 └─ IN segment-name.area-name ─┘
```

# 3.44 FETCH (programmatic only)

```
►►── FETCH cursor-name ──────────────────────────────────────────►
                  ┌──────── , ────────┐
►─┬─ INTO ─┬─┬─┬─↓─ host-variable ──┴──────────────────────────────┬──►◄
  │        │ │ │                                                  │
  │        │ │ └─ :dyn-buffer USING sql DESCRIPTOR descriptor-area-name ─┘
  │        │ │                                                    │
  │        │ └─ :bulk-buffer bulk-options ──────────────────────┬─┘
  │        │                                                    │
  └─ BULK ─┴─ :dyn-buffer dynamic-bulk-options2 ────────────────┘
```

# 3.45 GRANT access module execution privilege

```
►►── GRANT EXECUTE ──────────────────────────────────────────────────►
 ►── ON ACCESS MODULE ┬──────────────┬─ access-module-name ──────────►
                      └─ schema-name. ┘
                          ┌──────────── , ────────────┐
 ►── TO -↓─┬─ PUBLIC ──────────────────────────────────┴──────────────►
           └─ authorization-identifier ┘
 ►┬──────────────────────┬──────────────────────────────────────────►◄
  └─ WITH GRANT OPTION ──┘
```

# 3.46 GRANT definition privileges

```
►►── GRANT ─┬─ DEFINE ──────────────────────────────────────────────►
          │   ┌──────── , ────────┐
          └─↓─┬─ ALTER ─────────┬─┘
              ├─ CREATE ────────┤
              ├─ DISPLAY ───────┤
              ├─ DROP ──────────┤
              └─ REFERENCES ────┘

►── ON ─┬─ ACCESS MODULE ─┬──────────────┬─ access-module-name ─┬──────►
       │                 └─ schema-name. ─┘                      │
       ├─ SCHEMA schema-name ──────────────────────────────────────┤
       └─ table table-name ────────────────────────────────────────┘

          ┌──────────────── , ────────────────┐
►── TO -─↓─┬─ PUBLIC ──────────────────────────┬──────────────────────►
          └─ authorization-identifier ────────┘

►─┬──────────────────────────────────────────────────────────────►◄
  └─ WITH GRANT OPTION ─┘
```

# 3.47 GRANT table access privileges

```
►►── GRANT -┬─ ACCESS ──────────────────────────────────────────────►
          │   ┌──────── , ────────┐
          └─↓─┬─ DELETE ─┬──┘
              ├─ INSERT ─┤
              ├─ SELECT ─┤
              └─ UPDATE ─┘

 ►── ON table table-name ──────────────────────────────────────────►

          ┌──────────────── , ──────────────┐
 ►── TO -↓─┬─ PUBLIC ──────────────────────┬──────────────────────►
           └─ authorization-identifier ────┘

 ►┬──────────────────────────────────────────────────────────────►◄
  └─ WITH GRANT OPTION ─┘
```

# 3.48 INCLUDE (programmatic only)

```
►►── INCLUDE ─┬─ SQLCA ──────────────────────────────┬──────────────►◄
              ├─ SQLDA sqlda-options ────────────────┤
              ├─ TABLE table-name ─┬──────────────┬──┤
              │                    └─ table-options ┘  │
              └─ MODULE module-name ─────────────────┘
```

# 3.49 INSERT

```
►►── INSERT INTO table-name ──┬──────────────────────────────────────────────┬──►
                              └─ ( -┌──────── , ────────┐-- column-name -┘ ) ┘

►┬─ VALUES ( -┌──────────── , ────────────┐- ) ──────────────────────────────┬──►◄
 │            │ literal ──────────────────┤                                  │
 │            ├ host-variable ────────────┤                                  │
 │            ├ special-register ─────────┤                                  │
 │            └ NULL ─────────────────────┘                                  │
 ├─ query-specification ─────────────────────────────────────────────────────┤
 └─ BULK :bulk-buffer ──┬──────────────────┬──────────────────────────────────┘
                        └─ bulk-options ─┘
```

# 3.50 OPEN (programmatic only)

```
►►─ OPEN cursor-name ──────────────────────────────────────────►
►──┬──────────────────────────────────────────────────────────┬──►◄
  └─ USING ──┬─┬──────────── , ────────┬─ host-variable ──┬──┬──┘
             │ └──────────────────────┘                   │  │
             └─ :dyn-buff sql DESCRIPTOR descriptor-area-name ─┘
```

# 3.51 PREPARE (programmatic only)

```
►►─ PREPARE statement-name FROM ─┬─ :statement-text ─┬──────────────────────►
                                 └─ 'statement-text' ─┘

 ►─┬──────────────────────────────────┬──────────────────────────────────────►◄
   ├─ describe-output-expression ─────┤
   └─ describe-input-expression ──────┘
```

# 3.52 RELEASE

```
►►── RELEASE ──────────────────────────────────────────────────►◄
```

# 3.53 RESUME SESSION

```
►►── RESUME SESSION ──────────────────────────────────────────►◄
```

# 3.54 REVOKE all table privileges

```
►►── REVOKE ALL PRIVILEGES ──────────────────────────────────────►
 ►── ON table table-name ──────────────────────────────────────►

 ►── FROM ─┬─┬─ PUBLIC ──────────────────┬──────────────────────►◄
       ↓   │ └─ authorization-identifier ─┘
       └───┘
```

# 3.55 REVOKE SQL definition privileges

```
►►── REVOKE -┬─ DEFINE ──────────────────────────────────────────►
          │   ┌──────── , ────────┐
          └─↓─┴─ ALTER ─────────┴─┘
              ├─ CREATE ──────┤
              ├─ DISPLAY ─────┤
              ├─ DROP ────────┤
              └─ REFERENCES ──┘

►── ON ─┬─ ACCESS MODULE ─┬──────────────┬─ access-module-name ─┬──────►
       │                 └─ schema-name. -┘                      │
       ├─ SCHEMA schema-name ────────────────────────────────────┤
       └─ table table-name ──────────────────────────────────────┘

       ┌──────────────── , ──────────────┐
►── FROM -↓─┬─ PUBLIC ──────────────────┬──────────────────────────►◄
          └─ authorization-identifier ─┘
```

# 3.56 REVOKE execution privilege

```
►►── REVOKE EXECUTE ──────────────────────────────────────────►
 ►── ON ACCESS MODULE ─┬──────────────────┬─ access-module-name ──────►
                       └─ schema-name. ───┘
                                  ┌──────────── , ────────────┐
 ►── FROM -↓──┬─ PUBLIC ──────────────────────────────────────┴──────►◄
              └─ authorization-identifier ─┘
```

# 3.57 REVOKE table access privileges

```
►►── REVOKE ─┬─ ACCESS ──────────────────────────────────────────►
               │   ┌──────── , ────────┐
               └──↓─┬─ DELETE ─┬──┘
                    ├─ INSERT ─┤
                    ├─ SELECT ─┤
                    └─ UPDATE ─┘

 ►── ON table table-name ──────────────────────────────────────►

                 ┌──────────────── , ────────────────┐
 ►── FROM ─┬─↓─┬─ PUBLIC ──────────────────────────┬──────────────►◄
           │   └─ authorization-identifier ─┘
```

# 3.58 ROLLBACK

```
►►── ROLLBACK work ─┬──────────┬──────────────────────────────────►◄
                    └─ RELEASE ─┘
```

# 3.59 SELECT

```
►►── SELECT ──┬──────────┬──────────────────────────────────────────►
             ├─ ALL ◄ ──┤
             └─ DISTINCT ─┘

►─┬─ * ──────────────────────────────────────────────────────────►
  │ ┌──────────────────────── , ──────────────────────────┐
  └↓─┬─ value-expression -┬──────────────────────────┬──┘
     │                    │                          │
     │                    └──┬─────────┬─result-name ─┘
     │                       └─ AS ────┘
     ├─ table-name.* ──────────────────────────────────┤
     └─ alias.* ───────────────────────────────────────┘

►──┬──────────────────────────────────────────────────────────────┬──►
   │                  ┌──────── , ────────┐                        │
   ├─ INTO ─┬──────────↓─ host-variable ──┬──────────────────────┤
   │        │                             │                        │
   └─ BULK :bulk-buffer -┬──────────────┬──┘                        │
                         └─ bulk-options ─┘

►── FROM ─┬──────────────────────── , ──────────────────────────┐
         ↓─┬─ table-name ──────────┬──┬──────────┬──┬─────────┬──┘
           └─ (query-expression) ──┘  │          │  │         │
                                      └─ AS ─────┘  └─ alias ─┘

►──┬──────────────────────────────────────────────────────────────►
   └─ WHERE -┬─ search-condition ──────────────┬──┘
             └─ extended-search-condition ─────┘

►──┬──────────────────────────────────────────────────────────────►
   └─ PRESERVE ─┬─ table-name ─┬──┘
                └─ alias ──────┘

►──┬──────────────────────────────────────────────────────────────►
   │                  ┌──────────── , ────────────┐                │
   └─ GROUP BY ─┬──────↓─┬──────────────────┬─ column-name -┬──┘
                │        │                  │               │
                │        ├─ table-name. ────┤               │
                │        └─ alias. ──────────┘               │

►──┬──────────────────────────────────────────────────────────────►
   └─ HAVING search-condition ─┘

►──┬──────────────────────────────────────────────────────────────►
   └─ OPTIMIZE FOR literal ROWS ─┘

►──┬──────────────────────────────────────────────────────────────►
   │                  ┌──────────────────────────┐                │
   └─↓─ UNION ─┬──────┬──────────────────────────┬──┘
               └─ ALL -┘                          │
                                                  │

►──┬──────────────────────────────────────────────────────────────►◄
   │                  ┌──────────────────── , ────────────────────┐
   └─ ORDER BY ─┬──────↓─┬──────────────────┬─ column-name -┬──┬──┬──┬──┘
                │        │                  │               │  │  │  │
                │        ├─ table-name. ────┤               │  ├─ ASC ◄ ─┤
                │        └─ alias. ──────────┘               │  └─ DESC -─┘
                └─ column-number ────────────────────────────┘
```

# 3.60 SET ACCESS MODULE (programmatic only)

```
►►── SET ACCESS MODULE ─┬─ access-module-name ──────────────────────────►◄
                        └─ :access-module-variable-name ─┘
```

# 3.61 SET SESSION

```
►►── SET SESSION ──────────────────────────────────────────►
                  ┌──────────────────────────┐
►──────────────┬──┴──┬──────────────────────┬──────────────►◄
               ↓     │                      │
               └─ session-attribute ─┘
```

# 3.62 SET TRANSACTION

```
►►── SET TRANSACTION ──────────────────────────────────────────►
►──┬──────────────┬──────────────────────────────────────────►
  ├─ READ ONLY ─┤
  └─ READ WRITE ─┘

►──┬──────────────────┬──────────────────────────────────────►◄
  ├─ CURSOR STABILITY ─┤
  └─ TRANSIENT READ ───┘
```

# 3.63 SUSPEND SESSION

```
►►── SUSPEND SESSION ──────────────────────────────────────────►◄
```

# 3.64 TRANSFER OWNERSHIP

```
►►── TRANSFER OWNERSHIP OF SCHEMA schema-name ──────────────────────►
 ►── TO authorization-identifier ──────────────────────────────────►◄
```

# 3.65 UPDATE

```
►►── UPDATE table-name ─┬──────────────────────────────────────►
                        └─ alias ─┘

 ►── SET ─┬──────────────────────── , ──────────────────────┐
          ↓── column-name ── = ─┬─ value-expression ─┬──────►
                                ├─ NULL ─────────────┤
                                └─ (subquery) ───────┘

 ►─┬──────────────────────────────────────────────────────┬──►◄
   └─ WHERE ─┬─ search-condition ──────────────────────┬─┘
             └─ CURRENT OF ─┬─ cursor-name ──────────┬─┘
                            └─ dynamic-name-clause ─┘
```

# 3.66 WHENEVER (programmatic only)

```
►►── WHENEVER ─┬─ NOT FOUND -─┬─┬─ CONTINUE ──────────────────────────────┬──►◄
             ├─ SQLERROR ───┤ │ ┌─ GO TO ─┬─ label -─┐                │
             └─ SQLWARNING ─┘ │ └─ GOTO -─┘ └─ :label ─┘                │
                              └─ CALL subroutine-name ─────────────────┘
```

。

# Chapter 4. Syntax Expansions


4.1 Access-module-specification . . . . . 4-3
4.2 Aggregate-function . . . . . 4-4
4.3 Authorization-identifier . . . . . 4-5
4.4 Between-predicate . . . . . 4-6
4.5 Bulk-options . . . . . 4-7
4.6 Column-definition . . . . . 4-8
4.7 Comparison-predicate . . . . . 4-9
4.8 Cursor-name . . . . . 4-10
4.9 Cursor-specification . . . . . 4-11
4.10 Data-type . . . . . 4-12
4.11 Describe-input-expression . . . . . 4-14
4.12 Describe-output-expression . . . . . 4-15
4.13 Dynamic-bulk-options1 . . . . . 4-16
4.14 Dynamic-bulk-options2 . . . . . 4-17
4.15 Dynamic-name-clause . . . . . 4-18
4.16 Dynamic-parameter-marker . . . . . 4-19
4.17 Exists-predicate . . . . . 4-20
4.18 Extended-cursor-name . . . . . 4-21
4.19 Extended-search-condition . . . . . 4-22
4.20 Extended-statement-name . . . . . 4-23
4.21 Host-variable . . . . . 4-24
4.22 Index-block-specifications . . . . . 4-25
4.23 In-predicate . . . . . 4-26
4.24 Labeled-duration . . . . . 4-27
4.25 Like-predicate . . . . . 4-28
4.26 Linked-constraint-options . . . . . 4-29
4.27 Literal . . . . . 4-30
4.28 Null-predicate . . . . . 4-31
4.29 Order-by-specification . . . . . 4-32
4.30 Parameter-definition . . . . . 4-33
4.31 Procedure-reference . . . . . 4-34
4.32 Quantified-predicate . . . . . 4-35
4.33 Query-expression . . . . . 4-36
4.34 Query-specification . . . . . 4-37
4.35 Rcm-specification . . . . . 4-38
4.36 Ready-options . . . . . 4-39
4.37 Search-condition . . . . . 4-40
4.38 Session-attribute . . . . . 4-41
4.39 Special-register . . . . . 4-42
4.40 SQLDA-options . . . . . 4-43
4.41 Statement-name . . . . . 4-44
4.42 Static-cursor-name . . . . . 4-45
4.43 Static-statement-name . . . . . 4-46
4.44 Subquery . . . . . 4-47
4.45 Table-name . . . . . 4-48
4.46 Table-options . . . . . 4-49
4.47 Table-procedure-reference . . . . . 4-50

4.48 Value-expression . . . . . 4-51

# 4.1 Access-module-specification

