

# Index

## SYMBOLS AND NUMERICS

### \* (asterisk)

password, replacing with asterisk display in view, 646  
text, replacing trimmed with, 145, 191  
wildcard character, 60, 69, 103, 532

### @ (at sign)

procedure argument prefix, 298  
UDF parameter prefix, 479  
variable prefix, 298

### \ (backslash)

escape character, 114, 532  
, function character set prefix, 114

### || (bars) concatenation operator, 68

### [ ] (brackets, square)

PostgreSQL array delimiters, 532  
validation string delimiters, 192  
wildcard text delimiters, 253

### ^ (caret)

wildcard character, 253  
XOR operator, 584

### : (colon) array slice indicator, 532

### , (comma)

list element separator, 532  
string set, comma-delimited, 341, 342, 346

### \$ (dollar sign) function positional argument indicator, 532

### # (number sign)

temporary table prefix, 298  
UDF name prefix, 479

### ( ) (parentheses) expression group delimiters, 532

### % (percent sign)

modulo operator, 71, 78  
wildcard character, 253

### . (period) UDT attribute indicator, 532

### + (plus sign) concatenation operator, 68

### ? (question mark) Java parameter placemaker, 612

### "" (quotation marks, double) string delimiters, 192, 384

### ' ' (quotation marks, single)

escape delimiters, 532  
string delimiters, 192, 384

### ; (semicolon) command suffix, 532

### \_ (underscore)

UDF name prefix, 479  
wildcard character, 253

### 3GL (third-generation language)

ESQL, using in, 577–578  
4GL compared, 19  
procedure, 20–21

### 4GL (fourth-generation language), 19–20, 22

### 0 (zero) division error, 435–437, 463–464, 502, 518

## A

### ABS function

ANSI, 70, 72–73  
IBM DB2, 46  
Microsoft SQL Server, 46  
MySQL, 46, 320, 322–323  
Oracle, 46, 123–124  
PostgreSQL, 46, 388, 389  
standard identifier, 42  
Sybase, 46, 276, 277–278

### abstract data type (ADT), 515

### ACOS function

ANSI, 70, 73  
MySQL, 320, 323, 326  
PostgreSQL, 388, 390  
Sybase, 276, 278

### ActiveX error, 471

### Adaptive Server Enterprise. See ASE

### ADD\_MONTHS function (Oracle), 114, 115–116

### ADT (abstract data type), 515

### AGE function (PostgreSQL), 399–400

### aggregate function, 33–34, 427, 430, 585–586

### ALL\_MVIEW system tables, 636

### ALL PROCEDURES view, 444

### ALL\_SOURCE view, 444

## ALTER FUNCTION statement

---

### ALTER FUNCTION **statement**

- ANSI, 419
- IBM DB2, 459
- Microsoft SQL Server, 495
- MySQL, 525
- Oracle, 432
- Sybase, 516–517

### ALTER SESSION **statement, 447**

### ALTER SYSTEM **statement, 447**

### ALTER TABLE **statement, 6**

### **AL16UTF16 character set, 114**

### **analytic function, 29, 36, 97**

### **analytical database model, 629**

### AND

- bitwise, 324
- statement, 138

### **ANSI (American National Standards Institute). See also specific function and statement**

- data type, 5, 695
- IBM DB2 compliance, 16, 46–47
- INCITS, 15
- levels of compliance, 716
- Microsoft SQL Server compliance, 46–47
- MySQL compliance, 16, 46–47
- Oracle compliance, 3, 46–47
- PostgreSQL compliance, 16, 46–47
- procedural extension, 51–57
- query syntax, 60
- SQLJ standard, 54
- Sybase compliance, 16, 46–47
- UDB compliance, 16

### **API (Application Programming Interface)**

- JDBC, 603–604
- Oracle, 604

### **application**

- data heavy, 605
- database connection, establishing from, 602
- function, calling from, 601, 607
- login application
  - ASP.NET, creating using, 613–615
  - Java, creating using, 611–613
  - VB.Net, creating using, 608–611
- multitier, 603, 605
- permission, acquiring from, 602
- process model, 604–605, 606
- session, returning application running current, 219
- transaction heavy, 605

### **Application Programming Interface. See API**

### **arccosine, returning, 73, 323, 390**

### **archiving older data, optimizing performance via, 548**

### **arcsine, returning, 73–74, 323, 390**

### **arctangent, returning, 74, 323–324, 390–391**

### **AREA function (PostgreSQL), 404–405**

### **array**

- defined, 713
- UDF array static value support, declaring, 418

### **The Art of Computer Programming, Volume 3: Sorting and Searching (Knuth), 104**

### **ASCII character set**

- character corresponding to code, returning, 66, 99–100, 339, 379
- code of character, returning, 66–67, 187, 338, 377–378
- US7ASCII, 109

### **ASCII function**

- ANSI, 66–67
- Microsoft SQL Server, 187
- MySQL, 334, 338
- PostgreSQL, 375, 377–378

### **ASE (Adaptive Server Enterprise). See also specific Sybase function**

- background, historical, 2, 3
- boot time/date, returning, 301
- busy time, returning, 303, 305
- character set information, returning, 301–302, 306, 307, 313
- client information, returning, 301–302
- connections, returning maximum number of simultaneous, 306–307
- error
  - log, 303–304
  - number, returning, 303, 310–311
- IDENTITY column value, returning, 304
- idle time, returning, 305
- language information, returning, 305–306
- login attempts, returning number of, 302–303
- OmniConnect feature, 510
- SQLJ licensing, 509
- version, returning, 313–314

### **ASIN function**

- ANSI, 70, 73–74
- MySQL, 320, 323
- PostgreSQL, 388, 390
- Sybase, 276, 278

### **ASP.NET, creating login application using, 613–615**

### **asterisk (\*)**

- password, replacing with asterisk display in view, 646
- text, replacing trimmed with, 145, 191
- wildcard character, 60, 69, 103, 532

### **at sign (@)**

- procedure argument prefix, 298
- UDF parameter prefix, 479
- variable prefix, 298

**ATAN function**

ANSI, 70, 74  
 MySQL, 320, 323  
 PostgreSQL, 388, 390–391  
 Sybase, 276, 278–279

**ATAN2 function**

ANSI, 70, 74  
 MySQL, 320, 323–324  
 PostgreSQL, 388, 391

**ATN2 function (Sybase), 276, 279****average, returning**

ANSI, 62–63  
 MySQL, 318  
 optimizing, 632  
 Oracle, 91–92  
 PostgreSQL, 372  
 Sybase, 273  
 view, using, 620–621

**AVG function**

ANSI, 61, 62–63  
 IBM DB2, 46  
 Microsoft SQL Server, 46  
 MySQL, 46, 317, 318  
 Oracle, 46, 90, 91–92  
 PostgreSQL, 46, 371, 372  
 SELECT clause, 632  
 standard identifier, 41  
 Sybase, 46, 273

**B****backslash (\)**

escape character, 114, 532  
 UNISTR function character set prefix, 114

**bars (||) concatenation operator, 68****BEGIN TRAN statement, 312****BEGIN TRANSACTION statement, 503****BEGIN...END statement, 522****BENCHMARK function (MySQL), 365, 366****BI (Business Intelligence), 713****big-endian data format, 270****BIN function (MySQL), 334, 339****binary value, converting to/from string, 339, 380****bit status, determining, 124****BIT\_AND function (MySQL), 320, 324****BITAND function (Oracle), 123, 124****BIT\_COUNT function (MySQL), 320, 324–325****BIT\_LENGTH function**

ANSI, 41, 46, 68, 135  
 PostgreSQL, 376, 378

**BIT\_OR function (MySQL), 320, 325****BLOB data type, 713****Boolean data type, 713****@@BOOTTIME function (Sybase), 299, 301****box**

height, returning, 406  
 intersection of two boxes, returning, 405  
 width, returning, 408

**BOX\_INTERSECT function (PostgreSQL), 404, 405****brackets, square ( [ ] )**

PostgreSQL array delimiters, 532  
 validation string delimiters, 192  
 wildcard text delimiters, 253

**branch point, 13****BTRIM function (PostgreSQL), 375, 378****built-in function**

described, 19  
 dropping, 461  
 executing, 17  
 IBM DB2, 36–37  
 Microsoft SQL Server, 37–38  
 MySQL, 39  
 Oracle, 35–36  
 PostgreSQL, 39–40  
 SQL Server, 37–38  
 Sybase, 37–38, 245  
 UDF, overriding with, 458–459  
 UDF, performance compared, 633  
 UDF, system, 492–493  
 UDF, using in, 481

**Business Intelligence (BI), 713****C****C data type, 579****CALL statement, 52****calling function**

application, from, 601, 607  
 UDF, 445–447, 483, 523, 543

**caret (^)**

wildcard character, 253  
 XOR operator, 584

**CASE**

function (Microsoft SQL Server), 216, 219–220  
 statement, 52, 129

**CAST function**

IBM DB2, 46  
 Microsoft SQL Server, 46, 217, 220–224  
 Oracle, 46, 106, 107–108  
 PostgreSQL, 46  
 standard identifier, 42  
 Sybase, 46

**CBRT function (PostgreSQL), 388, 391**

## CEIL function

---

### CEIL function

ANSI, 70, 74–75  
MySQL, 320, 325  
Oracle, 123, 124  
PostgreSQL, 388, 391–392

### CEILING function

ANSI, 70, 74–75  
IBM DB2, 47, 70  
Microsoft SQL Server, 47, 70  
MySQL, 47, 320, 325  
Oracle, 47  
standard identifier, 42  
Sybase, 47, 276, 279–280

### CENTER function (PostgreSQL), 404, 405

### CHAR function

ANSI, 66  
Microsoft SQL Server, 187–188  
MySQL, 334, 339

### character function, 35, 97–99

#### character set. *See also text; specific character set*

ASE character set information, returning, 301–302, 306, 307, 313  
client character set information, returning, 301–302  
data type  
  converting between sets, 108–109, 113–114, 220, 266–270, 379  
  Unicode representation, returning, 108  
national, 99–100, 107, 113–114  
server character set information, returning, 301–302, 306, 307, 313

### CHARACTER\_LENGTH function

IBM DB2, 46  
Microsoft SQL Server, 46  
MySQL, 335, 344–345  
Oracle, 46  
PostgreSQL, 46  
standard identifier, 41  
Sybase, 46

### CHARINDEX function

Microsoft SQL Server, 185, 188  
Sybase, 248, 249–250

### CHAR\_LENGTH function

ANSI, 68  
MySQL, 335, 344–345  
PostgreSQL, 376, 379  
Sybase, 190, 248, 250

### CHR function

ANSI, 66, 67  
Oracle, 97, 99–100  
PostgreSQL, 376, 379

### CI.exe compiler, 577

### classification, function, 27

### clause, 7

### @@CLIENT\_CSID function (Sybase), 299, 301–302

### @@CLIENT\_CSNAME function (Sybase), 299, 302

### CLOB data type, 714

### CLOSE statement, 585

### COALESCE function

ANSI, 85  
IBM DB2, 177, 178  
Microsoft SQL Server, 217, 224–225  
MySQL, 365, 366  
Oracle, 128–129, 133  
PostgreSQL, 409–410

### COBOL data type, 579

### CODASYL (Committee on Data Systems and Languages), 15, 714

### code block, 714

### code generation optimization (Oracle), 714

### COLLATIONPROPERTY function (Microsoft SQL Server), 218, 233

### COL\_LENGTH function

Microsoft SQL Server, 203, 204–205  
Sybase, 288, 289–290

### COL\_NAME function (Sybase), 288, 290

### colon (:) array slice indicator, 532

### column function, 35, 37, 621

### comma (,)

list element separator, 532  
string set, comma-delimited, 341, 342, 346

### COMMIT statement, 447, 628

### COMMIT TRANSACTION statement, 503

### Committee on Data Systems and Languages (CODASYL), 15, 714

### COMPARE function (Sybase), 248, 250–252

### compilation. *See also specific compiler*

back end, 713  
ESQL, 577, 578  
function, embedded, 577, 578  
in-line, 56  
make file, 441–442  
module, compiled, 714  
3GL, 20–21  
UDF  
  interpreted mode, 432, 441, 443  
  native, 422, 432, 441, 443  
  Oracle, 421–422, 431–432, 440–443  
  smart, 422

### COMPOSE function (Oracle), 106, 108

### COMPRESS function (MySQL), 334, 339–340

- CONCAT function**  
 ANSI, 66, 67–68  
 IBM DB2, 140, 142, 472–473  
 MySQL, 334, 340  
 overloading, 472–473
- concatenation**  
 data warehouse data, 574–575  
 string  
   ANSI, 67–68  
   IBM DB2, 142  
   MySQL, 340–341  
   Sybase, 266
- CONCAT\_WS function (MySQL), 334, 340–341**
- configuration function, 38**
- CONNECT statement, 4, 21**
- connection. See database, connection; server, connection**
- @@CONNECTION function (Microsoft SQL Server), 207–208**
- CONNECTION\_ID function (MySQL), 365, 366**
- @@CONNECTIONS function**  
 Microsoft SQL Server, 481  
 Sybase, 299, 302–303
- control structure, 714. See also 3GL**
- CONV function (MySQL), 320, 325–326**
- conversion function, 36, 38, 40, 106–107**
- CONVERT function**  
 Microsoft SQL Server, 217, 220  
 Oracle, 106, 108–109  
 PostgreSQL, 179, 376, 379  
 Sybase, 266–270
- Coordinated Universal Time (UTC), 117, 176, 202, 481**
- CORR function**  
 Oracle, 90, 92–93  
 standard identifier, 43
- COS function**  
 ANSI, 70, 75  
 MySQL, 320  
 PostgreSQL, 388, 392  
 Sybase, 276, 280
- COSH function (ANSI), 70, 75**
- cosine, returning, 75, 280, 326, 392**
- COT function**  
 ANSI, 70, 75–76  
 MySQL, 320, 326  
 PostgreSQL, 388, 392  
 Sybase, 276, 280
- cotangent, returning, 75–76, 280, 326, 392**
- COUNT function**  
 ANSI, 61, 63–64  
 IBM DB2, 46  
 Microsoft SQL Server, 46  
 MySQL, 317, 318  
 Oracle, 46, 90, 93–94  
 PostgreSQL, 46, 371, 373  
 standard identifier, 41  
 Sybase, 46, 273, 274  
 UDF, overriding with, 458–459
- COVAR\_POP function, 43**
- COVAR\_SAMP function, 43**
- @@CPU\_BUSY function**  
 Microsoft SQL Server, 234, 235, 481  
 Sybase, 299, 303
- CREATE DATABASE statement, 5**
- CREATE FUNCTION statement**  
 AGGREGATE USING clause, 427  
 AUTHID clause, 426  
 BEGIN ATOMIC clause, 454, 476  
 CALL ON NULL INPUT clause, 418  
 CALLED ON NULL INPUT clause, 453  
 CLUSTER BY clause, 427  
 COMMENT clause, 522  
 CONTAINS SQL clause, 418, 453  
 DETERMINISTIC clause, 418, 426, 522  
 EXTERNAL ACTION clause, 453  
 FEDERATED clause, 453  
 IN clause, 424, 452, 522  
 IN OUT clause, 424, 452, 522  
 INHERITS SPECIAL REGISTERS clause, 453  
 LANGUAGE clause  
   ANSI, 418  
   IBM DB2, 452, 468, 469  
   MySQL, 522, 528  
 MAPPING clause, 458  
 MODIFIES SQL DATA clause, 418  
 NO EXTERNAL ACTION clause, 453  
 NO SQL clause, 418  
 NOCOPY clause, 425  
 NON FEDERATED clause, 453  
 NONDETERMINISTIC clause, 418, 522  
 ORDER BY clause, 427  
 OUT clause, 424, 452, 522  
 PARALLEL ENABLE clause, 427  
 PARAMETER STYLE clause, 418  
 PIPELINED clause, 427  
 PRAGMA AUTONOMOUS\_TRANSACTION clause, 427  
 PRAGMA EXCEPTION\_INIT clause, 435  
 PRAGMA RESTRICT\_REFERENCES clause, 446, 447  
 PRAGMA SERIALLY\_REUSABLE clause, 438  
 PREDICATES clause, 453  
 READS SQL clause, 453  
 READS SQL DATA clause, 418, 469, 476

## CREATE FUNCTION statement (continued)

---

### CREATE FUNCTION **statement (continued)**

REPLACE clause, 424  
RETURN NULL ON NULL INPUT clause, 418, 514  
RETURNS clause, 457, 524, 525  
SELECTIVITY clause, 453  
SOURCE clause, 457  
SPECIFIC clause, 418, 452, 460, 471  
SQL SECURITY clause, 522  
SQLJ function syntax, 510–513  
STATIC DISPATCH clause, 418, 453  
WITH clause, 480, 487, 488

CREATE INDEX **statement, 6–7, 29**

CREATE LIBRARY **statement, 427**

CREATE OR REPLACE FUNCTION **statement, 15**

CREATE TABLE **statement, 5–6**

**CRTSSQLCI (Create Structure Query Language ILE C) compiler, 449**

**cube root, returning, 391**

**CUME\_DIST function, 34, 45**

**CURDATE function (MySQL), 354, 356**

**CURRENT\_TIME function (PostgreSQL), 117**

**CURRENT DATE function**  
IBM DB2, 169, 170–171  
PostgreSQL, 121

**CURRENT DEFAULT TRANSFORM GROUP function (IBM DB2), 169, 171**

**CURRENT DEGREE function (IBM DB2), 169, 171**

**CURRENT EXPLAIN MODE function (IBM DB2), 169, 171–172**

**CURRENT EXPLAIN SNAPSHOT function (IBM DB2), 169, 172**

**CURRENT ISOLATION function (IBM DB2), 172–173**

**CURRENT NODE function (IBM DB2), 170, 173**

**CURRENT PATH function (IBM DB2), 170**

**CURRENT QUERY OPTIMIZATION function (IBM DB2), 170, 174**

**CURRENT REFRESH AGE function (IBM DB2), 170, 174**

**CURRENT SCHEMA function (IBM DB2), 170, 174**

**CURRENT SERVER function (IBM DB2), 170, 175**

**CURRENT TIME function (IBM DB2), 170, 175**

**CURRENT TIMESTAMP function (IBM DB2), 175–176**

**CURRENT TIMEZONE function (IBM DB2), 176**

**CURRENT\_DATABASE function (PostgreSQL), 409, 410**

**CURRENT\_DATE function**  
IBM DB2, 46  
Microsoft SQL Server, 46  
MySQL, 117  
Oracle, 46, 117, 120, 121  
PostgreSQL, 46, 399, 400  
standard identifier, 42  
Sybase, 46

**CURRENT\_DEFAULT\_TRANSFORM\_GROUP function (IBM DB2), 660**

**CURRENT\_SCHEMA function (PostgreSQL), 409, 410–411**

**CURRENT\_SCHEMAS function (PostgreSQL), 409**

**CURRENT\_TIME function**  
IBM DB2, 46  
Microsoft SQL Server, 46  
Oracle, 46  
PostgreSQL, 46, 399, 400  
standard identifier, 42  
Sybase, 46