```
►►── ACCESS MODULE access-module-name ──────────────────────────►
►─┬──────────────────────────────┬──────────────────────────────►
  └─ VERSION am-version-number ─┘
►─┬──────────────────────────────────────────────────────────►◄
  │          ┌──────── , ────────┐
  └─ MODULE -↓- rcm-name ─────────┘
```

# 4.2 Aggregate-function

```
►►─┬─┬─ AVG ─┬─ ( ─┬─┬─────────────────┬─┬─────────────────┬─ column-name -─┬─ ) ─┬─►◄
   │ │ MAX   │     │ └─ DISTINCT ─┘  │ │ table-name.     │                │     │
   │ │ MIN   │     │                 │ └─ alias.       ─┘                │     │
   │ └─ SUM ─┘     └─ all value-expression ─────────────────────────────┘     │
   └─ COUNT ( ─┬─ * ─────────────────────────────────────────────────────┬─ ) ─┘
               │                                                         │
               │                 ┌─────────────────┬─ column-name -─┬─┘
               └─ DISTINCT ─┘  │ table-name.     │                │
                               └─ alias.       ─┘
```

### Data types of function results

| Function | Data type of the value returned |
|---|---|
| AVG | Determined by the rules for data type conversion in arithmetic operations |
| COUNT | INTEGER |
| MAX | Same as the data type of the values specified by the argument |
| MIN | Same as the data type of the values specified by the argument |
| SUM | Determined by the rules for data type conversion in arithmetic operations |

## 4.3 Authorization-identifier

```
►►─┬─ user-identifier ─┬──────────────────────────────────────────►◄
   └─ group-identifier ─┘
```

# 4.4 Between-predicate

```
►►── value-expression ──────────────────────────────────────────►
►──┬──────────┬── BETWEEN value-expression AND value-expression ──────────►◄
  └─ NOT ────┘
```

## 4.5 Bulk-options

```
►►─┬──────────────────────────────┬──────────────────────────────────►
   └─ START :start-variable-name ─┘

►──┬──────────────────────────────┬──────────────────────────────────►◄
   └─ ROWS :row-count-variable-name ─┘
```

# 4.6 Column-definition

```
►►── column-name  data-type ──────────────────────────────────►
►──┬──────────────────────────────────────────────────────────►
  └─ NOT NULL ─┘
►──┬──────────────────────────────────────────────────────────►◄
  └─ WITH DEFAULT ─┘
```

# 4.7 Comparison-predicate

```
►►── value-expression  comparison-operator ─┬─ value-expression ─┬──────────►◄
                                            └─ ( subquery ) ─────┘
```

# 4.8 Cursor-name

```
►►─┬─ static-cursor-name ─┬──────────────────────────────────────►◄
   └─ extended-cursor-name ─┘
```

# 4.9 Cursor-specification

```
>>-- query-expression -+-------------------------+-------------------------->
                       +- order-by-specification -+
>-----------------------+-------------------------------------------------+-><
                       +- FOR -+- READ ONLY ---------------------------+
                               +- UPDATE --------------------------------+
                                         +-------------,-------------+
                                         +- OF -v-+-- column-name --+-+
```

# 4.10 Data-type

```
>>-+- BINary -----------------------------------------------+-------------------------><
   |            +- ( -- length -- ) -+                      |
   +- CHARacter -------------------------------------------+
   |            +- ( -- length -- ) -+                      |
   +- DATE ------------------------------------------------+
   +- DECimal ---------------------------------------------+
   |            +- ( -- precision ----------------- ) -+   |
   |                                +- , -- scale -+       |
   +- DOUBLE PRECISION ------------------------------------+
   +- FLOAT -----------------------------------------------+
   |            +- ( -- precision -- ) -+                   |
   +- GRAPHIC ---------------------------------------------+
   |            +- ( -- length -- ) -+                      |
   +- INTeger ---------------------------------------------+
   +- LONGINT ---------------------------------------------+
   +- NUMeric ---------------------------------------------+
   |            +- ( -- precision ----------------- ) -+   |
   |                                +- , -- scale -+       |
   +- REAL ------------------------------------------------+
   +- SMALLINT --------------------------------------------+
   +- TIME ------------------------------------------------+
   +- TIMESTAMP -------------------------------------------+
   +- UNSIGNED DECimal ------------------------------------+
   |                  +- ( - precision ----------- ) -+   |
   |                                   +- , - scale -+     |
   +- UNSIGNED NUMeric ------------------------------------+
   |                  +- ( - precision ----------- ) -+   |
   |                                   +- , - scale -+     |
   +- VARCHAR ---------------------------------------------+
   |            +- ( -- length -- ) -+                      |
   +- VARGRAPHIC ------------------------------------------+
                +- ( -- length -- ) -+
```

### Categories of data types

| Category | Data types |
|---|---|
| Approximate numeric | DOUBLE PRECISION<br>FLOAT<br>REAL |
| Binary | BINARY |
| Character string | CHARACTER<br>VARCHAR |
| Date/time | DATE<br>TIME<br>TIMESTAMP |
| Exact numeric | DECIMAL<br>INTEGER<br>LONGINT<br>NUMERIC<br>SMALLINT<br>UNSIGNED DECIMAL<br>UNSIGNED NUMERIC |

| Category | Data types |
| --- | --- |
| Graphics character string | GRAPHIC<br>VARGRAPHIC |

# 4.11 Describe-input-expression

```
►►─ DESCRIBE INPUT USING sql DESCRIPTOR descriptor-area-name2 ──────────►
►─┬──────────────────────────────────────────────────────────────────────►◄
  └─ OUTPUT USING sql DESCRIPTOR descriptor-area-name1 ─┘
```

## 4.12 Describe-output-expression

```
►►─ DESCRIBE output USING sql DESCRIPTOR descriptor-area-name1 ──────────►
►─┬──────────────────────────────────────────────────────────────────────►◄
  └─ INPUT USING sql DESCRIPTOR descriptor-area-name2 ─┘
```

# 4.13 Dynamic-bulk-options1

```
►►─┬──────────────────────────────┬──────────────────────────────────►
   └─ START :start-variable-name ─┘

 ►── ROWS :row-count-variable-name ──────────────────────────────────►

 ►── sql DESCRIPTOR descriptor-area-name ────────────────────────────►◄
```

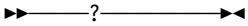
# 4.14 Dynamic-bulk-options2

```
►►─┬──────────────────────────────┬──────────────────────────────────►
   └─ START :start-variable-name ─┘

 ►── ROWS :row-count-variable-name ──────────────────────────────────►

 ►── using sql DESCRIPTOR descriptor-area-name ──────────────────────►◄
```

# 4.15 Dynamic-name-clause

```
►►─┬──────────────┬─ cursor-name ──────────────────────────────────►◄
   ├─ LOCAL ◄─────┤
   └─ GLOBAL ─────┘
```

## 4.16 Dynamic-parameter-marker



# 4.17 Exists-predicate

```
►►── EXISTS ( subquery ) ──────────────────────────────────────────►◄
```

# 4.18 Extended-cursor-name

```
►►─┬──────────────┬─┬─ 'cursor-name' ─┬──────────────────────────────►◄
   ├─ LOCAL ◄ ─┤ └─ :host-variable -┘
   └─ GLOBAL -┘
```

# 4.19 Extended-search-condition

```
►►─┬─ search-condition ─┬──┬──────────────────────────────────────────►◄
   └─ set-specification ─┘  │  ┌──────────────────────────────────┐
                            └──↓── AND ─┬─ search-condition ─┬──┘
                                        └─ set-specification ─┘
```

# 4.20 Extended-statement-name

```
►►─┬──────────────┬─┬─ 'statement-name' ─┬──────────────────────────►◄
   ├─ LOCAL ◄ ────┤ └─ :host-variable ───┘
   └─ GLOBAL ─────┘
```

# 4.21 Host-variable

```
►►── :host-variable-name ──┬──────────────────────────────┬──►◄
                           └─ indicator :indicator-variable-name ─┘
```

## 4.22 Index-block-specifications

```
►►── INDEX BLOCK CONTAINS key-count KEYs ──────────────────────────►
►─┬──────────────────────────────────────────────────────────────┬►◄
  └─ DISPLACEMENT IS page-count PAGES ─┘
```

# 4.23 In-predicate

```
►►── value-expression ┬──────┬── IN ──────────────────────────────────►
                      └─ NOT ─┘

►┬─ ( ┬─┬─────────── , ──────┬─┬─ ) ──────────────────────────────────►◄
 │    │ │↓- value-expression │ │
 │    │ └─ subquery ─────────┘ │
 │    └──────────────────────┘
 └─ value-expression ──────────┘
```

**Without NOT:** This table presents examples of results of IN predicates without NOT:

| Predicate | Result |
|---|---|
| 'A' IN ('A','B') | True |
| 'A' IN ('B') | False |
| 'A' IN (*null-value*) | Unknown |
| 'A' IN ('A',*null-value*) | True |
| 'A' IN ('B',*null-value*) | Unknown |

**With NOT:** This table presents examples of results of IN predicates with NOT:

| Predicate | Result |
|---|---|
| 'A' NOT IN ('A','B') | False |
| 'A' NOT IN ('B') | True |
| 'A' NOT IN (*null-value*) | Unknown |
| 'A' NOT IN ('A',*null-value*) | False |
| 'A' NOT IN ('B',*null-value*) | Unknown |

# 4.24 Labeled-duration

```
>>-+- aggregate-function ----------+-+- YEAR -----------+-------------------><
   |- scalar-function -------------| |- YEARS ----------|
   |-----------------+- column-name-| |- MONTH ----------|
   | |- table-name. -|             | |- MONTHS ---------|
   | |- alias. ------|             | |- DAY ------------|
   |- literal ---------------------| |- DAYS -----------|
   |- host-variable ---------------| |- HOUR -----------|
   |- value-expression ------------| |- HOURS ----------|
   |- dynamic-parameter-marker ----| |- MINUTE ---------|
                                     |- MINUTES --------|
                                     |- SECOND ---------|
                                     |- SECONDS --------|
                                     |- MICROSECOND ---|
                                     |- MICROSECONDS --|
```

# 4.25 Like-predicate

```
>>--- value-expression ---------------------------------------------------->
 >--+-------+-- LIKE --+- 'pattern' ----------------------------+---------->
    +- NOT -+          +- G 'graphics pattern' -----------------+
                       +- host-variable ------------------------+
                       +- special-register ---------------------+
                       +- dynamic-parameter-marker -------------+

 >--+-------------------------------------------------------------+-------><
    +- ESCAPE --+- 'escape-character' ----------+
                +- host-variable ---------------+
                +- dynamic-parameter-marker ----+
```

# 4.26 Linked-constraint-options

```
►►─┬─ CLUSTERED ──────────────────────────────────────────────────────►
   └─ INDEX ─┬─┬─ COMPRESSED ──────┬─┬─ index-block-specification ─┬─
             │ └─ UNCOMPRESSED ◄ ─┘ │                             │
             ├──────────────────────┤                             │
             │                      │                             │
             └──────────────────────┘                             │

►─┬──────────────────────────────────────────────────────────────────►◄
  │                                                                  │
  │                   ┌──────────────,──────────────┐                │
  │                   │                             │                │
  └─ ORDER BY ( -↓- sort-column ─┬─┬─────────────┬─┬─ ) ─┬─────────┬─┘
                                 │ │             │ │     │         │
                                 │ ├─ ASC ◄ ─────┤ │     └─ UNIQUE ─┘
                                 │ │             │ │
                                 │ └─ DESC ────────┘
```

# 4.27 Literal

```
►►─┬─ 'character-string-literal' ──────────────────────────────────►◄
   ├─ G'double-byte-character-string-literal' ─┤
   ├─ X'hexadecimal-literal' ──────────────────┤
   ├─ exact-numeric-literal ───────────────────┤
   └─ approximate-numeric-literal ─────────────┘
```

# 4.28 Null-predicate

```
►►─ value-expression ─ IS ─┬─────────┬─ NULL ──────────────────────────►◄
                           └─ NOT ─┘
```

# 4.29 Order-by-specification

```
          ┌──────────────────────── , ──────────────────────────┐
►►── ORDER BY ─↓──────────────────── column-name ─────────────────┴───────────►◄
               │  ├─ table-name. ─┤                │  ├─ ASC ◄ ─┤
               │  └─ alias. ──────┘                │  └─ DESC ──┘
               ├─ column-number ───────────────────┤
               └─ result-name ─────────────────────┘
```

# 4.30 Parameter-definition

```
►►── parameter-name ── data-type ┬──────────────┬──────────────────►◄
                                 └─ WITH DEFAULT ─┘
```

# 4.31 Procedure-reference

```
►►──┬──────────────────┬─ procedure-identifier ──────────────────►
    └─ schema-name. ───┘

►►──┬──────────────────────────────────────────────────────────┬──►
    │                  ┌──────────────┐                        │
    │                  ↓              │                        │
    └─ ( ─┬─ parameter-specification ─┴─ ) ────────────────────┘
          │                  ,
```

# 4.32 Quantified-predicate

```
►►── value-expression ──────────────────────────────────────────►
 ►── comparison-operator ─┬─ ALL ──────┬─ ( subquery ) ──────────►◄
                         │            │
                         └─ SOME ─────┤
                         └─ ANY ──────┘
```

# 4.33 Query-expression

```
>>─┬─ query-specification ─┬──────────────────────────────────────>
   └─ (query-expression) -─┘

 >─┬──────────────────────────────────────────────────────────────>><
   │ ┌──────────────────────────────────────┐
   └─↓- UNION ─┬─ query-specification ─┬─┬─┘
               └─ (query-expression) -─┘ │
```

# 4.34 Query-specification

```
►►── SELECT ┬──────────┬──────────────────────────────────────────►
           ├─ ALL  ◄──┤
           └─ DISTINCT┘

 ►┬─ * ──────────────────────────────────────────────────────────►
  │ ┌──────────────────────── , ──────────────────────────┐
  └↓┬── value-expression ──┬──────────────────────────────┬┘
    │                      └─ AS result-name ─┘           │
    ├─ table-name.* ─────────────────────────────────────┤
    └─ alias.* ──────────────────────────────────────────┘

 ►── FROM ─┌──────────────────────── , ──────────────────────────┐
           ↓┬── table-name ──────────┬──┬──────────┬──────────────►
            └── (query-expression) ─┘  └─ alias ─┘

 ►┬──────────────────────────────────────────────────────────────►
  └─ WHERE ┬── search-condition ──────────┬┘
           └── extended-search-condition ─┘

 ►┬──────────────────────────────────────────────────────────────►
  └─ PRESERVE ┬── table-name ──┬┘
              └── alias ────────┘

 ►┬──────────────────────────────────────────────────────────────►
  │ ┌──────────────────────── , ──────────────────────────┐
  └─ GROUP BY ─↓┬──────────────────── column-name ──┬┘
                ├─ table-name. ─┤
                └─ alias. ──────┘

 ►┬──────────────────────────────────────────────────────────────<
  └─ HAVING search-condition ─┘

 ►┬──────────────────────────────────────────────────────────────►◄
  └─ OPTIMIZE FOR literal ROWS ─┘
```

# 4.35 Rcm-specification

```
►►─┬──────────────────────┬─ rcm-name ─┬──────────────────────────────┬──►◄
   └─ dictionary-name. ─┘            └─ VERSION rcm-version-number ─┘
```

# 4.36 Ready-options

```
►►─┬─────────────────────────┬──────────────────────────────────────►
   ├─ SHARED RETRIEVAL ──────┤
   ├─ SHARED UPDATE ─────────┤
   ├─ PROTECTED RETRIEVAL ───┤
   ├─ PROTECTED UPDATE ──────┤
   └─ EXCLUSIVE ─────────────┘

►──┬──────────────────┬──────────────────────────────────────────►◄
   ├─ INCREMENTAL ────┤
   └─ PRECLAIM ───────┘
```

# 4.37 Search-condition

```
►►─┬─────────┬─┬─ between-predicate ──────────────────────────────►
   └─ NOT ───┘ ├─ comparison-predicate ─┤
               ├─ exists-predicate ─────┤
               ├─ in-predicate ─────────┤
               ├─ like-predicate ───────┤
               ├─ null-predicate ───────┤
               ├─ quantified-predicate ─┤
               └─ ( search-condition ) ─┘

►─┬──────────────────────────────────────────────────────────────┬►◄
  │  ┌──────────────────────────────────────────────────────────┐│
  └─↓─┬─ AND ─┬─┬─────────┬─┬─ between-predicate ──────────────┬┴┘
      └─ OR ──┘ └─ NOT ───┘ ├─ comparison-predicate ─┤
                            ├─ exists-predicate ─────┤
                            ├─ in-predicate ─────────┤
                            ├─ like-predicate ───────┤
                            ├─ null-predicate ───────┤
                            ├─ quantified-predicate ─┤
                            └─ ( search-condition ) ─┘
```

# 4.38 Session-attribute

```
►─┬─ CHECK SYNTAX ─┬─ SQL89 ──────────┬──────────────────────────────────┬──►◄
  │                │─ FIPS ───────────│                                  │
  │                └─ EXTENDED ───────┘                                  │
  │─ CURRENT SCHEMA ─┬─ schema-name ─┬──────────────────────────────────│
  │                  └─ NULL ────────┘                                  │
  │─┬─ CURSOR STABILITY ─┬──────────────────────────────────────────────│
  │ └─ TRANSIENT READ ───┘                                              │
  │                                                                     │
  └─┬─ READ ONLY ─┬──────────────────────────────────────────────────────┘
    └─ READ WRITE ─┘
```

# 4.39 Special-register

```
►►─┬─ USER ──────────────┬──────────────────────────────────────►◄
   ├─ GROUP ─────────────┤
   ├─ CURRENT DATE ──────┤
   ├─ CURRENT TIME ──────┤
   ├─ CURRENT TIMESTAMP ─┤
   ├─ CURRENT TIMEZONE --┤
   ├─ CURRENT DATABASE --┤
   ├─ CURRENT SCHEMA ────┤
   └─ CURRENT SQLID ─────┘
```

# 4.40 SQLDA-options

```
►►─┬──────────────────────────────────────────┬─┬──────────────────────────────────────────┬─►◄
   └─ NUMBER OF COLUMNS column-count ─┘   └─ AS descriptor-area-name ─┘
```

# 4.41 Statement-name

```
►►──┬─ static-statement-name ─┬──────────────────────────────────►◄
    └─ extended-statement-name -┘
```

## 4.42 Static-cursor-name

```
►►─── cursor-name ──────────────────────────────────────────────────►◄
```

# 4.43 Static-statement-name

```
►►─── statement-name ──────────────────────────────────────────►◄
```

## 4.44 Subquery

```
►►── query-specification ──────────────────────────────────────────►◄
```

# 4.45 Table-name

```
►►──┬──────────────────┬─┬─ table-identifier ─┬──────────────────────────►◄
    ├─ schema-name. ───┘ └─ view-identifier -─┘│
    ├─ table-procedure-reference ──────────────┤
    └─ procedure-reference ────────────────────┘
```

# 4.46 Table-options

```
►►─┬──────────────────────────────────────────────────────────────────────►
   │ ( -┌──────────,──────────┐-- column-name -┘ ) -┬──────────────────────┐
   │    │                                           │ AS ( -┌──────────,──────────┐- column-alias ┘ ) ┘
   │    │                                           │       │                                       │
   │    │                                           │       │                                       │

►─┬─────────────────────────────────────────────────────────────────────►
  │ AS structure-name ┤
  │ NO STRUCTURE ─────┘

►─┬─────────────────────────────────────────────────────────────────────►
  │ NUMBER OF ROWS row-count ┘

►─┬─────────────────────────────────────────────────────────────────────►
  │ PREFIX 'column-prefix' ┘

►─┬─────────────────────────────────────────────────────────────────────►
  │ SUFFIX 'column-suffix' ┘

►─┬─────────────────────────────────────────────────────────────────────►◄
  │ LEVEL level-number ┘
```

# 4.47 Table-procedure-reference

```
►►──┬──────────────────┬─ table-procedure-identifier ──────────────►
    └─ schema-name. ───┘

►►──┬──────────────────────────────────────────────────────────────►
    │                  ┌──────────────┐
    │                  ┌──────────,───┐
    └─ ( ─┬─ parameter-specification ─┴─ ) ──────────────────────┘
```

# 4.48 Value-expression

```
►►─┬──────────┬─┬─ aggregate-function ──────────────────────────────►
   ├─ + ◄ ────┤ ├─ scalar-function ─────────────────────────────────┤
   └─ - ──────┘ ├──────────────────── column-name -─┬──────────────┤
                │ ├─ table-name. ─┤                 │              │
                │ └─ alias. ──────┘                 │              │
                ├─ literal ─────────────────────────┤              │
                ├─ host-variable ───────────────────┤              │
                ├─ special-register ────────────────┤              │
                ├─ ( value-expression ) ────────────┤              │
                └─ dynamic-parameter-marker ────────┘              │
►──┬──────────────────────────────────────────────────────────────┬──►◄
   │  ┌──────────────────────────────────────────────────────────┐│
   └─↓─┬─ * -─┬─┬──────┬─┬─ aggregate-function ──────────────────┤│
     ├─ / -─┤ ├─ + ◄ ─┤ ├─ scalar-function ─────────────────────┤│
     ├─ + -─┤ └─ - ────┘ ├──────────────────── column-name -─┬──┤│
     ├─ - -─┤            │ ├─ table-name. ─┤                 │  │
     └─ || ─┘            │ └─ alias. ──────┘                 │  │
                         ├─ literal ─────────────────────────┤  │
                         ├─ host-variable ───────────────────┤  │
                         ├─ special-register ────────────────┤  │
                         ├─ ( value-expression ) ────────────┤  │
                         ├─ labeled-duration ────────────────┤  │
                         └─ dynamic-parameter-marker─────────┘  │
```

# Chapter 5. DISPLAY and PUNCH Syntax


5.1 DISPLAY/PUNCH ALL statement . . . . . 5-4
5.2 DISPLAY/PUNCH ACCESS MODULE . . . . . 5-8
5.3 DISPLAY/PUNCH CALC KEY . . . . . 5-9
5.4 DISPLAY/PUNCH CONSTRAINT . . . . . 5-10
5.5 DISPLAY/PUNCH INDEX . . . . . 5-11
5.6 DISPLAY/PUNCH KEY . . . . . 5-12
5.7 DISPLAY/PUNCH PROCEDURE . . . . . 5-13
5.8 DISPLAY/PUNCH SCHEMA . . . . . 5-14
5.9 DISPLAY/PUNCH TABLE . . . . . 5-15
5.10 DISPLAY/PUNCH TABLE PROCEDURE . . . . . 5-16
5.11 DISPLAY/PUNCH VIEW . . . . . 5-17

DISPLAY and PUNCH operations produce as output the SQL statements that describe the named entity. DISPLAY and PUNCH operations do not update the entity description. You can choose to display or punch all the entity occurrences defined within an entity or only specific entity occurrences.

The location of the output depends on which verb is used and whether you are using the online or batch command facility:

- **DISPLAY** displays online output at the terminal and lists batch output in the command facility's activity listing.
- **PUNCH** writes the output to the system punch file. All punched output is also listed in the command facility's activity listing.

# 5.1 DISPLAY/PUNCH ALL statement

```
►►─┬─ DISplay ─┬─┬─ ALL ──────────────────────────┬─ entity-type ──────►
   └─ PUNch ───┘ │ FIRST ─┬─────────────────────┬│
                 └─ LAST ─┘ ├─ 1 ◄ ─────────────┤│
                            └─ entity-count ─────┘│

►─┬──────────────────────────────────────────────┬──────────────────────►
  └─ WHERe conditional-expression ───────────────┘
►─┬──────────────────────────────────────────────┬──────────────────────►
  └─ AS ─┬─ COMments ◄ ─┬─┘
         └─ SYNtax ─────┘
►─┬──────────────────────────────────────────────┬──────────────────►◄
  └─ VERB ─┬─ DISplay ◄ ─┬─┘
           ├─ ALTer ─────┤
           ├─ CREate ────┤
           ├─ DROp ──────┤
           └─ PUNch ─────┘
```

*Expansion of conditional-expression*

```
►►─┬─ mask-comparison ──────────────────────────┬──────────────────────►
   ├─ value-comparison ─────────────────────────┤
   └─┬──────────┬─ ( ─┬─ mask-comparison ─┬─ ) ─┘
     └─ NOT ────┘     └─ value-comparison ─┘

►─┬──────────────────────────────────────────────────────────────────┬──►◄
  │ ┌──────────────────────────────────────────────────────────────┐ │
  └─↓─┬─ AND ─┬─┬─ mask-comparison ──────────────────────────┬─┬──┘ │
      └─ OR ──┘ ├─ value-comparison ────────────────────────┤ │    │
                └─┬──────────┬─ ( ─┬─ mask-comparison ─┬─ ) ─┘ │    │
                  └─ NOT ────┘     └─ value-comparison ─┘       │    │
```

*Expansion of mask-comparison*

```
►►── entity-option-keyword ──────────────────────────────────────────►

►─┬─ CONTAINs ─┬─ 'mask-value' ──────────────────────────────────────►◄
  └─ MATCHES ──┘
```

*Expansion of value-comparison*

```
►►─┬─ 'character-string-literal' ─┬──────────────────────────────────►
   ├─ numeric-literal ────────────┤
   └─ entity-option-keyword ──────┘