**CURRENT\_TIMESTAMP function**  
IBM DB2, 46, 170  
Microsoft SQL Server, 46, 217, 225, 243  
Oracle, 46  
PostgreSQL, 46  
standard identifier, 42  
Sybase, 46, 295

**CURRENT\_TIMEZONE function (IBM DB2), 170**

**CURRENT\_USER function**  
Microsoft SQL Server, 217, 225–226  
PostgreSQL, 409, 411

**cursor**  
defined, 714  
function, 38, 585–586, 606  
migrating, 557  
status, returning, 309–310

**CURTIME function (MySQL), 354, 356–357**

## D

**Data Definition Language (DDL), 15**

**Data Dictionary, 443–445**

**data heavy application, 605**

**Data Manipulation Language (DML), 445, 533**

**Data Query Language (DQL), 533**

**data type**

ADT, 515  
ANSI, 5, 695  
BLOB, 713  
Boolean, 713  
C, 579  
CLOB, 714  
COBOL, 579  
column type, assigning, 6  
composite, 534–536  
converting  
binary value to/from string, 339, 380  
character data, to, 108, 110–113, 165–166, 194–195  
character sets, between, 108–109, 113–114, 220, 266–270, 379

- compatible types, between, 107–108, 220–224, 264–270
- decimal representation, to, 163–164
- disassembling result, 110
- explicit, 107
- floating-point representation, to, 164–165
- function, conversion, 36, 38, 40, 106–107
- hexadecimal representation, to, 164, 240, 270–271, 342
- IBM DB2, 162–166
- implicit, 107, 138, 142
- integer representation, to, 165, 271
- Microsoft SQL Server, 220–224
- Oracle, 106–114
- Sybase, 257–258, 263, 264–271
- Unicode representation, returning, 108
- custom, 714
- DBCLOB, 715
- ESQL, 579
- expression data type code, returning, 128, 130–131
- IBM DB2
  - converting, 162–166
  - listing of IBM DB2 types, 696
- Java, 513, 514–516
- Microsoft SQL Server
  - converting, 220–224
  - listing of SQL Server types, 697
- migrating, 557, 560, 575
- MySQL, 522, 697–698
- Oracle
  - converting, 106–114
  - listing of Oracle types, 696
- PostgreSQL
  - listing of PostgreSQL types, 698
  - UDT, 534–536
- SQLJ, 513, 514–516
- Sybase
  - converting, 257–258, 263, 264–271
  - listing of SQL Sybase types, 697
- UDF
  - argument, 425, 510
  - UDT, using in, 476, 479, 534–536
- UDT
  - identifier, returning, 182
  - name, returning, 182
  - PostgreSQL, 534–536
  - UDF, using in, 476, 479, 534–536
- data warehouse**
  - accuracy, 569
  - concatenating data, 574–575
  - consistency, 566–567, 569
  - disassembling/reassembling data, 575–576
  - inserting data, 566–567
  - naming convention, 566–567
  - querying, 567–569
  - schema, 566
  - scrubbing data, 569–570
  - standardizing data in multiple database environment, 570–572
  - summarizing data, 572–574
  - table
    - design, 566–567, 571
    - filtering column input, 569
    - join, 568
- database. See also schema; table**
  - analytical processing database architecture, 629, 713
  - blocking, 630–631
  - connection
    - application, establishing from, 602
    - changing to another database, 4
    - closing, 4, 602, 607, 612
    - node connected to, returning, 173
    - number of open connections, returning, 208
    - opening, 4, 602, 607, 610
    - partition connected to, returning, 173
    - pooling, 602, 607, 631
  - creating, 5–7
  - driver, 603–604, 711
  - ID
    - returning database ID, 205, 291–292
    - returning ID of object in database, 292–293
  - index
    - covering, 7
    - creating, 6–7
    - dynamic SQL, 172
    - function-based, 29, 30, 31
    - migrating, 557
    - performance, optimizing using, 373
  - instance timestamp, returning, 225
  - locking, 630–631
  - migrating
    - cursor, 557
    - data type, 557, 560, 575
    - field, 560–561
    - generating SQL, implementing via, 591–594
    - index, 557
    - integrity of new database, verifying, 559, 563–564
    - Microsoft SQL Server, 560–563, 575, 592–593
    - permission, 557
    - planning, 557–558
    - platform, to another, 556, 591–594
    - schema, 556–557
    - UDF, using, 560–564
    - user account, 557
    - version, to another, 556