►─┬─ IS ─┬──────────┬─┬─ 'character-string-literal' ─┬──────────────►◄
  │      └─ NOT ────┘ │ ├─ numeric-literal ────────────┤
  ├─ NE ──────────────┤ └─ entity-option-keyword ──────┘
  │                   │
  └─┬──────────┬─┬─ EQ ─┬─┬
    └─ NOT ────┘ │ └─ = ──┘ │
                 ├─ GT ─┬─┤
                 │ └─ > ──┘ │
                 ├─ LT ─┬─┤
                 │ └─ < ──┘ │
                 ├─ GE ─────┤
                 └─ LE ─────┘
```

### Keyword tables by type

| Entity type | Entity-option keyword | Selects based on |
|---|---|---|
| All entity types | entity-type NAMe | Unqualified name (1) |
| | entity-type | Unqualified name (1) |
| | FULl entity-type NAMe | Qualified name (1) |
| | DATe CREated | Date (MM/DD/YY) occurrence created |
| | MONth CREated | Month occurrence created |
| | DAY CREated | Day occurrence created |
| | YEAr CREated | Year occurrence created |
| | DATe last UPDated | Date (MM/DD/YY) occurrence last updated** |
| | MONth last UPDated | Month occurrence last updated** |
| | DAY last UPDated | Day occurrence last updated** |
| | YEAr last UPDated | Year occurrence last updated** |
| | CREated by | User who created occurrence** |
| | PREpared by | User who created occurrence** |
| | REVised by | User who last updated occurrence** |
| | LASt UPDated by | User who last updated occurrence** |
| ACCESS MODULE | AM name | Unqualified access module name (1) |
| | SCHema name | Name of access module's schema |
| | Version | Version number |
| | FULl TABle NAMe | Qualified name of a table referenced by the access module |
| | TABle SCHema name | Schema name of a table referenced by the access module |
| | TABle name | Unqualified name of a table referenced by the access module |
| | DATe COMpiled | Date (MM/DD/YY) access module compiled |
| | COMpiled | Date (MM/DD/YY) access module compiled |
| | MONth COMpiled | Month access module compiled |
| | DAY COMpiled | Day access module compiled |
| | YEAr COMpiled | Year access module compiled |
| CALC KEY | SCHema name | Schema name of the table containing the CALC key |
| | TABle name | Unqualified name of table containing the CALC key |

| Entity type | Entity-option keyword | Selects based on |
|---|---|---|
| CONSTRAINT | SCHema name | Schema name of the constraint |
| | REFERENCEd FULl TABle NAMe | Qualified name of the referenced table |
| | REFERENCEd table SCHema name | Schema name of the referenced table |
| | REFERENCEd TABle name | Unqualified name of the referenced table |
| | REFERENCIng FULl TABle NAMe | Qualified name of the referencing table |
| | REFERENCIng table SCHema name | Schema name of the referencing table |
| | REFERENCIng TABle name | Unqualified name of the referencing table |
| INDEX | SCHema name | Schema name of the indexed table |
| | TABle name | Unqualified name of the indexed table |
| | FULl AREa NAMe | Qualified name of the area containing the index |
| | SEGment name | Segment name of the area containing the index |
| | AREa name | Unqualified name of the area containing the index |
| KEY | SCHema name | Schema name of the keyed table procedure |
| | TABle PROcedure name | Unqualified name of the keyed table procedure |
| | FULl TABle PROcedure NAMe | Qualified name of the keyed table procedure |
| PROCEDURE | SCHema name | Schema name of the procedure |
| | EXTernal NAMe | Name of the program called to process the procedure |
| SCHEMA | TYPe | Type of Schema (NONSQL or SQL) |
| | full DICtname | NonSQL Schema Dictionary name |
| | DBName | NonSQL Schema DBName |
| | NODe name | NonSQL Schema Node Name |
| | NODename | NonSQL Schema Node Name |
| | NONsql SCHema | Name of the NonSQL Schema |
| | nonsql schema Version | Version of the NonSQL Schema |
| | DEFault FULl AREA NAMe | Qualified area name of the Schema's default area |
| | default SEGment name | Segment name of Schema's default area |
| | default AREa name | Unqualified area name of the Schema's default area |

| Entity type | Entity-option keyword | Selects based on |
|---|---|---|
| TABLE | SCHema name | Schema name of the table |
| | FULl AREA NAMe | Qualified area name containing the table |
| | SEGment name | Segment name of the area containing the table |
| | AREa name | Unqualified name of the area containing the table |
| TABLE PROCEDURE | SCHema name | Schema name of the table procedure |
| | EXTernal NAMe | Name of the program called to process the procedure |
| VIEW | SCHema name | Schema name of the View |
| | REFerenced FULl TABle NAMe | Qualified name of a table referenced by the View |
| | REFerenced table SCHema NAMe | Schema name of a table(s) referenced by the View |
| | REFerenced TABle name | Unqualified name of a table referenced by the View |

(1)Unqualified name selections are based on the primary name of the entity occurrence only. To select based on the fully qualified occurrence name, token FULL NAME must be specified. SQL components with qualified names are specified in the table below.

**You can specify this keyword option only when using SCHEMA, TABLE, TABLE PROCEDURE, and VIEW entities.

# 5.2 DISPLAY/PUNCH ACCESS MODULE

```
►►─┬─ DISplay ─┬─ ACCess MODule is ─┬──────────────────┬─ access-module-name ──►
   └─ PUNch ───┘   AM ──────────────┘  └─ schema-name.─┘

►─┬──────────────────────────────────────────────────────────────────────────►
  └─ Version ─┬─ 1 ◄ ──────────┬
              ├─ version-number ─┤
              ├─ HIGhest ────────┤
              └─ LOWest ─────────┘

►─┬──────────────────────────────────────────────────────────────────────────►
  └─┬─ WITh ──────────┬─┬─ ALL ──────────────┬─ AS ─┬─COMments ◄ ─┬
    ├─ ALSo WITh ─────┤ ├─ NONe ─────────────┤      └─ SYNtax ────┘
    ├─ WITHOut ───────┤ ├─ DETails ──────────┤
    └─ ALSo WITHOut ──┘ ├─ HIStory ──────────┤
                        └─ TABles ───────────┘

►─┬──────────────────────────────────────────────────────────────────────────►◄
  └─ VERb ─┬─ CREate ◄ ─┬
           ├─ ALTer ────┤
           └─ DROp ─────┘
```

# 5.3 DISPLAY/PUNCH CALC KEY

```
►►─┬─ DISplay ─┬─ CALc key ON ─┬──────────────┬─ table-name ──────────►
   └─ PUNch ───┘               └─ schema-name.┘

►─┬──────────────────────────────────────────────────────────────────►
  │  ┌─ WITh ──────┬─ ALL ──────────────┬─ AS ┬─ COMments ◄ ─┬
  │  ├─ WITHOut ───┤  ├─ NONe ──────────┤     └─ SYNtax ─────┘
  │  ├─ ALSo WITh ─┤  ├─ DETails ───────┤
  │  └─ ALSo WITHOut┘  └─ HIStory ──────┘
  │
►─┴─ VERb ─┬─ CREate ◄ ─┬──────────────────────────────────────────►◄
           ├─ DISplay ──┤
           ├─ DROp ─────┤
           └─ PUNch ────┘
```

# 5.4 DISPLAY/PUNCH CONSTRAINT

```
►►─┬─ DISplay ─┬─── CONstraint constraint-name ────────────────────────────────►
   └─ PUNch ───┘

►─┬──────────────────────────────────────────────────────────────────────────┬─►
  └─────┬─ WITh ────────────┬─┬─ ALL ──────────────────── AS ─┬─ COMments ◄ ─┬┘
        ├─ WITHOut ─────────┤ ├─ NONe ──────────────────┤     └─ SYNtax ────┘
        ├─ ALSo WITh ───────┤ ├─ DETails ───────────────┤
        └─ ALSo WITHOut ────┘ └─ HIStory ───────────────┘

►─┬────────────────────────────────┬───────────────────────────────────────────►◄
  └─ VERb ─┬─ CREate ◄ ─────┬──────┘
           ├─ DISplay ──────┤
           ├─ DROp ─────────┤
           └─ PUNch ────────┘
```

# 5.5 DISPLAY/PUNCH INDEX

```
►►─┬─ DISplay ─┬─ INDex index-name ─ ON ─┬──────────────┬─ table-name ─►
   └─ PUNch ───┘                         └schema-name.──┘
   ►─┬──────────────────────────────────────────────────────────────────►
     └─┬─ WITh ──────────┬─┬─ ALL ──────────────┬─ AS ─┬─COMments ◄─┬─┘
       ├─ ALSo WITh ─────┤ ├─ NONe ─────────────┤      └─ SYNtax ───┘
       ├─ WITHOut ───────┤ ├─ DETails ──────────┤
       └─ ALSo WITHOut ──┘ ├─ HIStory ──────────┤
                           └─ TABles ───────────┘
   ►─┬─ VERB ─┬─ CREate ◄─┬─┘──────────────────────────────────────────►◄
     └        ├─ DISplay ─┤
              ├─ DROp ────┤
              └─ PUNch ───┘
```

# 5.6 DISPLAY/PUNCH KEY

```
►►─┬─ DISplay ─┬─ KEY key-name ON ─┬──────────────────┬─┬─ procedure-name ──────────┬►
   └─ PUNch ───┘                   └─ schema-name. ───┘ └─ table-procedure-name ────┘

►─┬──────────────────────────────────────────────────────────────────────────────────┬►
  └─┬─ WITh ──────────┬─┬─ ALL ──────────────┬─ AS ─┬─ COMments ◄ ─┬─┬──────────────┬──┬►◄
    ├─ ALSo WITh ─────┤ ├─ NONe ─────────────┤      └─ SYNtax ─────┘ │              │  │
    ├─ WITHOut ───────┤ ├─ DETails ──────────┤                       │              │  │
    └─ ALSo WITHOut ──┘ └─ HIStory ────────┬─┘                       │              │  │
                                           │                         │              │  │
                                           └─ VERb ─┬─ CREate ◄ ─┬─┬─┘              │  │
                                                    ├─ DISplay ──┤ │                │  │
                                                    ├─ DROp ─────┤ │                │  │
                                                    └─ PUNch ────┘ │                │  │
```

# 5.7 DISPLAY/PUNCH PROCEDURE

```
►►─┬─ DISplay ─┬─ PROcedure ─┬──────────────────┬─ procedure-name ──────►
   └─ PUNch ───┘             └─ schema-name. ───┘

►─┬──────────────────────────────────────────────────────────────────────►
  │  ┌─ WITh ──────────┬─ ALL ──────────────┬─ AS COMments ← ─┐
  │  ├─ ALSo WITh ─────┤  ├─ NONe ──────────┤  └─ SYNtax ─────┤
  │  ├─ WITHOut ───────┤  ├─ DETails ───────┤                 │
  │  └─ ALSo WITHOut ──┘  ├─ HIStory ───────┤                 │
  │                       └─ KEYs ──────────┘                 │
  └─────────────────────────────────────────────────────────────┘

►─┬──────────────────────────────────────────────────────────────────►◄
  │  ┌─ CREATE ← ─┐
  └─ VERb ─┬─ DISplay ─┤
           ├─ DROp ────┤
           └─ PUNch ───┘
```

# 5.8 DISPLAY/PUNCH SCHEMA

```
►►─┬─ DISplay ─┬──── SCHema name ─┬──────────────────────────────────────────►
   └─ PUNch ───┘                  ├─ FULl ─┤
                                  └─ ALL ──┘

►─┬──────────────────────────────────────────────────────────────────┬────►
  └──┬─ WITh ──────────┬──┬─ ALL ───────────────────┬──┬─ AS COMments ◄ ─┬─┘
     ├─ ALSo WITh ─────┤  ├─ NONe ──────────────────┤  └─ SYNtax ───────┘
     ├─ WITHOut ───────┤  ├─ DETails ───────────────┤
     └─ ALSo WITHOut ──┘  ├─ HIStory ───────────────┤
                          ├─ CALC keys ─────────────┤
                          ├─ CONstraints ───────────┤
                          ├─ INDexes ───────────────┤
                          ├─ KEYs ──────────────────┤
                          ├─ TABles ────────────────┤
                          ├─ TABle PROcedures ──────┤
                          └─ VIEws ─────────────────┘

►─┬─────────────────────────────┬──────────────────────────────────────────►◄
  └─ VERb ──┬─ CREate ◄ ──┬─────┘
            ├─ ALTer ─────┤
            ├─ DISplay ───┤
            ├─ DROp ──────┤
            └─ PUNch ─────┘
```

# 5.9 DISPLAY/PUNCH TABLE

```
►►─┬─ DISplay ─┬─── TABle ─┬──────────────┬─ table-name ─┬──────────┬──►
   └─ PUNch ───┘           └─ schema-name.┘              └─ FULl ───┘

►─┬──────────────────────────────────────────────────────────────────────►
  │  ┌─ WITh ──────┐  ┌─ ALL ──────────┐  ┌─ AS COMments ◄ ─┐
  │  ├─ WITHOut ───┤  ├─ NONe ─────────┤  └─ SYNtax ────────┘
  │  ├─ ALSo WITh ─┤  ├─ DETails ──────┤
  │  └─ ALSo WITHOut┘  ├─ HIStory ──────┤
  │                    ├─ CALc keys ────┤
  │                    ├─ COLumns ──────┤
  │                    ├─ CONstraints ──┤
  │                    ├─ INDexes ──────┤
  │                    └─ VIEws ────────┘

►─┬─ VERb ─┬─ CREate ◄ ─┬──────────────────────────────────────────────►◄
  │        ├─ ALTer ────┤
  │        ├─ DISplay ──┤
  │        ├─ DROp ─────┤
  │        └─ PUNch ────┘
```

*In IDD record format with COBOL elements*

```
►►─┬─ DISplay ─┬─── TABle ─┬──────────────┬─ table-name ─────────────────►
   └─ PUNch ───┘           └─ schema-name.┘

►── LIKe RECord ──────────────────────────────────────────────────────────►

►─┬──────────────────────────────────────────────────────────────────────►
  │  ┌─ WITh ──────┐  ┌─ ALL ──────────────┐  ┌─ AS COMments ◄ ─┐
  │  ├─ WITHOut ───┤  ├─ null INDIcators ──┤  └─ SYNtax ────────┘
  │  ├─ ALSo WITh ─┤  ├─ record ELEments ──┤
  │  └─ ALSo WITHOut┘  └─ record SYNonyms ──┘

►─┬─ VERb ─┬─ DISplay ◄ ─┬──────────────────────────────────────────────►◄
  │        ├─ ALTer ─────┤
  │        ├─ CREate ────┤
  │        ├─ DROp ──────┤
  │        └─ PUNch ─────┘
```

# 5.10 DISPLAY/PUNCH TABLE PROCEDURE

```
►►─┬─ DISplay ─┬─ TABle PROcedure ─┬──────────────────┬─ table-procedure-name ──►
   └─ PUNch ───┘                   └─ schema-name. ───┘

►─┬──────────────────────────────────────────────────────────────────────────►
  │ ┌─ WITh ──────────┐   ┌─ ALL ──────────┐   ┌─ AS COMments ◄ ─┐
  │ ├─ ALSo WITh ─────┤   ├─ NONe ─────────┤   └─ SYNtax ────────┘
  │ ├─ WITHOut ───────┤   ├─ DETails ──────┤
  │ └─ ALSo WITHOut ──┘   ├─ HIStory ──────┤
  │                       └─ KEYs ─────────┘
  │
►─┴─┬──────────────────────────────────────────────────────────────────────►◄
    │ ┌─ CREATE ◄ ─┐
    └─ VERb ─┬─ DISplay ─┤
             ├─ DROp ────┤
             └─ PUNch ───┘
```

# 5.11 DISPLAY/PUNCH VIEW

```
►►─┬─ DISplay ─┬─ VIEW ─┬──────────────┬─ view-name ──────────────►
   └─ PUNch ───┘        └─ schema-name.┘

►─┬──────────────────────────────────────────────────────────────►
  │  ┌─ WITh ──────┐  ┌─ ALL ──────────┐  ┌─ AS COMments ◄ ─┐
  │  ├─ WITHOut ───┤  ├─ NONe ─────────┤  └─ SYNtax ────────┘
  │  ├─ ALSo WITh ─┤  ├─ DETails ──────┤
  │  └─ ALSo WITHOut┘  └─ HIStory ───────┘
  │
  │  ┌──────────────────────────────────┐
  └─ VERb ─┬─ CREate ◄ ─┬──────────────►◄
           ├─ DISplay ──┤
           ├─ DROp ─────┤
           └─ PUNch ────┘
```

*In IDD record format with COBOL elements*

```
►►─┬─ DISplay ─┬─ VIEW ─┬──────────────┬─ view-name ──────────────►
   └─ PUNch ───┘        └─ schema-name.┘

►── LIKe RECord ──────────────────────────────────────────────────►

►─┬──────────────────────────────────────────────────────────────►
  │  ┌─ WITh ──────┐  ┌─ ALL ──────────┐  ┌─ AS COMments ◄ ─┐
  │  ├─ WITHOut ───┤  ├─ null INDIcators┤  └─ SYNtax ────────┘
  │  ├─ ALSo WITh ─┤  ├─ record ELEments┤
  │  └─ ALSo WITHOut┘  └─ record SYNonyms┘
  │
  │  ┌──────────────────────────────────┐
  └─ VERb ─┬─ CREate ◄ ─┬──────────────►◄
           ├─ ALTer ────┤
           ├─ DISplay ──┤
           ├─ DROp ─────┤
           └─ PUNch ────┘
```

# Chapter 6. Identifiers

### Entities with identifiers

```
Access modules          Areas
Columns                 RCMs (SQL statement modules)
Cursors                 Referential constraints
Segments                Schemas
Groups                  Tables
Indexes                 Users
                        Views
```

**Valid characters:** The first character of an identifier must be a letter, @, $, or #. A valid identifier consists of a combination of:

- Letters (A through Z and a through z)
- Digits (0 through 9)
- At sign (@)
- Dollar sign ($)
- Pound sign (#)
- Underscore (_)

### Identifier maximum length

| Entity type | Max length |
|---|---|
| Access modules | 8 characters |
| Segments | 8 characters |
| RCMs | 8 characters |
| RCMs | 8 characters |
| Column | 32 characters |
| All others | 18 characters |

# Chapter 7. Date/time values

### Internal representation

| Bits | Meaning |
|---|---|
| Bits 0 through 26 | Number of days since January 1, 0001 |
| Bits 27 through 43 | Number of seconds in the day since midnight |
| Bits 44 through 63 | Number of microseconds since the last second |

### External date values

| Standard | Format | Example |
|---|---|---|
| ISO | *yyyy-mm-dd* | 1991-12-15 |
| USA | *mm/dd/yyyy* | 12/15/1991 |
| EUR | *dd.mm.yyyy* | 15.12.1991 |
| JIS | *yyyy-mm-dd* | 1991-12-15 |

### External time values

| Standard | Format | Example |
|---|---|---|
| ISO | *hh.mm.ss* | 16.43.17 |
| USA | `hh:mm AM`<br>`hh:mm PM` | 4:43 PM |
| EUR | *hh.mm.ss* | 16.43.17 |
| JIS | *hh:mm:ss* | 16:43:17 |

### External timestamp format

`yyyy-mm-dd-hh.mm.ss.nnnnnn`

# Chapter 8. Null values

**Specifying a null value:** Use the keyword NULL to indicate a null value. For example:

```
insert into department
   (dept_id, dept_name, div_code, dept_head_id)
   values (4040, 'Audit', 'D09', null);
```

**Comparison and arithmetic operations:**

- The result of a **comparison operation** involving one or more null values is always unknown
- The result of an **arithmetic operation** involving one or more null values is always a null value

**Sort operations:** In a sort operation, a null value is a high value. Thus, a null is placed at the end of an ascending sort sequence.

# Chapter 9. CA-ADS

### Data types

| PICTURE and USAGE clause | CA-IDMS/DB data type |
|---|---|
| PIC X(*n*) USAGE DISPLAY | CHAR(*n*) |
| `01 name`<br>`  49 name-LEN  PIC S9(4) COMP`<br>`  49 name-TEXT  PIC X(n)` | VARCHAR(*n*) |
| PIC S9(*p-s*)V9(*s*) USAGE COMP-3 | DECIMAL(*p*,*s*) |
| PIC 9(*t-s*)V9(*s*) USAGE COMP-3 | UNSIGNED DECIMAL(*t*,*s*)[1] |
| USAGE COMP-2 | DOUBLE PRECISION |
| USAGE COMP-1 | REAL |
| USAGE COMP-1 | FLOAT |
| `PIC S9(n) USAGE COMP`<br>`(where n<5)` | SMALLINT |
| `PIC S9(n) USAGE COMP`<br>`(where n>4 and n<10)` | INTEGER |
| `PIC S9(n) USAGE COMP`<br>`(where n>9)` | LONGINT[1] |
| PIC S9(*p-s*)V9(*s*) USAGE DISPLAY | NUMERIC(*p*,*s*) |
| PIC 9(*t-s*)V9(*s*) USAGE DISPLAY | UNSIGNED NUMERIC(*t*,*s*)[1] |
| PIC X(*n*) USAGE SQLBIN | BINARY(*n*)[1] |
| PIC G(*n*) USAGE DISPLAY-1 | GRAPHIC(*n*)[1] |
| `01 name`<br>`  49 name-LEN  PIC S9(4) COMP`<br>`  49 name-TEXT  PIC G(n) DISPLAY-1` | VARGRAPHIC(*n*)[1] |
| PIC X(10) USAGE DISPLAY | DATE[1] |
| PIC X(8) USAGE DISPLAY | TIME[1] |
| PIC X(26) USAGE DISPLAY | TIMESTAMP[1] |

[1] This data type is a CA-IDMS/DB extension of ANSI-standard SQL.

# Chapter 10. COBOL

### Data types

| COBOL PICTURE and USAGE clause | CA-IDMS/DB data type |
|---|---|
| PIC X(*n*) USAGE DISPLAY | CHAR(*n*) |
| `01 name`<br>`  49 name-LEN  PIC S9(4) COMP`<br>`  49 name-TEXT  PIC X(n)` | VARCHAR(*n*) |
| PIC S9(*p-s*)V9(*s*) USAGE COMP-3 | DECIMAL(*p*,*s*) |
| PIC 9(*p-s*)V9(*s*) USAGE COMP-3 | UNSIGNED DECIMAL(*p*,*s*)[1] |
| USAGE COMP-2 | DOUBLE PRECISION |
| USAGE COMP-1 | REAL |
| USAGE COMP-1 | FLOAT |
| `PIC S9(n) USAGE COMP`<br>`(where n<5)` | SMALLINT |
| `PIC S9(n) USAGE COMP`<br>`(where n>4 and n<10)` | INTEGER |
| `PIC S9(n) USAGE COMP`<br>`(where n>9)` | LONGINT[1] |
| PIC S9(*p-s*)V9(*s*) USAGE DISPLAY | NUMERIC(*p*,*s*) |
| PIC 9(*p-s*)V9(*s*) USAGE DISPLAY | UNSIGNED NUMERIC(*p*,*s*)[1] |
| PIC X(*n*) USAGE SQLBIN | BINARY(*n*)[1] |
| PIC G(*n*) USAGE DISPLAY-1 | GRAPHIC(*n*)[1] |
| `01 name`<br>`  49 name-LEN  PIC S9(4) COMP`<br>`  49 name-TEXT  PIC G(n) DISPLAY-1` | VARGRAPHIC(*n*)[1] |
| PIC X(10) USAGE DISPLAY | DATE[1] |
| PIC X(8) USAGE DISPLAY | TIME[1] |
| PIC X(26) USAGE DISPLAY | TIMESTAMP[1] |

[1] This data type is a CA-IDMS/DB extension of ANSI-standard SQL.

### COBOL precompiler

```
►►──┬──────────────────────────────────────────────────────────►
    └─ COBOL = -─┬─ 1 -─┬
                 ├─ 2 -─┤
                 └─ 85 ─┘
```

### COPY IDMS FILE

```
►►── FILE SECTION. ──────────────────────────────────────────►

►─┬─────────────────────────────────────────────────────────┬──►◄
  ↓                                                         │
  └─ COPY IDMS FILE file-name ─┬──────────────────────────┬─ . ─┘
                               └─ VERSION version-number ─┘
```

### COPY IDMS RECORD

```
►►─┬─ WORKING-STORAGE SECTION. ─┬──────────────────────────────►
   └─ LINKAGE SECTION. ─────────┘

►─┬─────────────────────────────────────────────────────────┬──►◄
  ↓                                                         │
  │ ┌──────────────────────────────────────────────────────┐│
  └─┴─┬──────────────────────┬─ COPY IDMS RECORD record-specification ─┴─ . ─┘
      └─ level-number ───────┘
```

*Expansion of record-specification*

```
►►── record-name ─┬──────────────────────────────────────────►
                  └─ VERSION version-number ─┘

►─┬──────────────────────────────────────────────────────────►◄
  └─ REDEFINES record-name ─┘
```

### COPY IDMS MODULE

```
►►── PROCEDURE DIVISION. ──────────────────────────────────►

►─┬─────────────────────────────────────────────────────────┬──►◄
  ↓                                                         │
  └─ COPY IDMS module module-name ─┬──────────────────────┬─ . ─┘
                                   └─ VERSION version-number ─┘
```

# Chapter 11. PL/I

### Data types

| Equivalent PL/I data type | CA-IDMS/DB data type |
|---|---|
| CHAR (*n*) | CHAR(*n*) |
| CHAR (*n*) VAR | VARCHAR(*n*) |
| FIXED DECIMAL (*p*,*s*) where *p* <=15 and *s*<*p* | DECIMAL(*p*,*s*) |
| ▪ FLOAT BINARY (*n*) where 21<*n*<=53<br>▪ FLOAT DECIMAL (*n*) where 6<*n*<=16 | DOUBLE PRECISION |
| ▪ FLOAT BINARY (21) where precision <=24<br>▪ FLOAT BINARY (53) where precision >24 | FLOAT |
| ▪ FLOAT BINARY (*n*) where 1<*n*<=21<br>▪ FLOAT DECIMAL (*n*) where 1<*n*<=6 | REAL |
| FIXED BINARY (15) | SMALLINT |
| FIXED BINARY (31) | INTEGER |
| CHAR (*n*) | BINARY(*n*)[1] |
| GRAPHIC (*n*) | GRAPHIC(*n*)[1] |
| GRAPHIC (*n*) VAR | VARGRAPHIC(*n*)[1] |
| CHAR (10) | DATE[1] |
| CHAR (8) | TIME[1] |
| CHAR (26) | TIMESTAMP[1] |

[1] This data type is a CA-IDMS/DB extension of ANSI-standard SQL.

### SQLXQ1 ENTRY:

```
►►─┬─ DECLARE ─┬─ SQLXQ1 ENTRY OPTIONS (INTER, ASSEMBLER); ──────────────►◄
   └─ DCL ─────┘
```

### INCLUDE IDMS RECORD

```
►─┬─────────────────────────────────────────────────────────────────────┬─
  │↓                                                                    │
  │  ┌──────────────────────────────────────────────────────────────┐   ;
  │  │                                                              │
  └──┴──┬──────────────────────────────────────────────────────────┬──┘
        │                                                          │
        └─ level-number ─┬─ INCLUDE IDMS record-specification ─────┘
                         │
                         └──────────────────────────────────────────
```

*Expansion of record specification*

```
►►── record-name ──┬──────────────────────────┬──────────────────────►
                   └─ VERSION version-number ─┘

►──┬──────────────┬──────────────────────────────────────────────►◄
   └─ attribute ─┘
```

**INCLUDE IDMS MODULE**

```
►►─┬──────────────────────────────────────────────────────────────┬──►◄
   │                                                              │
   │   ┌──────────────────────────────────────────────────────┐   │
   └─ INCLUDE IDMS (module-name ──┬──────────────────────────┬── ); ─┘
                                  └─ VERSION version-number ─┘
```

# Chapter 12. Precompiler Options and Directives

### Overriding DDLDML area ready mode

```
►─┬──────────────────────┬──────────────────────────────────────────►◄
  ├─ *RETRIEVAL ─────────┤
  └─ *PROTECTED-UPDATE ──┘
```

### No logging of program activity statistics

```
►─┬──────────────────────┬──────────────────────────────────────────►◄
  └─ *NO-ACTIVITY-LOG ───┘
```

### Generating a source listing

```
►─┬──────────────────────┬──────────────────────────────────────────►◄
  ├─ *DMLIST ────────────┤
  └─ *NODMLIST ──────────┘
```

### Precompiler-option

```
►►─┬─ RCM = rcm-name ─┘──────────────────────────────────────►
   └

►─┬─ RCMVERSION = rcm-version-number ─┘──────────────────────►
  └

►─┬─ AM = access-module-name ─┘──────────────────────────────►
  └

►─┬─ SCHEMA = schema-name ─┘─────────────────────────────────►
  └

►─┬─ NOINSTALL ─┘────────────────────────────────────────────►
  └

►─┬─ DICTNAME = dictionary-name ─┬─┘──────────────────────────►
  └                              │
                                 └

►─┬─ SQL = -┬─ NO ──┬──────────────────────────────────────────►
  └         ├─ 89 ──┤
            └─ FIPS ─┘

►─┬─ LIST ──┬──────────────────────────────────────────────►
  ├─ NOList ─┘
  └

►─┬─ DATE = ┬─ ISO ─┬─┘──────────────────────────────────────►
  └         ├─ USA ─┤
            ├─ EUR ─┤
            └─ JIS ─┘

►─┬─ TIME = ┬─ ISO ─┬─┘──────────────────────────────────────►◄
  └         ├─ USA ─┤
            ├─ EUR ─┤
            └─ JIS ─┘
```

*COBOL precompiler only*