## database (continued)

---

### database (continued)

#### name

returning name of database, 205–206, 292, 367, 410

returning name of object in database, 293

standardizing data in multiple database environment, 570–572

time zone, returning, 116–117

transactional, 628

user information, returning, 213, 214–216, 296–297, 411

### DATABASE function (MySQL), 365, 367

### DATALENGTH function

Microsoft SQL Server, 135, 194, 217, 226

Sybase, 135, 288, 290, 291

### date

abbreviation recognition by Sybase, 260–261

current, returning

IBM DB2, 170–171

Microsoft SQL Server, 202, 481

MySQL, 356, 362

Oracle, 120–121

PostgreSQL, 400, 403

server time zone date, 42

session time zone date, 42

Sybase, 264

system date, 115, 120–121, 170–171, 202, 362

### day

adding to/subtracting from input argument, 261, 357

month, returning day of, 151, 360

name, returning, 151–152, 359

rounding to nearest, 115, 119–120

week, returning day of, 152

week, specifying first day of, 199

year, returning day of, 153, 360

year zero, returning number of days from, 154–155, 360, 364–365

difference between two dates, calculating, 115, 118, 199, 262, 363

ESQL date function, 588–589

field, returning date value from, 117–118, 401

### formatting

IBM DB2, 154

ISO format, 263

Microsoft SQL Server, 198, 200–201

MySQL, 356, 358–359

Oracle, 119–122

Sybase, 267

### month

adding to/subtracting from input argument, 115–116, 198, 261, 357, 362–363

day of, returning, 151, 360

difference between two dates, returning in months, 363

name, returning, 157, 362

returning month part of date value, 156–157, 200, 201–203, 362

### quarter

adding to/subtracting from input argument, 261, 357

returning quarter part of date value, 200

query, restricting by, 17, 18

### server

boot date, returning, 301

system date, returning, 115, 120–121, 170–171, 202, 362

time zone, returning date in current, 42

timestamp, returning from date/time value, 42, 158

truncating date value, 115, 121–122, 262

### week

adding to/subtracting from input argument, 357

day of, returning, 152

day of, specifying first, 199

returning week part of date value, 161, 200

year, returning week of, 161, 263

### year

adding to/subtracting from input argument, 261, 357–358

day of, returning, 153, 360

days from year zero, returning number of, 154–155, 360, 364–365

returning year part of date value, 161–162, 200, 203

week of, returning, 161, 263

### DATE function (IBM DB2), 148, 150–151

### DATEADD function

Microsoft SQL Server, 197, 198–199

Sybase, 260, 261

### DATE\_ADD function (MySQL), 354, 357–358

### DATEDIFF function

Microsoft SQL Server, 118, 159, 197, 199

Sybase, 118, 159, 260, 262

### DATE\_DIFF function (MySQL), 118

### @@DATEFIRST function (Microsoft SQL Server), 197, 199

### DATE\_FORMAT function (MySQL), 354, 358–359

### DATENAME function

Microsoft SQL Server, 197, 200

Sybase, 260, 262–263

### DATEPART function

Microsoft SQL Server, 161, 197, 200–201

Sybase, 161, 260, 263

### DATE\_PART function

Microsoft SQL Server, 152

PostgreSQL, 151, 399, 400–401

Sybase, 152

### DATE\_SUB function (MySQL), 354, 357–358

### DATE\_TRUNC function (PostgreSQL), 399, 401

- DAY **function**
  - IBM DB2, 148, 151
  - Microsoft SQL Server, 197, 201–202
- DAYNAME **function**
  - IBM DB2, 148, 151–152
  - MySQL, 354, 359
- DAYOFMONTH **function (MySQL), 202, 354, 360**
- DAYOFWEEK **function**
  - IBM DB2, 148
  - MySQL, 152
- DAYOFWEEK\_ISO **function (IBM DB2), 148, 152**
- DAYOFYEAR **function**
  - IBM DB2, 148, 153
  - MySQL, 354, 360
- DAYS **function (IBM DB2), 149, 153**
- DBA\_MVIEW **system tables, 637**
- DBA\_PROCEDURES **view, 444**
- DBA\_SOURCE **view, 444**
- DBA\_VIEWS **system table, 637**
- DBCLOB **data type, 715**
- DB\_ID **function**
  - Microsoft SQL Server, 203, 205
  - Sybase, 288, 291–292
- DBMS\_DEBUG **package (Oracle), 434**
- DBMS\_OUTPUT **package (Oracle), 433–434, 440**
- DBMSSTDIX.SQL **file, 423**
- DB\_NAME **function**
  - Microsoft SQL Server, 204, 205–206
  - Sybase, 288, 292
- DBTIMEZONE **function (Oracle), 114, 116–117**
- DB2 database platform. See specific function and topic**
- DB2 LUW (DB2 for Linux, UNIX, and Windows), 463**
- DB2 SQL Procedural Language for Linux, UNIX and Windows (Yip et al.), 56**
- DDL (Data Definition Language), 15**
- Debug Procedure dialog box (Query Analyzer), 498–499**
- DEC **function (IBM DB2), 162, 163–164**
- DECIMAL
  - data type, 5, 163–164
  - function (IBM DB2), 162, 163–164
- declarative language, 49–51**
- DECLARE **statement, 585**
- DECODE **function**
  - Oracle, 128, 129–130
  - PostgreSQL, 167, 376, 380
- DECOMPOSE **function (Oracle), 106, 110**
- DECRYPT\_BIN **function (IBM DB2), 167**
- DECRYPT\_CHAR **function (IBM DB2), 167, 168**
- DEGREES **function**
  - ANSI, 70, 76
  - MySQL, 320, 326–327
  - PostgreSQL, 388, 393
  - Sybase, 276, 280–281
- delete **ESQL statement, 578**
- DELETE **SQL statement, 10–11, 507**
- DENSE\_RANK **function, 34, 45**
- DENY **statement, 478**
- deterministic/non-deterministic function**
  - ANSI, 418
  - IBM DB2, 29–30, 452
  - introduced, 27–28
  - Microsoft SQL Server, 30–31, 481, 507
  - MySQL, 522
  - Oracle, 29, 426
  - SQLJ, 512
  - Sybase, 31, 510
- DIAMETER **function (PostgreSQL), 404, 405–406**
- DIANA (Descriptive Intermediate Attributed Notation for Ada), 421**
- DICT\_COLUMNS **view, 636, 637**
- dictionary**
  - Data Dictionary, 443–445
  - string comparison, used in, 251
- Dictionary view, 443–445, 636**
- DIFFERENCE **function**
  - Microsoft SQL Server, 186, 188–189
  - Sybase, 248, 252
- DIGITS **function (IBM DB2), 177**
- direct SQL implementation, 715**
- dirty read, 628**
- DISCONNECT **statement, 4**
- DIVIDEBYZERO **function, 518**
- DML (Data Manipulation Language), 445, 533**
- dollar sign (\$) function positional argument indicator, 532**
- DOUBLE **function (IBM DB2), 162, 164–165**
- DOUBLE PRECISION **function (IBM DB2), 162, 164–165**
- DQL (Data Query Language), 533**
- DROP FUNCTION **statement**
  - ANSI, 419
  - Microsoft SQL Server, 496–497
  - MySQL, 526
  - Oracle, 432
  - PostgreSQL, 537–538
  - Sybase, 417
- DROP **statement, 459–461**
- DUMP **function (Oracle), 128, 130–131**
- dynamic SQL**
  - Explain information, enabling/disabling, 172
  - index, 172
  - intra-partition parallelism degree, returning, 171
  - overhead, 606
  - path of dynamic statement, resolving, 173
  - query, 172, 579–581, 606

### dynamic SQL (continued)

- schema, returning current, 174
- transform group used, returning, 171
- UDF, executing from, 507

## E

**Edit** ⇨ **Insert Template (Query Analyzer), 489**

**Edit** ⇨ **Replace Template Parameters (Query Analyzer), 489**

**ELT function (MySQL), 334, 341**

**Embedded SQL. See ESQL**

**encapsulation, 24, 715**

**ENCODE function (PostgreSQL), 376, 380**

**ENCRYPT function**

- IBM DB2, 167, 168
- Microsoft SQL Server, 239, 240

### encryption

- decrypting encrypted data, 167, 168, 240
- ESQL, defining encryption key using, 584–585
- login application, Java-based, 611
- MD5 hash, 382
- password, 168, 169, 241–242
- string, 168, 240, 382
- UDF, 480, 488

### E/R schema, 566

**error. See also specific error**

- ActiveX, 471
- ESQL, 581
- I/O errors, returning number of, 236–237, 310–311
- log, 303–304, 503, 612
- login application, Java-based, 612, 613
- message
  - custom, displaying, 464, 508
  - IBM DB2, 463, 464–471
- number, returning, 303, 310–311
- OLE, 471
- UDF error handling
  - IBM DB2, 463–464
  - Microsoft SQL Server, 502–503
  - MySQL, 526
  - Oracle, 434–437
  - PostgreSQL, 538
  - SQLJ, 517–518
  - Sybase, 517–518
- 0 (zero) division, 435–437, 463–464, 502, 518

**@@ERROR function**

- Microsoft SQL Server, 217, 226–227
- Sybase, 299, 303, 517, 518

**@@ERRORLOG function (Sybase), 299, 303–304**

**ESQL (Embedded SQL)**

- aggregate function, 585–586
- compiling, 577, 578

- data type, 579
- date function, 588–589
- encryption key, defining using, 584–585
- error handling, 581
- host variable usage, 579
- IBM DB2, 577
- language support, 577, 578
- math function, 586–587
- Microsoft SQL Server, 577
- MySQL, 577
- Oracle, 577
- PostgreSQL, 577
- query, dynamic, 579–581
- scalar function, 584–585
- statement, 578–579
- string function, 589–590
- Sybase, 577
- table
  - single-row function, 582–583
  - updating, 580–581
- 3GL, using in, 577, 578
- time function, 588–589
- UDF, embedded, 578

**EXCEPTION error handler, 463**

**EXEC statement, 578**

**EXECUTE IMMEDIATE statement, 476**

**EXECUTE statement, 476, 580**

**execution speed of expression, timing, 366**

**EXP function**

- ANSI, 70, 76–77
- IBM DB2, 47
- Microsoft SQL Server, 47
- MySQL, 47, 320, 327
- Oracle, 47
- PostgreSQL, 47, 388, 393
- standard identifier, 42
- Sybase, 47, 276, 281

**Explain facility, 171–172**

**expression**

- comparing expressions, 85–86, 128, 129–133
- data type code, returning, 128, 130–131
- execution speed, timing, 366
- list, returning first non-null value in
  - ANSI, 85
  - IBM DB2, 178
  - Microsoft SQL Server, 224–225
  - MySQL, 366
  - Oracle, 128–129
  - PostgreSQL, 409–410
  - Sybase, 85
- list, returning greatest value in, 128, 131–132

null, checking if  
 ANSI, 85–86  
 IBM DB2, 180  
 Microsoft SQL Server, 229–230  
 MySQL, 343–344, 347  
 Oracle, 132–133  
 PostgreSQL, 411  
 regular expression, 106, 188, 250, 386  
 size in bytes, returning, 130–131, 134–135, 226, 291

**eXtensible Markup Language (XML), returning query result as, 184**

**extension, procedural, 38, 39, 50, 51–57**

**EXTRACT function**

Oracle, 114, 117–118, 203  
 PostgreSQL, 399, 401  
 Sybase, 203

## F

**F7DEC character set, 109**

**FETCH statement, 585**

**field**

migrating, 560–561  
 returning date/time value from, 117–118, 401  
 string index value, returning, 341–342

**FIELD function (MySQL), 334, 341–342**

**file**

compilation make file, 441–442  
 ID, returning, 206  
 name, returning, 207  
 query result, sending to, 316  
 string, returning as, 367

**FILE\_ID function (Microsoft SQL Server), 204, 206**

**FILE\_NAME function (Microsoft SQL Server), 204, 207**

**FIND\_IN\_SET function (MySQL), 335, 342**

**FIRST\_VALUE function (Oracle), 28**

**FLOOR function**

ANSI, 70, 74–75  
 IBM DB2, 47, 70  
 Microsoft SQL Server, 47, 70  
 MySQL, 47, 321, 327  
 Oracle, 47, 123, 124–125  
 PostgreSQL, 47, 388, 393  
 standard identifier, 42  
 Sybase, 47, 277, 281

**fn\_get\_sql function, 239, 240–241, 492**

**fn\_helpcollations function, 492**

**fn\_listextendedproperty function, 492, 503**

**fn\_serversharedrive function, 493**

**fn\_trace\_getinfo function, 493**

**fn\_virtualfilestats function, 235, 238, 493**

**fn\_virtualservernodes function, 493**

**FOR statement, 52**

**FORMAT function (MySQL), 321, 327**

**4GL (fourth-generation language), 19–20, 22**

**FROM\_DAYS function (MySQL), 354, 360**

**FROM\_TZ function, 119**

**FROM\_UNIXTIME function (MySQL), 354, 361**

## G

**GENERATE\_UNIQUE function (IBM DB2), 28, 177**

**generating SQL**

database migration, implementing via, 591–594  
 random value, populating column with using generated  
 SQL, 594–598  
 UDF, using, 592–593, 596–598