```
►►─┬─ COBOL = -┬─ 1 -┬─┘──────────────────────────────────────►
  └           ├─ 2 -┤
              └─ 85 ─┘
```

# Chapter 13. SQLCA

13.1 Structure . . . . . 13-3
13.2 SQLSTATE values . . . . . 13-6
13.3 SQLCODE error values . . . . . 13-9
13.4 SQLCODE and SQLCNRP values . . . . . 13-10

## 13.1 Structure

| Field | Meaning | Additional information |
|---|---|---|
| SQLCAID | Eye-catcher (SQLCA) | Initialized to 'SQLCA*'. |
| SQLCODE | SQL error code | |
| SQLCERC | Extended information error code | This field contains the reason code for error or warning conditions. |
| SQLCNRP | Number of rows processed by the SQL statement | |
| SQLCSER | Offset into the user-provided SQL statement buffer in which a syntax error was recognized | |
| SQLCLNO | Source file line number from which the SQL statement was obtained | This field is maintained by the precompiler and is provided for use in forming error messages. |
| SQLCMCT | Count of messages issued for this request | |
| SQLCARC | Reserved | |
| SQLCFJB | Reserved | |
| SQLCERRML | Length of error message text in SQLERRMC | |
| SQLERRMC | Text of the error messages | This is a 256-byte field containing one or more messages. Each message is preceded by a one-byte binary field containing the length of the message text. |
| SQLSTATE | SQL status code | |

### COBOL/CA-ADS SQLCA

```
01  SQLCA.
    02  SQLCAID                 PIC X(8).
    02  SQLCODE                 PIC S9(8) COMP.
    02  SQLCSID                 PIC X(8).
    02  SQLCINFO.
     03  SQLCERC                PIC S9(8) COMP.
     03  FILLER                 PIC S9(8) COMP.
     03  SQLCNRP                PIC S9(8) COMP.
     03  FILLER                 PIC S9(8) COMP.
     03  SQLCSER                PIC S9(8) COMP.
     03  FILLER                 PIC S9(8) COMP.
     03  SQLCLNO                PIC S9(8) COMP.
     03  SQLCMCT                PIC S9(8) COMP.
     03  SQLCARC                PIC S9(8) COMP.
     03  SQLCFJB                PIC S9(8) COMP.
     03  FILLER                 PIC S9(8) COMP.
     03  FILLER                 PIC S9(8) COMP.
    02  SQLCINF2 REDEFINES SQLCINFO.
     03  SQLERRD                PIC S9(8) COMP
                                OCCURS 12.
    02  SQLCMSG.
     03  SQLCERL                PIC S9(8) COMP.
     03  SQLERM                 PIC X(256).
    02  SQLCMSG2 REDEFINES SQLCMSG.
     03  FILLER                 PIC X(2).
     03  SQLERRM.
      04  SQLCERRML             PIC S9(4) COMP.
      04  SQLERRMC              PIC X(256).
    02  SQLSTATE                PIC X(5).
    02  FILLER                  PIC X(11).
                                          ┌──────────────────────
    02  SQLWORK                 PIC X(16).│
    02  SQLCWRK2 REDEFINES SQLWORK.       │
     03  SQLERRP.                         │
      04  SQLCVAL               PIC X(5). │  Included by the
      04  FILLER                PIC X(3). │  precompiler for
     03  SQLWARN.                         │  DB2 compatibility;
      04  SQLWARN0              PIC X(1). │  not used by CA-IDMS
      04  SQLWARN1              PIC X(1). │
      04  SQLWARN2              PIC X(1). │
      04  SQLWARN3              PIC X(1). │
      04  SQLWARN4              PIC X(1). │
      04  SQLWARN5              PIC X(1). │
      04  SQLWARN6              PIC X(1). │
      04  SQLWARN7              PIC X(1). │
                                          └──────────────────────
```

### PL/I SQLCA

```
DECLARE 1  SQLCA,
          2  SQLCAID          CHARACTER (8),
          2  SQLCODE          FIXED BINARY (31),
          2  SQLCSID          CHARACTER (8),
          2  SQLCINFO,
            3  SQLCERC        FIXED BINARY (31),
            3  FILLERnnnn     FIXED BINARY (31),
            3  SQLCNRP        FIXED BINARY (31),
            3  FILLERnnnn     FIXED BINARY (31),
            3  SQLCSER        FIXED BINARY (31),
            3  FILLERnnnn     FIXED BINARY (31),
            3  SQLCLNO        FIXED BINARY (31),
            3  SQLCMCT        FIXED BINARY (31),
            3  SQLCARC        FIXED BINARY (31),
            3  SQLCFJB        FIXED BINARY (31),
            3  FILLERnnnn     FIXED BINARY (31),
            3  FILLERnnnn     FIXED BINARY (31),
          2  SQLCMSG,
            3  SQLCERL        FIXED BINARY (31),
            3  SQLCERM        CHARACTER (256),
          2  SQLSTATE         CHARACTER (5),
          2  FILLERnnnn       CHARACTER (11),
          2  SQLWORK          CHARACTER (16) ;

DECLARE 1 SQLCINF2 BASED (ADDR(SQLCINFO)),
          2  SQLERRD   FIXED BINARY (31),

DECLARE 1 SQLCMSG2 BASED(ADDR(SQLCMSG)),
          2  FILLERnnnn        CHARACTER (2),
          2  SQLERRM,
            3  SQLERRML        CHARACTER (5),
            3  SQLERRMC        CHARACTER (256) ;

DECLARE 1 SQLCWRK2 BASED(ADDR(SQLWORK)),
          2  SQLERRP,
            3  SQLCVAL        CHARACTER (5),
            3  FILLERnnnn     CHARACTER (3),
          2  SQLWARN,
            3  SQLWARN0  CHARACTER (1),
            3  SQLWARN1  CHARACTER (1),
            3  SQLWARN2  CHARACTER (1),
            3  SQLWARN3  CHARACTER (1),
            3  SQLWARN4  CHARACTER (1),
            3  SQLWARN5  CHARACTER (1),
            3  SQLWARN6  CHARACTER (1),
            3  SQLWARN7  CHARACTER (1) ;
```

Included by the precompiler for DB2 compatibility; not used by CA-IDMS

# 13.2 SQLSTATE values

```
00 Successful completion
   000  No subclass

01 Warning
   000  No subclass
   004  String data, right truncation
   600  Inconsistent or invalid option
   602  Entity or association already exists
   605  Entity not defined in Catalog
   606  Invalid option for physical DDL
   607  Invalid option for DMCL
   608  Connecting to a dictionary which is missing either or
        or both of DDLCAT/DDLDML areas
   610  Database is inconsistent with request
   638  Warning returned from table procedure

02 No data
   000  No subclass

07 Dynamic SQL error
   000 No subclass
   001 USING clause does not match dynamic parameter specification
   002 USING clause does not match target specification
   003 Cursor specification cannot be executed
   004 USING clause required for dynamic parameters

08 Connection exception
   000  No subclass
   004  SQL-server rejected establishment of SQL-connection
   006  Connection failure

21 Cardinality violation
   000  No subclass

22 Data Exception
   000  No subclass
   001  String data, right truncation
   002  Null value, no indicator parameter
   003  Numeric value out of range
   005  Error in assignment
   007  Invalid datetime format
   008  Datetime field overflow
   011  Substring error
   012  Division by zero
   019  Invalid escape character

23 Constraint violation
   000  No subclass
   501  Duplicate key violation

24 Invalid cursor state
   000  No subclass

25 Invalid transaction state
   000  No subclass
   006  Read-only SQL-transaction

26 Invalid SQL statement name
   000  No subclass
```

```
28 Invalid authorization specification
   000  No subclass
   602  Entity or association already defined
   605  Entity or association not previously defined
   607  Authorization ids not specified

2C Invalid character set name
   000  No subclass

37 Syntax error or access rule violation
   000  No subclass

38 External routine exception
   000  No subclass

39 External routine invocation exception
   000  No subclass

3F Invalid schema name
   000  No subclass

40 Transaction rollback
   000  No subclass
   001  Serialization failure

42 Syntax error or access rule violation
   000  No subclass
   500  Table not found
   501  Column not found
   502  Entity already defined
   503  Authorization failure
   504  Cursor not declared or previously declared
   505  Entity not found
   506  Invalid identifier
   507  Keyword used as identifier
   600  Invalid statement
   601  Statement not valid in this context
   603  Statement not valid for this schema
   604  Invalid data type
   606  Invalid statement option
   607  Missing statement option
   609  Invalid constraint definition
   610  Invalid number of columns

50 CA-defined errors
   000  No subclass
   002  Limit exceeded
   003  Space exceeded
   00B  Internal error
   00I  Schema mismatch
   00J  Invalid entity definition
   00K  Uncategorized error
   00L  Invalid calling parameters

60 CA-IDMS specific errors
   000  No subclass
   001  Problem with load module or synchronization stamps
   002  Database error
   003  Rollback failed
```

```
64 CA-IDMS Physical DDL error
   000  No subclass

6U CA-IDMS Utility error
   000  No subclass
```

## 13.3 SQLCODE error values

| Value | Meaning |
|---|---|
| < 0 | The SQL statement returned an error (see error values below) |
| 0 | The SQL statement was executed successfully |
| 1 | The SQL statement was executed successfully with a warning |
| 100 | There are no more rows associated with the current query |

| Value | Level of failure | Meaning |
|---|---|---|
| -7 | Task | An internal error caused a task abend, leading to rollback and termination of the SQL transaction and termination of the SQL session |
| -6 | SQL session | An error has caused an SQL session failure, leading to rollback and termination of the SQL transaction and termination of the SQL session |
| -5 | SQL transaction | An error has caused an SQL transaction failure, leading to rollback and termination of the SQL transaction. |
| -4 | SQL statement | An error has caused a failure of the SQL statement to execute, leading to the rollback of the effect, if any, on the database |

# 13.4 SQLCODE and SQLCNRP values

### Bulk fetch return values

| Result of bulk fetch | SQLCODE value | SQLCNRP value |
|---|---|---|
| No rows are returned | 100 | 0 |
| At least one row is returned but fewer rows than the maximum allowed | 100 | Equals the number of rows returned |
| The number of rows returned matches the maximum allowed | 0 | Equals the number of rows returned |

### Bulk select return values

| Result of bulk select | SQLCODE value | SQLCNRP value |
|---|---|---|
| No rows are returned | 100 | 0 |
| At least one row is returned | 100 | Greater than 0 and less than or equal to the maximum allowed |
| The number of rows returned exceeds the maximum allowed | Less than 0 | Equal to the maximum allowed |

### Bulk insert return values

| Result of bulk insert | SQLCODE value | SQLCNRP value |
|---|---|---|
| Fewer rows than the number of rows specified are inserted because the insert failed on a row | Less than 0 | Equal to the relative row number of the failing row |
| The number of rows inserted matches the number of rows specified | 0 | Equal to the number of rows inserted |

# Chapter 14. SQLDA


14.1 Structure . . . . . 14-3
14.2 SQLLEN . . . . . 14-5
14.3 SQLTYPE . . . . . 14-6
14.4 SQLSCALE . . . . . 14-7
14.5 SQLPRECISION . . . . . 14-8
14.6 SQLALN and SQLNALN . . . . . 14-9
14.7 SQLNULL . . . . . 14-10

# 14.1 Structure

| Field | Data type | Meaning |
|---|---|---|
| SQLDAID | CHARACTER(8) | Set to SQLDA* on a DESCRIBE |
| SQLN | INTEGER | Maximum number of SQLVAR occurrences |
| SQLD | INTEGER | Actual number of SQLVAR occurrences:<br>▪ **0** — Not a SELECT statement<br>▪ **1 through SQLN** — Number of columns<br>▪ **Greater than SQLN** — Not enough SQLVAR entries |
| SQLVAR | | Structure occurring SQLN times |

### SQLVAR fields

| Field | Data type | Meaning |
|---|---|---|
| SQLLEN | INTEGER | Length<br>(Additional information provided below under "SQLLEN") |
| SQLTYPE | SMALLINT | Column data type<br>(Additional information provided below under "SQLTYPE") |
| SQLSCALE | SMALLINT | Scale (for exact numeric data types)<br>(Additional information provided below under "SQLSCALE") |
| SQLPRECISION | SMALLINT | Precision<br>(Additional information provided below under "SQLPRECISION") |
| SQLALN | SMALLINT | Data alignment flag<br>(Additional information provided below under "SQLALN and SQLNALN ") |
| SQLNALN | SMALLINT | Null indicator alignment flag<br>(Additional information provided below under "SQLALN and SQLNALN ") |

| Field | Data type | Meaning |
|---|---|---|
| SQLNULL | SMALLINT | Length of null indicator<br>(Additional information provided below under "SQLNULL") |
| SQLNAME | CHARACTER(32) | Column name |

## 14.2 SQLLEN

| Data type | Definition of SQLLEN |
|---|---|
| CHARACTER | Number of characters |
| VARCHAR | Maximum number of characters |
| DATETIME | Length of the date/time column |
| BINARY | Number of bytes |
| GRAPHIC | Number of two-byte characters |
| VARGRAPHIC | Maximum number of two-byte characters |

# 14.3 SQLTYPE

| Code | Meaning |
|---|---|
| 1 | CHARACTER |
| 2 | NUMERIC |
| 3 | DECIMAL |
| 4 | INTEGER (four bytes) |
| 5 | SMALLINT (two bytes) |
| 6 | FLOAT |
| 7 | REAL |
| 8 | DOUBLE PRECISION |
| 9 | Datetime (DATE, TIME, or TIMESTAMP) |
| 10 | Reserved |
| 11 | Reserved |
| 12 | VARCHAR |

Current CA-IDMS extensions are:

| Code | Meaning |
|---|---|
| 128 | NUMERIC (UNSIGNED) |
| 129 | DECIMAL (UNSIGNED) |
| 130 | LONGINT (eight bytes) |
| 131 | Reserved |
| 132 | GRAPHIC |
| 133 | VARGRAPHIC |
| 134 | Reserved for precompiler use (GROUP) |
| 135 | BINARY/SQLBIN |
| 136 | Reserved |
| 140 | Reserved |
| 141 | Filler field (entry is used to adjust subsequent alignments but is otherwise ignored) |
| 142 | Reserved for precompiler use (SQLIND) |

## 14.4 SQLSCALE

| Data type | Definition of SQLSCALE |
|---|---|
| Exact numeric | Scale. |
| Datetime | Code indicating the precision of the date/time value. Permissible values in the first implementation are:<br>**1** for DATE<br>**2** for TIME<br>**3** for TIMESTAMP |

# 14.5 SQLPRECISION

| Data type | Definition of SQLPRECISION |
|---|---|
| Exact numeric | Precision. |
| Approximate numeric | Precision. |
| Datetime | Precision of the fraction field (0 or 6), if applicable.<br>If passed on a dynamic FETCH statement, SQLPRECISION is ignored for date/time data types. |

## 14.6 SQLALN and SQLNALN

SQLALN and SQLNALN are flags that indicate whether data (SQLALN) and the null indicator (SQLNALN) are aligned:

- **0** — Not aligned.
- **1** — Aligned based on natural alignment rules for the platform. For example, on both the System/370 and VMS environments, the natural alignment for the INTEGER data type is on a four-byte boundary.

## 14.7 SQLNULL

- **0** — No null indicator
- **1, 2, 4** — Null indicator length

A DESCRIBE statement will return 4 if nulls are allowed and 0 otherwise.

# Chapter 15. SYSTEM Tables


15.1 SYSTEM.AM . . . . . 15-3
15.2 SYSTEM.AMDEP . . . . . 15-4
15.3 SYSTEM.AREA . . . . . 15-5
15.4 SYSTEM.BUFFER . . . . . 15-6
15.5 SYSTEM.COLUMN . . . . . 15-7
15.6 SYSTEM.CONSTKEY . . . . . 15-8
15.7 SYSTEM.CONSTRAINT . . . . . 15-9
15.8 SYSTEM.DBNAME . . . . . 15-11
15.9 SYSTEM.DBSEGMENT . . . . . 15-12
15.10 SYSTEM.DBSSC . . . . . 15-13
15.11 SYSTEM.DBTABLE . . . . . 15-14
15.12 SYSTEM.DMCL . . . . . 15-15
15.13 SYSTEM.DMCLAREA . . . . . 15-16
15.14 SYSTEM.DMCLFILE . . . . . 15-17
15.15 SYSTEM.DMCLSEGMENT . . . . . 15-18
15.16 SYSTEM.FILE . . . . . 15-19
15.17 SYSTEM.FILEMAP . . . . . 15-20
15.18 SYSTEM.INDEX . . . . . 15-21
15.19 SYSTEM.INDEXKEY . . . . . 15-23
15.20 SYSTEM.JOURNAL . . . . . 15-24
15.21 SYSTEM.LOADHDR . . . . . 15-25
15.22 SYSTEM.ORDERKEY . . . . . 15-26
15.23 SYSTEM.SCHEMA . . . . . 15-27
15.24 SYSTEM.SECTION . . . . . 15-28
15.25 SYSTEM.SEGMENT . . . . . 15-29
15.26 SYSTEM.SYMBOL . . . . . 15-30
15.27 SYSTEM.SYNTAX . . . . . 15-31
15.28 SYSTEM.TABLE . . . . . 15-32
15.29 SYSTEM.VIEWDEP . . . . . 15-34
15.30 NOTES . . . . . 15-35

## 15.1 SYSTEM.AM

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(8) | NOT NULL |
| VERSION | SMALLINT | NOT NULL |
| SCHEMA | CHAR(18) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| LENGTH | INTEGER | NOT NULL |
| FILLER | BINARY(20) | NOT NULL |

## 15.2 SYSTEM.AMDEP

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(8) | NOT NULL |
| VERSION | SMALLINT | NOT NULL |
| TABSCHEMA | CHAR(18) | NOT NULL |
| TABLE | CHAR(18) | NOT NULL |
| STAMP | TIMESTAMP | NOT NULL |
| TYPE | CHAR(1) | NOT NULL |
| FILLER | BINARY(25) | NOT NULL |

## 15.3 SYSTEM.AREA

| Column name | Data type | Null specification |
|---|---|---|
| SEGMENT | CHAR(8) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| TIMESTAMP | TIMESTAMP | NOT NULL |
| CUSER | TIMESTAMP | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| TYPE | CHAR(1) | NOT NULL |
| STAMPLEVEL | CHAR(1) | NOT NULL |
| NUMFILEMAPS | SMALLINT | NOT NULL |
| NUMSYMBOLICS | SMALLINT | NOT NULL |
| DISPLACEMENT | SMALLINT | NOT NULL |
| PAGEGROUP | SMALLINT | NOT NULL |
| LOWPAGE | INTEGER | NOT NULL |
| HIGHPAGE | INTEGER | NOT NULL |
| CALCHIGHPAGE | INTEGER | NOT NULL |
| MAXHIGHPAGE | INTEGER | NOT NULL |
| NUMPAGES | INTEGER | NOT NULL |
| PAGESIZE | INTEGER | NOT NULL |
| PAGERESERVE | INTEGER | NOT NULL |
| ORIGPAGESIZE | INTEGER | NOT NULL |
| NUMPAGESUSED | INTEGER | NOT NULL |
| NUMROWS | INTEGER | NOT NULL |
| PCTSPACEUSED | REAL | NOT NULL |
| FILLER | BINARY(40) | NOT NULL |

# 15.4 SYSTEM.BUFFER

| Column name | Data type | Null specification |
|---|---|---|
| DMCL | CHAR(8) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| TYPE | CHAR(2) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| PAGESIZE | INTEGER | NOT NULL |
| LOCALPAGES | INTEGER | NOT NULL |
| CVPAGES | INTEGER | NOT NULL |
| MAXPAGES | INTEGER | NOT NULL |
| KEYLENGTH | SMALLINT | NOT NULL |
| BUFNI | SMALLINT | NOT NULL |
| STRNO | SMALLINT | NOT NULL |
| STGFLAG | BINARY(1) | NOT NULL |
| FLAG | BINARY(1) | NOT NULL |
| FILLER | BINARY(40) | NOT NULL |

## 15.5 SYSTEM.COLUMN

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(32) | NOT NULL |
| NUMBER | SMALLINT | NOT NULL |
| SCHEMA | CHAR(18) | NOT NULL |
| TABLE | CHAR(18) | NOT NULL |
| TYPE | CHAR(18) | NOT NULL |
| TYPECODE | SMALLINT | NOT NULL |
| PRECISION | SMALLINT | NOT NULL |
| SCALE | SMALLINT | NOT NULL |
| NULLS | CHAR(1) | NOT NULL |
| DEFAULT | CHAR(1) | NOT NULL |
| VOFFSET | SMALLINT | NOT NULL |
| VLENGTH | SMALLINT | NOT NULL |
| NOFFSET | SMALLINT | NOT NULL |
| NLENGTH | SMALLINT | NOT NULL |
| NUMVALUES | INTEGER | NOT NULL |
| SECLOWVAL | BINARY(8) | NOT NULL |
| SECHIGHVAL | BINARY(8) | NOT NULL |
| PROCPARMTYPE | CHAR(1) | NOT NULL |
| FILLER | BINARY(39) | NOT NULL |

# 15.6 SYSTEM.CONSTKEY

| Column name | Data type | Null specification |
|---|---|---|
| SCHEMA | CHAR(18) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| SEQUENCE | SMALLINT | NOT NULL |
| REFNUMBER | SMALLINT | NOT NULL |
| REFCOLUMN | CHAR(32) | NOT NULL |
| NUMBER | SMALLINT | NOT NULL |
| COLUMN | CHAR(32) | NOT NULL |
| FILLER | BINARY(38) | NOT NULL |

## 15.7 SYSTEM.CONSTRAINT

| Column name | Data type | Null specification |
|---|---|---|
| SCHEMA | CHAR(18) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| REFSCHEMA | CHAR(18) | NOT NULL |
| REFTABLE | CHAR(18) | NOT NULL |
| TABSCHEMA | CHAR(18) | NOT NULL |
| TABLE | CHAR(18) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| NUMCOLUMNS | SMALLINT | NOT NULL |
| REFCOLUMNS | BINARY(64) | NOT NULL |
| COLUMNS | BINARY(64) | NOT NULL |
| NUMSORTCOLS | SMALLINT | NOT NULL |
| SORTCOLUMNS | BINARY(64) | NOT NULL |
| SORTORDER | CHAR(32) | NOT NULL |
| CLUSTER | CHAR(1) | NOT NULL |
| UNIQUE | CHAR(1) | NOT NULL |
| TYPE | CHAR(1) | NOT NULL |
| COMPRESS | CHAR(1) | NOT NULL |
| IXBLKLENGTH | SMALLINT | NOT NULL |
| IXBLKCONTAINS | SMALLINT | NOT NULL |
| DISPLACEMENT | SMALLINT | NOT NULL |
| REFNEXT | SMALLINT | NOT NULL |
| REFPRIOR | SMALLINT | NOT NULL |
| NEXT | SMALLINT | NOT NULL |
| PRIOR | SMALLINT | NOT NULL |
| OWNER | SMALLINT | NOT NULL |
| NUMSETS | INTEGER | NOT NULL |
| AVGMEMROWS | REAL | NOT NULL |
| LONGESTMEM | INTEGER | NOT NULL |
| SECLONGMEM | INTEGER | NOT NULL |

| Column name | Data type | Null specification |
|---|---|---|
| NUMLONGMEM | INTEGER | NOT NULL |
| AVGMEMPAGES | REAL | NOT NULL |
| MAXMEMPAGES | INTEGER | NOT NULL |
| AVGAMEMCLUSCNT | REAL | NOT NULL |
| AVGBMEMCLUSCNT | REAL | NOT NULL |
| AVGCMEMCLUSCNT | REAL | NOT NULL |
| AVGDMEMCLUSCNT | REAL | NOT NULL |
| AVGEMEMCLUSCNT | REAL | NOT NULL |
| AVGSR8ROWS | REAL | NOT NULL |
| LONGESTSR8 | INTEGER | NOT NULL |
| SECLONGSR8 | INTEGER | NOT NULL |
| NUMLONGSR8 | INTEGER | NOT NULL |
| AVGSR8PAGES | REAL | NOT NULL |
| MAXSR8PAGES | INTEGER | NOT NULL |
| AVGSR8LEAFS | REAL | NOT NULL |
| AVGSR8LEVELS | REAL | NOT NULL |