**GETDATE function**

Microsoft SQL Server, 27–28, 197, 198, 202, 481  
 Sybase, 117, 121, 260, 264

**GETHINT function (IBM DB2), 167, 169**

**GETUTCDATE function**

Microsoft SQL Server, 117, 197, 202, 481  
 Sybase, 117

**GRANT statement, 477–478**

**GREATEST function**

MySQL, 321, 328  
 Oracle, 128, 131–132

**GROUPING function**

ANSI, 64  
 Oracle, 90, 94

**GV\$FIXED\_VIEW\_DEFINITION system table, 637**

## H

**HAS\_DBACCESS function (Microsoft SQL Server), 213**

**HEIGHT function (PostgreSQL), 404, 406**

**HEX function**

IBM DB2, 162, 164  
 MySQL, 335, 342

**hexadecimal representation of value, returning, 164, 240, 270–271, 342**

**HEXTOINT function (Sybase), 266, 271**

**hint, query, 88**

**HOST\_ID function (Microsoft SQL Server), 217, 227**

**HOST\_NAME function (Microsoft SQL Server), 217, 227–228**

**HOURL function**

IBM DB2, 149, 153–154  
 MySQL, 355, 361

**hyperbolic cosine, returning, 75**

**hyperbolic sine, returning, 72, 82**

**hyperbolic tangent, returning, 72, 83–84**

**IBM DB2 database platform.** See *specific function and topic*

**IDENT\_CURRENT function, 234**

**identifier validity, checking, 297–298**

**@@IDENTITY function**

Microsoft SQL Server, 217, 228

Sybase, 299, 304

**IDENTITY function (Microsoft SQL Server), 217, 228–229**

**@@IDLE function**

Microsoft SQL Server, 234, 235, 481

Sybase, 299, 305

**IF statement, 51**

**IFNULL function, 133**

**image pointer, returning, 287–288**

**INCITS (International Committee for Information Technical Standards), 15**

**index, database.** See *database, index*

**INFORMATION\_SCHEMA view**

IBM DB2, 474

Microsoft SQL Server, 494, 504, 505–507

MySQL, 526

Oracle, 443

PostgreSQL, 533, 539, 542

Sybase, 518

**INITCAP function**

Oracle, 97, 100

PostgreSQL, 376, 380–381

**input/output.** See *I/O*

**INSERT**

function

IBM DB2, 140, 142–143

MySQL, 335, 342–343

statement, 9, 507, 591–594

**INSTALL JAVA statement, 516**

**INSTR function (MySQL), 335, 343**

**INT function (IBM DB2), 162, 165**

**International Committee for Information Technical Standards (INCITS), 15**

**International Standards Organization.** See *ISO*

**INTERVAL function (MySQL), 321, 328**

**intra-partition parallelism, 171**

**INTTOHEX function (Sybase), 266, 270–271**

**I/O (input/output)**

busy time, returning, 305

error total, returning, 236–237, 310–311

packet information, returning, 481

performance, optimizing, 631

read total, returning, 237, 311, 481

write total, returning, 237–238, 311, 481

**@@IO\_BUSY function**

Microsoft SQL Server, 234, 236, 481

Sybase, 299, 305

**IQ software, 3**

**ISCLOSED function (PostgreSQL), 404, 406**

**ISDATE function (Microsoft SQL Server), 218, 229**

**ISFINITE function (PostgreSQL), 399, 402**

**ISNULL function**

Microsoft SQL Server, 133, 218, 229–230

MySQL, 335, 343–344

Sybase, 133

**ISNUMERIC function (Microsoft SQL Server), 218, 230**

**ISO (International Standards Organization)**

date format, 263

SQL standard, 2

SQLJ standard, 54

time format, 154, 160

**isolation level, returning, 172–173**

**ISOPEN function (PostgreSQL), 404, 406–407**

**IS\_SEC\_SERVICE\_ON function (Sybase), 271, 272**

## J

**Java.** See also *SQLJ*

data type, 513, 514–516

database connection, opening using, 607

login application, creating using, 611–613

query, Java-based, 607–608

stub, 718

**Java Virtual Machine (JVM), 54**

**JDBC (Java Database Connectivity)**

API, 603–604

data type mapping, 514–516

driver, 603–604

native implementation, 603

procedure, calling, 711–712

SQLJ requirement, 54

**JOIN function (Microsoft SQL Server), 485**

**JULIAN\_DAY function (IBM DB2), 149, 154–155**

**JVM (Java Virtual Machine), 54**

## K

**keyword**

ANSI keyword list, 669

IBM DB2 keyword list, 683–685

introduced, 7

Microsoft SQL Server keyword list, 685–688

MySQL keyword list, 689

Oracle keyword list, 680–682

PostgreSQL keyword list, 691–693

Sybase keyword list, 690–691

**Knuth, Donald E. (*The Art of Computer Programming, Volume 3: Sorting and Searching*), 104**

**L****@@LANGID function**

Microsoft SQL Server, 207, 208

Sybase, 299, 305–306

**language. See also specific language**

ASE language information, returning, 305–306

declarative, 49–51

dictionary used in string comparison, 251

ESQL support, 577, 578

extension, 20

4GL, 19–20, 22

ID

current, returning, 208

description assigned to, returning, 306

procedural

ANSI procedural extension, 51–57

declarative language compared, 49–51

server language information, returning, 208–210,  
305–306

3GL, 19, 20–21, 577, 578

UDF language

declaring, 418, 511

returning from System Catalog, 641

**@@LANGUAGE function**

Microsoft SQL Server, 207, 208–210

Sybase, 300, 306

**Last In, First Out (LIFO), 13****lazy developer, 594****LCASE function**

IBM DB2, 190

MySQL, 335, 344

**LDL (Logic Database Language), 430****LEAST function (MySQL), 321, 328–329****least-squares-fit linear equation, 44****LEAVE statement, 52****LEFT function**

IBM DB2, 140, 143

Microsoft SQL Server, 186, 189

MySQL, 335, 344

Sybase, 248, 255

**LEN function**

ANSI, 66, 68–69

Microsoft SQL Server, 186, 189–190

**LENGTH function**

ANSI, 66, 68–69

IBM DB2, 135, 140, 143–144

MySQL, 335, 344–345

PostgreSQL, 376, 381, 404, 407

**LENGTHB function, 110, 134****LIFO (Last In, First Out), 13****Linker.exe file, 577****little-endian data format, 270****LN function**

ANSI, 71, 77

IBM DB2, 47

Microsoft SQL Server, 47

MySQL, 47

Oracle, 47

PostgreSQL, 47, 388, 394

standard identifier, 42

Sybase, 47

**load balancing, 311****LOAD\_FILE function (MySQL), 365, 367****LOCALTIME function**

IBM DB2, 46

Microsoft SQL Server, 46

Oracle, 46

PostgreSQL, 46, 117, 399, 402

standard identifier, 42

Sybase, 46

**LOCALTIMESTAMP function**

IBM DB2, 46

Microsoft SQL Server, 46

Oracle, 46

PostgreSQL, 46, 399, 402

standard identifier, 42

Sybase, 46

**LOCATE function**

IBM DB2, 141, 144

MySQL, 336, 345

**@@LOCK\_TIMEOUT function (Microsoft SQL Server), 207, 210****LOG function**

ANSI, 71, 77

MySQL, 321, 329

PostgreSQL, 388, 394

Sybase, 277, 281–282

**LOG10 function**

ANSI, 71, 77

MySQL, 321, 329

Sybase, 277, 282

**LOG2 function (ANSI), 71, 77****Logic Database Language (LDL), 430****login. See also server, connection**

ASP.NET, creating login application using, 613–615

Java, creating login application using, 611–613

name, returning, 214

number of logins attempted, returning, 208, 302–303, 481

SID of login name, returning, 214

VB.Net, creating login application using, 608–611

## LOOP statement

---

### LOOP **statement**, 52

#### LOWER **function**

ANSI, 66, 68  
IBM DB2, 46  
Microsoft SQL Server, 46, 190  
MySQL, 335, 344  
Oracle, 46  
PostgreSQL, 46, 376, 381  
standard identifier, 41  
Sybase, 46

#### LPAD **function**

MySQL, 336, 346  
Oracle, 98, 101–102  
PostgreSQL, 376, 381–382

#### LTRIM **function**

IBM DB2, 141, 144–145  
Microsoft SQL Server, 186, 190–191  
MySQL, 336, 346  
Oracle, 98, 102  
PostgreSQL, 376, 382  
Sybase, 248, 252–253

## M

### MAKE\_SET **function (MySQL)**, 336, 346–347

#### **math. See also number**

absolute value, returning  
ANSI, 72–73  
MySQL, 322–323  
Oracle, 123–124  
PostgreSQL, 389  
Sybase, 277–278  
arccosine, returning, 73, 323, 390  
arcsine, returning, 73–74, 323, 390  
arctangent, returning, 74, 323–324, 390–391  
area, returning, 404–405  
average, returning  
ANSI, 62–63  
MySQL, 318  
optimizing, 632  
Oracle, 91–92  
PostgreSQL, 372  
Sybase, 273  
view, using, 620–621  
bit status, determining using binary calculation, 124  
center point of object, returning, 405  
circle  
diameter, returning, 405–406  
radius, returning, 408  
cosine, returning, 75, 280, 326, 392  
cotangent, returning, 75–76, 280, 326, 392

cube root, returning, 391  
date difference, calculating, 115, 118, 199, 262  
degree/radian conversion  
ANSI, 76  
IBM DB2, 70, 71  
Microsoft SQL Server, 70, 71  
MySQL, 326–327, 331  
PostgreSQL, 393, 395–396  
Sybase, 280–281, 283  
exponentiation  
ANSI, 76–77, 79–80  
MySQL, 327, 330–331  
PostgreSQL, 393, 395  
Sybase, 281, 282–283  
factorial, returning, 428–429, 488–489, 523–524, 586–587  
function, mathematical  
ANSI, 69–72  
ESQL, 586–587  
PostgreSQL, 388–389  
Sybase, 275–277  
hyperbolic cosine, returning, 75  
hyperbolic sine, returning, 72, 82  
hyperbolic tangent, returning, 72, 83–84  
least-squares-fit linear equation, 44  
line length, returning, 407  
logarithm  
ANSI, 77  
MySQL, 329  
PostgreSQL, 394  
Sybase, 281–282  
modulo, 77–78, 125, 329–330, 394–395  
path, 406–408  
PI constant, 79, 282, 330, 395  
polygon  
box height, returning, 406  
box width, returning, 408  
boxes, returning intersection of two, 405  
corner points, returning number of, 407  
sign (positive/negative) of value, returning  
ANSI, 81–82  
MySQL, 332  
Oracle, 125–126  
PostgreSQL, 397  
Sybase, 285  
sine, returning, 285, 332, 397  
square root, returning  
ANSI, 83  
IBM DB2, 47, 457  
Microsoft SQL Server, 47  
MySQL, 332

- Oracle, 47
  - PostgreSQL, 398
  - Sybase, 286
  - UDF, using, 457, 458
  - standard deviation, returning, 43, 95–96, 332–333, 374
  - sum
    - ANSI, 65
    - MySQL, 319
    - Oracle, 96
    - PostgreSQL, 374–375
    - Sybase, 275
  - tangent, returning, 72, 83, 279, 286, 333
  - timestamp difference, calculating, 159
  - variance, returning, 43, 93, 375
  - 0 (zero) division error, 435–437, 463–464, 502, 518
  - MAX function**
    - ANSI, 61, 64
    - IBM DB2, 46
    - Microsoft SQL Server, 46
    - MySQL, 317, 319
    - Oracle, 46, 90, 95
    - PostgreSQL, 46, 371, 373
    - standard identifier, 42
    - Sybase, 46, 273, 274
    - syntax, 17
  - @MAXCHARLEN function (Sybase), 300, 306**
  - @MAX\_CONNECTIONS function**
    - Microsoft SQL Server, 207, 210, 481
    - Sybase, 300, 306–307
  - MD5 function (PostgreSQL), 376, 382**
  - MDX (Multidimensional Expressions) language, 40**
  - memory**
    - address, retrieving value from, 23–24
    - image pointer, returning, 287–288
    - stack, 13–14
    - text pointer, returning, 287–288, 481
    - UGA, 438
    - variable
      - pointer, 23
      - storage in/retrieval from memory, 22–23
  - metadata function, 38**
  - MICROSECOND function (IBM DB2), 149, 155**
  - Microsoft SQL Server. See specific function and topic**
  - @MICROSOFTVERSION function (Microsoft SQL Server), 239, 241, 314**
  - MIDNIGHT\_Seconds function (IBM DB2), 149, 155–156**
  - migrating data. See database, migrating**
  - MIN function**
    - ANSI, 61, 64
    - IBM DB2, 46
    - Microsoft SQL Server, 46
    - MySQL, 317, 319
    - Oracle, 46, 91, 95
    - PostgreSQL, 46, 371, 374
    - standard identifier, 42
    - Sybase, 46, 273, 275
  - MINUTE function**
    - IBM DB2, 149, 156
    - MySQL, 355, 361
  - MOD function**
    - ANSI, 71, 77–78
    - IBM DB2, 46, 71
    - Microsoft SQL Server, 46
    - MySQL, 46, 321, 329–330
    - Oracle, 46, 123, 125
    - PostgreSQL, 46, 389, 394–395
    - standard identifier, 42
    - Sybase, 46
  - module**
    - compiled, 714
    - defined, 716
    - PSM, 415
    - standard, 2
  - modulo, 77–78, 125, 329–330, 394–395**
  - MONTH function**
    - IBM DB2, 118, 149, 156–157
    - Microsoft SQL Server, 198, 202–203
    - MySQL, 203, 355, 362
  - MONTH\_BETWEEN function (Oracle), 115, 118**
  - MONTHNAME function**
    - IBM DB2, 149, 157
    - MySQL, 355, 362
  - Multidimensional Expressions (MDX) language, 40**
  - MySQL database platform. See specific function and topic**
  - mysql.func system table, 528**
  - MySQLGUI client, 4**
- ## N
- NCHAR function (Microsoft SQL Server), 114, 186, 191**
  - @NCHARSIZE function (Sybase), 300, 307**
  - NCHR function, 99–100**
  - @NESTLEVEL function**
    - Microsoft SQL Server, 207, 211
    - Sybase, 300, 307
  - Network Data Model, 15**
  - network function, 716**
  - networking software, proprietary, 604**
  - NEWID function (Microsoft SQL Server), 179, 218, 230, 481**
  - NEW\_TIME function (Oracle), 115, 118–119**
  - niladic function, 27**

**NO\_DATA\_FOUND error, 434**

**NOW function**

- MySQL, 355, 362
- PostgreSQL, 117, 121, 399, 403

**NPOINTS function (PostgreSQL), 404, 407**

**NULLIF function**

- ANSI, 85–86
- IBM DB2, 177, 180
- MySQL, 336, 347
- Oracle, 128, 132–133
- PostgreSQL, 409, 411

**number. See also math**

- base, converting, 325–326
- correlation coefficient for number pair, returning, 43, 92–93
- factorial, returning, 428–429, 488–489, 523–524, 586–587
- formatting, 112–113, 327, 630
- function, numeric
  - MySQL, 319–322
  - Oracle, 123
- integer representation, converting data type to, 165, 271
- integer, returning largest in set
  - ANSI, 74–75
  - IBM DB2, 70
  - Microsoft SQL Server, 70
  - MySQL, 327, 328
  - Oracle, 124–125
  - PostgreSQL, 373, 393
  - Sybase, 281
- integer, returning smallest in set
  - ANSI, 74–75
  - IBM DB2, 70
  - Microsoft SQL Server, 70
  - MySQL, 70
  - Oracle, 70
  - PostgreSQL, 374, 391–392
  - Sybase, 279–280
- rounding
  - ANSI, 80–81
  - MySQL, 331
  - Oracle, 119–120, 124–125, 126
  - PostgreSQL, 396–397
  - Sybase, 284
- sign (positive/negative), returning
  - ANSI, 81–82
  - MySQL, 332
  - Oracle, 125–126
  - PostgreSQL, 397
  - Sybase, 285

truncating

- ANSI, 84
- IBM DB2, 147–148
- MySQL, 333
- Oracle, 127
- PostgreSQL, 398
- value list, comparing to, 328
- value set, converting to predetermined, 341

**number sign (#)**

- temporary table prefix, 298
- UDF name prefix, 479

**NVL function (Oracle), 128, 133**

**NVL2 function (Oracle), 128, 133–134**

## O

**Object Linking and Embedding (OLE), 471, 490**

**object reference function, 36, 127**

**Object Viewer window (Query Analyzer), 498**

**OBJECT\_ID function (Sybase), 288, 292–293**

**OBJECT\_NAME function (Sybase), 288, 293**

**OCI (Oracle Call Interface), 604**

**OCT function (MySQL), 321, 330, 336, 347**

**OCTET\_LENGTH function**

- ANSI, 41, 46, 68
- PostgreSQL, 376, 383

**ODBC (Open Database Connectivity), 602–603, 686–688, 709–710**

**OLE (Object Linking and Embedding), 471, 490**

**OmniConnect feature (ASE), 510**

**OPEN statement, 585**

**@@OPTIONS function**

- Microsoft SQL Server, 207, 211
- Sybase, 300, 308

**OR**

- bitwise, 325
- statement, 138

**Oracle Call Interface (OCI), 604**

**Oracle database platform. See specific function and topic**

**ORD function (MySQL), 337, 348**

**OTHERS error category, 435**

**OVERLAY function (PostgreSQL), 376, 383**

**overloading function**

- IBM DB2, 26, 137–138, 471–473
- introduced, 25
- Microsoft SQL Server, 26
- MySQL, 26, 522
- Oracle, 25–26, 439–440
- package prerequisite, 440
- PostgreSQL, 538–539
- Sybase, 26

## P

### package, function

- Microsoft SQL Server, 25
- Oracle, 424, 428, 437–438, 446
- overloading package prerequisite, 440
- referencing, 438
- Sybase, 25
- UDF, 424, 428, 437–438, 446

**@@PACKET\_ERRORS function (Microsoft SQL Server), 481**

**@@PACK\_RECEIVED function (Microsoft SQL Server), 481**

**@@PACK\_SENT function (Microsoft SQL Server), 481**

### parallelism, intra-partition, 171

### parameter, passing to function, 22–24, 714

**PARAMETERS view, 504**

### parentheses ( ) expression group delimiters, 532

### partition connected to, returning, 173

### password

- encryption, 168, 169, 241–242
- login application password processing, 608–610, 611–612
- view, replacing with asterisk display in, 646

### path

- mathematical, 406–408
- statement, resolving path of dynamic, 173

### PATINDEX function

- Microsoft SQL Server, 191
- Sybase, 248, 253–254

### PCLOSE function (PostgreSQL), 404, 407–408

### percent sign (%)

- modulo operator, 71, 78
- wildcard character, 253

**PERCENTILE\_CONT function, 44**

**PERCENTILE\_DISC function, 45**

**PERCENT\_RANK function, 34, 45**

### performance

- archiving older data, via, 548
- average calculation, optimizing, 632
- code, improving via removing redundant, 629
- database connection pool, improving using, 602, 631
- diagnosing performance issue, 207
- dynamic SQL overhead, 606
- extension, improving using proprietary, 47
- function, optimizing, 629–633
- index, improving using, 373
- I/O performance, optimizing, 631
- JDBC API, improving using vendor-specific, 603, 604
- procedure, optimizing, 56, 629
- query, optimizing, 140, 174, 256, 627, 630–633
- security, balancing with, 633
- transaction processing, optimizing, 627–628
- UDF performance, optimizing, 425, 442, 453, 632

### period (.) UDT attribute indicator, 532

**PERIOD\_ADD function (MySQL), 355, 362–363**

**PERIOD\_DIFF function (MySQL), 355, 363**

### permission

- application, acquiring from, 602
- bitmap, returning, 231
- denying, 478
- granting, 423, 450, 477–478
- IBM DB2, 450–451, 703–705
- Microsoft SQL Server, 218, 231, 477–478, 706
- MySQL, 521, 707
- Oracle, 422–423, 426, 699–703
- PostgreSQL, 532–533, 707–708
- Privileges System Catalog, 636
- procedure, 423
- revoking, 451, 533
- SQLJ, 509
- Sybase, 509, 706–707
- view, 617