| AVGASR8CLUSCNT | REAL | NOT NULL |
| AVGBSR8CLUSCNT | REAL | NOT NULL |
| AVGCSR8CLUSCNT | REAL | NOT NULL |
| AVGDSR8CLUSCNT | REAL | NOT NULL |
| AVGESR8CLUSCNT | REAL | NOT NULL |
| FILLER | BINARY(40) | NOT NULL |

## 15.8 SYSTEM.DBNAME

| Column name | Data type | Null specification |
|---|---|---|
| DBTABLE | CHAR(8) | NOT NULL |
| NAME | CHAR(8) | NOT NULL |
| DMCL | CHAR(8) | NOT NULL |
| NODE | CHAR(8) | NOT NULL |
| LOADLIST | CHAR(8) | NOT NULL |
| FLAG | BINARY(1) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| GROUPFLAG | CHAR(1) | NOT NULL |
| NUMSEGMENTS | SMALLINT | NOT NULL |
| NUMSUBSCHEMAS | SMALLINT | NOT NULL |
| FILLER | BINARY(18) | NOT NULL |

## 15.9 SYSTEM.DBSEGMENT

| Column name | Data type | Null specification |
|---|---|---|
| DBTABLE | CHAR(8) | NOT NULL |
| DBNAME | CHAR(8) | NOT NULL |
| NAME | CHAR(8) | NOT NULL |
| FILLER | BINARY(20) | NOT NULL |

## 15.10 SYSTEM.DBSSC

| Column name | Data type | Null specification |
|---|---|---|
| DBTABLE | CHAR(8) | NOT NULL |
| DBNAME | CHAR(8) | NOT NULL |
| FROMSSC | CHAR(8) | NOT NULL |
| TOSSC | CHAR(8) | NOT NULL |
| FOREIGNDBNAME | CHAR(8) | NOT NULL |
| FILLER | BINARY(20) | NOT NULL |

# 15.11 SYSTEM.DBTABLE

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(8) | NOT NULL |
| CVSYSTEM | SMALLINT | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| NUMDBNAMES | SMALLINT | NOT NULL |
| NUMSEGMENTS | SMALLINT | NOT NULL |
| NUMSUBSCHEMAS | SMALLINT | NOT NULL |
| NUMGROUPS | SMALLINT | NOT NULL |
| FILLER | BINARY(18) | NOT NULL |

## 15.12 SYSTEM.DMCL

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(8) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| BUFFER | CHAR(18) | NOT NULL |
| DBTABLE | CHAR(8) | NOT NULL |
| NUMBUFFERS | SMALLINT | NOT NULL |
| NUMJRNLBUFFERS | SMALLINT | NOT NULL |
| NUMJOURNALS | SMALLINT | NOT NULL |
| SHAREDCACHE | CHAR(16) | NOT NULL |
| LOCKENTRIES | INTEGER | NOT NULL |
| MEMBERS | BINARY(1) | NOT NULL |
| DATASHARE | CHAR(1) | NOT NULL |
| ONCONNECTLOSS | CHAR(1) | NOT NULL |
| FILLER | CHAR(17) | NOT NULL |

# 15.13 SYSTEM.DMCLAREA

| Column name | Data type | Null specification |
|---|---|---|
| DMCL | CHAR(8) | NOT NULL |
| SEGMENT | CHAR(8) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| STARTUP | CHAR(1) | NOT NULL |
| WARMSTART | CHAR(1) | NOT NULL |
| PAGERESERVE | INTEGER | NOT NULL |
| DATASHARE | CHAR(1) | NOT NULL |
| FILLER | CHAR(39) | NOT NULL |

## 15.14 SYSTEM.DMCLFILE

| Column name | Data type | Null specification |
|---|---|---|
| DMCL | CHAR(8) | NOT NULL |
| SEGMENT | CHAR(8) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| BUFFER | CHAR(18) | NOT NULL |
| DDNAME | CHAR(8) | NOT NULL |
| DISP | CHAR(4) | NOT NULL |
| DATASPACE | CHAR(1) | NOT NULL |
| ESAREAD | CHAR(1) | NOT NULL |
| ESAPRELOAD | CHAR(1) | NOT NULL |
| SHAREDCACHE | CHAR(16) | NOT NULL |
| FILLER | BINARY(21) | NOT NULL |

# 15.15 SYSTEM.DMCLSEGMENT

| Column name | Data type | Null specification |
|---|---|---|
| DMCL | CHAR(8) | NOT NULL |
| NAME | CHAR(8) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| BUFFER | CHAR(18) | NOT NULL |
| STARTUP | CHAR(1) | NOT NULL |
| WARMSTART | CHAR(1) | NOT NULL |
| DATASHARE | CHAR(1) | NOT NULL |
| SHAREDCACHE | CHAR(16) | NOT NULL |
| FILLER | CHAR(23) | NOT NULL |

## 15.16 SYSTEM.FILE

| Column name | Data type | Null specification |
|---|---|---|
| SEGMENT | CHAR(8) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| NUMFILEMAPS | SMALLINT | NOT NULL |
| BLOCKSIZE | INTEGER | NOT NULL |
| DDNAME | CHAR(8) | NOT NULL |
| ACCESSMETHOD | CHAR(8) | NOT NULL |
| VMUSERID | CHAR(8) | NOT NULL |
| VMVIRTADDR | INTEGER | NOT NULL |
| FLAG | BINARY(1) | NOT NULL |
| NVSAMSET | CHAR(18) | NOT NULL |
| DSNAME | CHAR(54) | NOT NULL |
| DISP | CHAR(4) | NOT NULL |
| FILLER | BINARY(39) | NOT NULL |

# 15.17 SYSTEM.FILEMAP

| Column name | Data type | Null specification |
|---|---|---|
| SEGMENT | CHAR(8) | NOT NULL |
| AREA | CHAR(18) | NOT NULL |
| FILE | CHAR(18) | NOT NULL |
| PAGESIZE | INTEGER | NOT NULL |
| LOWPAGE | INTEGER | NOT NULL |
| HIGHPAGE | INTEGER | NOT NULL |
| LOWBLOCK | INTEGER | NOT NULL |
| HIGHBLOCK | INTEGER | NOT NULL |
| ACCESSMETHOD | CHAR(8) | NOT NULL |
| FLAG | BINARY(1) | NOT NULL |
| FILLER | BINARY(39) | NOT NULL |

# 15.18 SYSTEM.INDEX

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(18) | NOT NULL |
| SCHEMA | CHAR(18) | NOT NULL |
| TABLE | CHAR(18) | NOT NULL |
| SEGMENT | CHAR(8) | NOT NULL |
| AREA | CHAR(18) | |
| INDEXID | SMALLINT | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| NUMCOLUMNS | SMALLINT | NOT NULL |
| IXCOLUMNS | BINARY(64) | NOT NULL |
| IXORDERS | CHAR(32) | NOT NULL |
| UNIQUE | CHAR(1) | NOT NULL |
| CLUSTER | CHAR(1) | NOT NULL |
| COMPRESS | CHAR(1) | NOT NULL |
| IXBLKLENGTH | SMALLINT | NOT NULL |
| IXBLKCONTAINS | SMALLINT | NOT NULL |
| DISPLACEMENT | SMALLINT | NOT NULL |
| NEXT | SMALLINT | NOT NULL |
| PRIOR | SMALLINT | NOT NULL |
| NUMSETS | INTEGER | NOT NULL |
| AVGMEMROWS | REAL | NOT NULL |
| LONGESTMEM | INTEGER | NOT NULL |
| SECLONGMEM | INTEGER | NOT NULL |
| NUMLONGMEM | INTEGER | NOT NULL |
| AVGMEMPAGES | REAL | NOT NULL |
| MAXMEMPAGES | INTEGER | NOT NULL |
| AVGAMEMCLUSCNT | REAL | NOT NULL |
| AVGBMEMCLUSCNT | REAL | NOT NULL |
| AVGCMEMCLUSCNT | REAL | NOT NULL |
| AVGDMEMCLUSCNT | REAL | NOT NULL |

| Column name | Data type | Null specification |
|---|---|---|
| AVGEMEMCLUSCNT | REAL | NOT NULL |
| AVGSR8ROWS | REAL | NOT NULL |
| LONGESTSR8 | INTEGER | NOT NULL |
| SECLONGSR8 | INTEGER | NOT NULL |
| NUMLONGSR8 | INTEGER | NOT NULL |
| AVGSR8PAGES | REAL | NOT NULL |
| MAXSR8PAGES | INTEGER | NOT NULL |
| AVGSR8LEAFS | REAL | NOT NULL |
| AVGSR8LEVELS | REAL | NOT NULL |
| AVGASR8CLUSCNT | REAL | NOT NULL |
| AVGBSR8CLUSCNT | REAL | NOT NULL |
| AVGCSR8CLUSCNT | REAL | NOT NULL |
| AVGDSR8CLUSCNT | REAL | NOT NULL |
| AVGESR8CLUSCNT | REAL | NOT NULL |
| NUMUNIQKEYS | INTEGER | NOT NULL |
| NUMNULLKEYS | INTEGER | NOT NULL |
| NUMLONGKEYS | INTEGER | NOT NULL |
| AVGDUPSPERKEY | REAL | NOT NULL |
| AVGPAGESPERKEY | REAL | NOT NULL |
| PROCKEY | CHAR(1) | NOT NULL |
| PRIMEKEY | CHAR(1) | NOT NULL |
| FILLER | BINARY(38) | NOT NULL |

## 15.19 SYSTEM.INDEXKEY

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(18) | NOT NULL |
| SCHEMA | CHAR(18) | NOT NULL |
| TABLE | CHAR(18) | NOT NULL |
| SEQUENCE | SMALLINT | NOT NULL |
| NUMBER | SMALLINT | NOT NULL |
| COLUMN | CHAR(32) | NOT NULL |
| SORTORDER | CHAR(1) | NOT NULL |
| FILLER | BINARY(37) | NOT NULL |

# 15.20 SYSTEM.JOURNAL

| Column name | Data type | Null specification |
|---|---|---|
| DMCL | CHAR(8) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| TYPE | CHAR(4) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| FILLER | CHAR(2) | NOT NULL |
| NUMBLOCKS | INTEGER | NOT NULL |
| BLOCKSIZE | INTEGER | NOT NULL |
| DDNAME | CHAR(8) | NOT NULL |
| ACCESSMETHOD | CHAR(8) | NOT NULL |
| DATASPACE | CHAR(1) | NOT NULL |
| ESAREAD | CHAR(1) | NOT NULL |
| ESAPRELOAD | CHAR(1) | NOT NULL |
| FILLER | BINARY(41) | NOT NULL |

## 15.21 SYSTEM.LOADHDR

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(8) | NOT NULL |
| VERSION | SMALLINT | NOT NULL |
| RLDS | SMALLINT | NOT NULL |
| EPA | BINARY(4) | NOT NULL |
| LENGTH | INTEGER | NOT NULL |
| DATE | CHAR(8) | NOT NULL |
| TIME | CHAR(6) | NOT NULL |
| TYPE | BINARY(1) | NOT NULL |
| SEC | BINARY(1) | NOT NULL |
| STLENGTH | INTEGER | NOT NULL |
| MODE | BINARY(1) | NOT NULL |
| STLEVEL | BINARY(1) | NOT NULL |
| SCHEMA | CHAR(18) | NOT NULL |

# 15.22 SYSTEM.ORDERKEY

| Column name | Data type | Null specification |
|---|---|---|
| SCHEMA | CHAR(18) | NOT NULL |
| CONSTRAINT | CHAR(18) | NOT NULL |
| SEQUENCE | SMALLINT | NOT NULL |
| NUMBER | SMALLINT | NOT NULL |
| COLUMN | CHAR(32) | NOT NULL |
| SORTORDER | CHAR(1) | NOT NULL |
| FILLER | BINARY(39) | NOT NULL |

## 15.23 SYSTEM.SCHEMA

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(18) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| TYPE | CHAR(1) | NOT NULL |
| SEGMENT | CHAR(8) | NOT NULL |
| AREA | CHAR(18) | NOT NULL |
| NODE | CHAR(8) | NOT NULL |
| DICTIONARY | CHAR(8) | NOT NULL |
| NTWKSCHEMA | CHAR(8) | NOT NULL |
| VERSION | SMALLINT | NOT NULL |
| CHARSET | CHAR(18) | NOT NULL |
| FILLER | BINARY(22) | NOT NULL |

# 15.24 SYSTEM.SECTION

| Column name | Data type | Null specification |
|---|---|---|
| SCHEMA | CHAR(18) | NOT NULL |
| TABLE | CHAR(18) | NOT NULL |
| TYPE | CHAR(1) | NOT NULL |
| FORMAT | CHAR(1) | NOT NULL |
| SEQUENCE | SMALLINT | NOT NULL |
| TEXT | BINARY(512) | NOT NULL |

# 15.25 SYSTEM.SEGMENT

| Column name | Data type | Null specification |
|---|---|---|
| NAME | CHAR(8) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| SCHEMA | CHAR(18) | NOT NULL |
| PAGEGROUP | SMALLINT | NOT NULL |
| RECSPERPAGE | INTEGER | NOT NULL |
| NUMAREAS | SMALLINT | NOT NULL |
| NUMFILES | SMALLINT | NOT NULL |
| NUMDADS | SMALLINT | NOT NULL |
| NUMFILEMAPS | SMALLINT | NOT NULL |
| NUMSYMBOLICS | SMALLINT | NOT NULL |
| STAMPLEVEL | CHAR(1) | NOT NULL |
| TYPE | CHAR(1) | NOT NULL |
| FILLER | BINARY(40) | NOT NULL |

# 15.26 SYSTEM.SYMBOL

| Column name | Data type | Null specification |
|---|---|---|
| TYPE | SMALLINT | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| SEGMENT | CHAR(8) | NOT NULL |
| AREA | CHAR(18) | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CRITTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| FLAG | BINARY(1) | NOT NULL |
| FILLER | CHAR(1) | NOT NULL |
| VALUE1 | INTEGER | NOT NULL |
| VALUE2 | INTEGER | NOT NULL |
| FILLER | BINARY(40) | NOT NULL |

## 15.27 SYSTEM.SYNTAX

| Column name | Data type | Null specification |
|---|---|---|
| SCHEMA | CHAR(18) | NOT NULL |
| TABLE | CHAR(18) | NOT NULL |
| TYPE | CHAR(1) | NOT NULL |
| SEQUENCE | SMALLINT | NOT NULL |
| SYNTAX | CHAR(80) | NOT NULL |

# 15.28 SYSTEM.TABLE

| Column name | Data type | Null specification |
|---|---|---|
| SCHEMA | CHAR(18) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| SEGMENT | CHAR(8) | |
| AREA | CHAR(18) | |
| TABLEID | SMALLINT | NOT NULL |
| TYPE | CHAR(1) | NOT NULL |
| LOCMODE | CHAR(1) | NOT NULL |
| COMPRESS | CHAR(1) | NOT NULL |
| FORMAT | CHAR(1) | NOT NULL |
| UPDATABLE | CHAR(1) | NOT NULL |
| CHECKOPT | CHAR(1) | NOT NULL |
| TIMESTAMP | TIMESTAMP | NOT NULL |
| CTIME | TIMESTAMP | NOT NULL |
| UTIME | TIMESTAMP | NOT NULL |
| CUSER | CHAR(18) | NOT NULL |
| UUSER | CHAR(18) | NOT NULL |
| PUTROUTINE | CHAR(8) | NOT NULL |
| GETROUTINE | CHAR(8) | NOT NULL |
| LENGTH | SMALLINT | NOT NULL |
| DATALENGTH | SMALLINT | NOT NULL |
| PREFIXLENGTH | SMALLINT | NOT NULL |
| CTRLENGTH | SMALLINT | NOT NULL |
| FIXLENGTH | SMALLINT | NOT NULL |
| SECLENGTH | SMALLINT | NOT NULL |
| NUMSYNTAX | SMALLINT | NOT NULL |
| NUMCOLS | SMALLINT | NOT NULL |
| NUMINDEXES | SMALLINT | NOT NULL |
| NUMREFERENCED | SMALLINT | NOT NULL |
| NUMREFERENCING | SMALLINT | NOT NULL |

| Column name | Data type | Null specification |
|---|---|---|
| DISPLACEMENT | INTEGER | NOT NULL |
| ESTROWS | INTEGER | NOT NULL |
| NUMPAGES | INTEGER | NOT NULL |
| NUMROWS | INTEGER | NOT NULL |
| ROWSPERPAGE | INTEGER | NOT NULL |
| AVGROWLENGTH | REAL | NOT NULL |
| PCTSPACEUSED | REAL | NOT NULL |
| PCTFRAGROWS | REAL | NOT NULL |
| NUMIO01 | REAL | NOT NULL |
| NUMIO03 | REAL | NOT NULL |
| NUMIO05 | REAL | NOT NULL |
| NUMIO10 | REAL | NOT NULL |
| NUMIO20 | REAL | NOT NULL |
| PROCMODE | CHAR(1) | NOT NULL |
| SECLENGTH2 | INTEGER | NOT NULL |
| FILLER | BINARY(22) | NOT NULL |

# 15.29 SYSTEM.VIEWDEP

| Column name | Data type | Null specification |
|---|---|---|
| SCHEMA | CHAR(18) | NOT NULL |
| NAME | CHAR(18) | NOT NULL |
| SEQUENCE | SMALLINT | NOT NULL |
| REFSCHEMA | CHAR(18) | NOT NULL |
| REFTABLE | CHAR(18) | NOT NULL |
| REFTYPE | CHAR(1) | NOT NULL |
| FILLER | BINARY(5) | NOT NULL |

## 15.30 NOTES