**PERMISSIONS function (Microsoft SQL Server), 218, 231**

### Persistent Stored Module (PSM), 415

**Pg\_indexes view, 645**

**Pg\_locks view, 645**

**PG\_PROC functions (PostgreSQL), 539–540**

**Pg\_rules view, 645**

**Pg\_settings view, 645**

**Pg\_shadow system table, 646**

**Pg\_stats view, 645**

**Pg\_tables view, 645, 646**

**Pg\_user view, 645, 646–647**

**Pg\_views view, 645, 647**

### phone number, formatting, 630

### PI function

- ANSI, 71, 79
- MySQL, 321, 330
- PostgreSQL, 389, 395
- Sybase, 277, 282

### planning

- database migration, 557–558
- function development, 605–606

### PL/SQL compiler, 421

**PL/SQL (Procedural Language/SQL), 20. See also specific Oracle function**

**PL/SQL Virtual Machine (PVM), 421–422**

### plus sign (+) concatenation operator, 68

### polygon

- box height, returning, 406
- box width, returning, 408
- boxes, returning intersection of two, 405
- corner points, returning number of, 407

**POPEN function (PostgreSQL), 404, 408**

## POSITION function

---

### POSITION function

PostgreSQL, 376, 383–384  
standard identifier, 41

### POSIX regular expression, 386

POSSTR function (IBM DB2), 141, 144

**PostgreSQL database platform. See specific function and topic**

### POW function

MySQL, 321, 330–331  
PostgreSQL, 389, 395

### POWER function

ANSI, 71, 79–80  
IBM DB2, 47  
Microsoft SQL Server, 47  
MySQL, 47, 321, 330–331  
Oracle, 47  
PostgreSQL, 47  
standard identifier, 42  
Sybase, 47, 277, 282–283

PREPARE statement, 476, 580

**privilege. See permission**

**probe user instance ID, returning, 308**

@@PROBESUID function (Sybase), 300, 308

**Pro\*C precompiler, 21**

PROC system table, 528

### procedural language

declarative language compared, 49–51  
extension, 38, 39, 50, 51–57

**Procedural Language/SQL (PL/SQL), 20. See also specific Oracle function**

**procedure. See also specific procedure**

argument prefix, 298  
context switch, 56  
debugging, 432–433  
described, 18  
extended stored procedure, 55, 490  
function compared, 18–19, 20, 416–417  
JDBC, calling from, 711–712  
nesting level, returning, 211, 307  
OLE Automation procedure, 490  
Oracle, 711–712  
performance, optimizing, 56, 629  
permission, 423  
stand-alone, 437  
3GL, 20–21  
UDF, using in, 475, 490–492, 507–508  
unfenced mode, 56

**process ID, returning, 212, 309**

**program stack. See memory, stack**

PROGRAM\_ERROR error, 434

**project creep, 546**

**PSM (Persistent Stored Module), 415**

**pull reporting implementation, 553**

**push reporting implementation, 553**

**PVM (PL/SQL Virtual Machine), 421–422**

PWDCOMPARE function (Microsoft SQL Server), 239, 241

PWDENCRYPT function (Microsoft SQL Server), 239, 242

## Q

### query

ad hoc, 549–553  
case sensitivity, 8  
conditional statements, linking, 138  
data warehouse, 567–569  
date, restricting by, 17, 18  
dynamic, 172, 579–581, 606  
file, sending result to, 316  
grouping result set  
  IBM DB2, 138–139  
  Microsoft SQL Server, 184, 185  
  Oracle, 89, 91, 93, 94  
  Sybase, 246  
hint, 88, 184–185  
Java-based, 607–608  
looping through result set, 607  
performance, optimizing, 140, 174, 256, 627, 630–633  
process handle, 240–241  
semidynamic, 580  
sorting result set  
  IBM DB2, 139  
  Microsoft SQL Server, 184, 185  
  Oracle, 89, 94  
  Sybase, 255–256  
  view, 620  
static, 579  
subquery, 718  
syntax  
  ANSI, 60  
  IBM DB2, 138–140  
  Microsoft SQL Server, 183–185  
  MySQL, 315–317  
  Oracle, 87–90  
  PostgreSQL, 369–371  
  Sybase, 246–247  
  UDB, 138–140  
System Catalog query  
  IBM DB2, 641–645  
  introduced, 635  
  Microsoft SQL Server, 648–652  
  MySQL, 655

Oracle, 636–640  
 PostgreSQL, 645–647  
 Sybase, 652–655  
 system object, 290  
 TOP N analysis, 8, 185  
 UDF query function, 533–534, 549–553, 568–569  
 view, filling using, 617–619, 620  
 XML, returning result set as, 184

**Query Analyzer software, 4, 489, 497–501**

**Quest Central software, 461**

**question mark (?) Java parameter placemaker, 612**

**quotation marks, double (“ ”) string delimiters, 192, 384**

**quotation marks, single (‘ ’)**  
 escape delimiters, 532  
 string delimiters, 192, 384

**QUOTE\_IDENT function (PostgreSQL), 377, 384**

**QUOTE\_LITERAL function (PostgreSQL), 377, 384**

**QUOTENAME function (Microsoft SQL Server), 186, 192**

## R

**RADIANS function**  
 ANSI, 71, 76  
 MySQL, 321, 331  
 PostgreSQL, 395–396  
 Sybase, 277, 283

**RADIUS function (PostgreSQL), 404, 408**

**RAISE\_APPLICATION\_ERROR function (Oracle), 436**

**RAISE\_ERROR function (IBM DB2), 464**

**RAISEERROR statement, 508**

**RAND function**  
 ANSI, 71, 79–80, 80  
 IBM DB2, 80, 177, 180–181  
 Microsoft SQL Server, 71, 80, 481  
 MySQL, 71, 80, 322, 331  
 Oracle, 71  
 PostgreSQL, 71  
 Sybase, 71, 277, 283–284, 288, 293–294

**RANDOM function (PostgreSQL), 389, 396**

**random value, returning**  
 ANSI, 79–80  
 IBM DB2, 180–181  
 Microsoft SQL Server, 80, 481  
 MySQL, 331  
 PostgreSQL, 396  
 Sybase, 283–284, 293–294

**RANK function, 34, 45**

**RDL (Relational Database Language), 15**

**read, dirty, 628**

**register, special, 121, 167, 168, 169, 453**

**REGR\_AVGX function, 44**

**REGR\_AVGY function, 44**

**REGR\_COUNT function, 43**

**REGR\_INTERCEPT function, 44**

**REGR\_R2 function, 44**

**REGR\_SLOPE function, 44**

**REGR\_SXX function, 44**

**REGR\_SXY function, 44**

**REGR\_SYY function, 44**

**regular expression, 106, 188, 250, 386**

**Relational Data Model, 15**

**Relational Database Language (RDL), 15**

**REMOVE JAVA statement, 517**

**REPEAT function**  
 IBM DB2, 102, 141, 145  
 MySQL, 337, 348  
 PostgreSQL, 377, 384–385

**REPLACE function**  
 ANSI, 66, 69  
 case sensitivity, 146  
 IBM DB2, 141, 145–146  
 Microsoft SQL Server, 186, 192  
 MySQL, 337, 348  
 Oracle, 98, 103  
 PostgreSQL, 377, 385  
 TRANSLATE function compared, 103, 165

**REPLICATE function**  
 Microsoft SQL Server, 102, 186, 192–193  
 Sybase, 102, 248, 254

**reporting, 545–549, 553**

**ResultSet class, 515**

**RETURN statement**  
 IBM DB2, 427, 454, 467  
 introduced, 52  
 Microsoft SQL Server, 484, 508

**reusability, 17**

**REVERSE function**  
 Microsoft SQL Server, 186, 193  
 MySQL, 337, 349  
 Sybase, 248, 254–255

**REVOKE statement, 451, 533**

**RIGHT function**  
 IBM DB2, 141, 143  
 Microsoft SQL Server, 186, 189  
 MySQL, 337, 349  
 Sybase, 248, 255

**ROLLBACK statement, 447, 628**

**ROUND function**  
 ANSI, 71, 80–81  
 MySQL, 331  
 Oracle, 115, 119–120, 123, 126  
 PostgreSQL, 389, 396  
 Sybase, 277, 284

## routine

- executable, 415
  - external, 417, 418
  - invocable, 415
  - UDF versus, 415
- ROUTINE\_COLUMNS** **view**, 504, 505
- ROUTINE\_PRIVILEGES** **view**, 542
- ROUTINES** **view**, 504, 540–542
- @@ROWCOUNT** **function**
- Microsoft SQL Server, 218, 231–232
  - Sybase, 300, 308–309
- ROWCOUNT\_BIG** **function (Microsoft SQL Server)**, 218, 231–232
- RPAD** **function**
- MySQL, 337, 349
  - Oracle, 98, 101–102
  - PostgreSQL, 377, 385
- RTRIM** **function**
- IBM DB2, 141, 144–145
  - Microsoft SQL Server, 186, 190–191
  - MySQL, 337, 350
  - Oracle, 98, 102
  - PostgreSQL, 377, 386
  - Sybase, 249, 252–253

## S

### **SADL (Simple Aggregate Definition Language)**, 430

### **sales tax calculation**, 15

### **scalar function**

- built-in, 33, 34–35, 36
- ESQL, 584–585
- UDF, 451, 452–456, 480, 482–484

### **schema**

- current, returning, 174, 410–411
- data warehouse, 566
- E/R, 566
- ID, returning, 642
- implicit, 450, 457
- migrating, 556–557
- name, returning, 642, 643
- snowflake, 566
- star, 566
- state schema, 718
- SYSFUN, 137, 449
- SYSIBM, 137, 140, 173, 449
- timestamp, returning, 642
- UDF binding, 480, 487–488

### **scope**

- function, 24–25
- IDENTITY column value, returning, 234

- report, 546
  - variable, 24–25, 298
- SCOPE\_IDENTITY** **function (Microsoft SQL Server)**, 218, 233–234
- scrubbing, data**, 569–570
- SECOND** **function**
- IBM DB2, 149, 157
  - MySQL, 355, 363
- SEC\_TO\_TIME** **function (MySQL)**, 355, 363–364
- security. See also specific security mechanism**
- function scope, role in, 25
  - isolation level, returning, 172–173
  - performance, balancing with, 633
  - service
    - listing all active, 272
    - status, checking, 272
  - variable scope, role in, 25
- security identification number (SID)**, 214
- SELECT** **statement**
- ALL clause, 138, 316
  - analytical database model, processing within, 629
  - COMPUTE clause, 184, 246
  - CONNECT BY clause, 88–89
  - CUBE clause, 138–139, 185
  - DISTINCT clause, 8, 138, 184, 246, 316
  - DISTINCTROW clause, 316
  - EXCEPT clause, 370
  - FETCH FIRST clause, 140
  - FOR clause, 184, 246
  - FOR UPDATE clause, 371
  - FROM clause
    - ANSI, 60
    - IBM DB2, 138, 454
    - introduced, 7
    - MySQL, 316
    - Oracle, 88
    - PostgreSQL, 370
  - FROM DUAL clause, 73
  - FULL OUTER clause, 29
  - GROUP BY clause
    - ANSI, 60
    - IBM DB2, 138
    - Microsoft SQL Server, 185
    - MySQL, 316
    - Oracle, 89
    - PostgreSQL, 370
  - GROUPING SETS clause, 139
  - HAVING clause
    - ANSI, 60
    - IBM DB2, 138, 139

- MySQL, 316–317
- Oracle, 89
- PostgreSQL, 370
- HIGH PRIORITY clause, 316
- INTERSECT clause, 370
- INTO clause, 316, 434
- JOIN clause, 568
- LIMIT clause, 317, 370, 371
- OFFSET clause, 370
- OPTIMIZE FOR clause, 140
- OPTION clause, 184
- ORDER BY clause
  - IBM DB2, 139–140
  - introduced, 8
  - Microsoft SQL Server, 184, 185
  - MySQL, 317
  - Oracle, 89
  - PostgreSQL, 370, 371, 533
  - Sybase, 246
- ROLLUP clause, 138–139, 185
- rows returned by, returning number of, 232
- SQL\_BIG\_RESULT clause, 316
- SQL\_CALC\_FOUND\_ROWS clause, 316
- START WITH clause, 88–89
- STRAIGHT\_JOIN clause, 316
- syntax, 7, 60, 87–88, 138–140
- tables, joining using, 316, 370, 568
- TOP clause, 8
- UNION clause, 185, 370
- WHERE clause
  - ANSI, 60
  - IBM DB2, 138
  - introduced, 8
  - MySQL, 316–317
  - Oracle, 89
  - PostgreSQL, 370
- semicolon (;) command suffix, 532**
- semidynamic statement, 580**
- server**
  - bitmask information about current configuration, returning, 211
  - character set information, returning, 301–302, 306, 307, 313
  - connection
    - login name, returning, 214
    - login SID, returning, 214
    - logins attempted, returning number of, 208, 302–303, 481
    - simultaneous connections, returning maximum number of, 210, 306–307, 481
    - thread connection ID, returning, 366
  - current, returning, 175
  - date
    - boot date, returning, 301
    - system date, returning, 115, 120–121, 170–171, 202, 362
    - time zone, returning date in current, 42
  - language information, returning, 208–210, 305–306
  - load balancing, 311
  - lock timeout, returning, 210
  - time
    - boot time, returning, 301
    - busy time, returning, 235–236, 303, 305, 481
    - date in server time zone, returning current, 42
    - idle time, returning, 235–236, 305, 481
    - system time, returning, 27–28, 115, 120–121, 202
    - tick, returning, 236, 310, 481
  - user
    - information about, returning, 294–295, 308
    - validating user ID, 298–299
- session**
  - application name of current, returning, 219
  - debugger/debuggee, 715
  - ID of current, returning, 212
  - identifier, returning, 128, 134
  - time zone
    - returning, 114, 116–117
    - returning current date/time in, 42
  - transactions pending, returning number of, 232–233
  - user, returning, 176–177, 225–226, 412
  - workstation ID, returning, 227
- SESSIONTIMEZONE function (Oracle), 114, 116–117**
- SESSION\_USER function**
  - IBM DB2, 176–177
  - PostgreSQL, 409, 412
- SETSEED function (PostgreSQL), 389, 397**
- SHOW CREATE FUNCTION statement, 526–529**
- SHOW\_SEC\_SERVICES function (Sybase), 271, 272**
- SID (security identification number), 214**
- SIGN function**
  - ANSI, 71, 81–82
  - MySQL, 332
  - Oracle, 123, 125–126
  - PostgreSQL, 389, 397
  - Sybase, 277, 285
- SIGNAL statement, 52**
- Simple Aggregate Definition Language (SADL), 430**
- SIN function**
  - MySQL, 322, 332
  - PostgreSQL, 389
- sine, returning, 285, 332, 397**
- SINH function**
  - ANSI, 72, 82
  - Sybase, 277, 285

## SMALLINT function (IBM DB2)

---

**SMALLINT function (IBM DB2), 163, 165**

**snowflake schema, 566**

**SORTKEY function (Sybase), 249, 252, 255–256**

**SOUNDEX function**

difference between two SOUNDEX values, returning,  
188–189, 252

IBM DB2, 141, 146

Microsoft SQL Server, 186, 193–194

MySQL, 337, 350

Oracle, 98, 103–104

Sybase, 249, 257

**SPACE function**

IBM DB2, 141, 146–147

Microsoft SQL Server, 186, 194

Sybase, 249

**sp\_addextendedproperty procedure, 503**

**sp\_depends procedure, 503, 519**

**sp\_dropextendedproperty procedure, 503**

**special register, 121, 167, 168, 169, 453**

**sp\_help procedure, 503, 519**

**sp\_helpjava procedure, 519**

**sp\_helpprotect procedure, 519**

**sp\_helptext procedure, 503**

**@@SPID function**

Microsoft SQL Server, 207, 211–212

Sybase, 300, 309

**spnc\_makefile.mk file, 441–442**

**sp\_OA procedures, 490**

**sp\_rename procedure, 496**

**sp\_trace\_generateevent procedure, 501**

**sp\_updateextendedproperty procedure, 503**

**sp\_who procedure, 212**

**SQL Query Analyzer software, 4, 489, 497–501**

**SQL Server. See specific function and topic**

**SQLCA (SQL communications area), 581**

**SQLCODE special register, 463, 465**

**SQLJ (Structured Query Language for Java)**

CREATE FUNCTION statement syntax, 510–513

data type, 513, 514–516

function

altering, 516–517

debugging, 517

deterministic/non-deterministic, 512

dropping, 517

error handling, 517–518

information about, returning, 518–519

null value, handling, 513–514

permission, 509

UDF, 510–511

JDBC requirement, 54

JVM requirement, 54

licensing, 509

standard, 54

**SQL\*Plus client (Oracle), 4, 432, 433, 442, 443**

**SQLPlus Worksheet client (Oracle), 4**

**SQLSTATE special register, 463, 465**

**@@SQLSTATUS function (Sybase), 300, 309–310**

**SQRT function**

ANSI, 72, 83

IBM DB2, 47

Microsoft SQL Server, 47

MySQL, 47, 322, 332

Oracle, 47

PostgreSQL, 47, 389, 398

standard identifier, 42

Sybase, 47, 277, 286

**SQUARE function (ANSI), 72, 82**

**standard deviation, returning, 43, 95–96, 332–333, 374**

**star schema, 566**

**statement. See also specific statement**

case sensitivity, 7–8

clause, 7

compound, 51

database connection pooling environment,  
execution in, 602

dynamic

Explain information, enabling/disabling, 172

index, recommended, 172

intra-partition parallelism degree, returning, 171

path, resolving, 173

query, 172, 579–581, 606

schema, returning current, 174

transform group used, returning, 171

UDF, executing from, 507

ESQL, 578–579

rows affected by, returning number of, 231–232

semidynamic, 580

transaction, statement grouping within, 628

**static SQL, 579**

**station, 717**

**STD function (MySQL), 322, 332–333**

**STDDEV function**

MySQL, 322, 332–333

Oracle, 91, 95–96

PostgreSQL, 371, 374

**STDDEV\_POP function**

IBM DB2, 47

Oracle, 47

standard identifier, 43

**STDDEV\_SAMP function**

IBM DB2, 47

Oracle, 47

standard identifier, 43

- STDEV function**  
 Microsoft SQL Server, 93  
 MySQL, 93
- STDEVP function (Microsoft SQL Server), 93**
- STR function**  
 Microsoft SQL Server, 187, 194–195  
 Sybase, 249, 257–258
- string function**  
 ANSI, 65–66  
 ESQL, 589–590  
 IBM DB2, 140–141  
 Microsoft SQL Server, 37, 185–187  
 MySQL, 39, 333–338  
 PostgreSQL, 40, 375–377  
 Sybase, 37, 247–249
- STR\_REPLACE function (Sybase), 69**
- Structured Query Language for Java. See SQLJ stub, 718**
- STUFF function**  
 Microsoft SQL Server, 187, 195  
 Sybase, 249, 258–259
- subscription reporting implementation, 553**
- SUBSTR function**  
 IBM DB2, 141, 147  
 Oracle, 99, 104–105
- SUBSTR2 function (Oracle), 104**
- SUBSTR4 function (Oracle), 104**
- SUBSTRB function (Oracle), 104**
- SUBSTRC function (Oracle), 104**
- SUBSTRING function**  
 IBM DB2, 46  
 Microsoft SQL Server, 46, 187, 196  
 MySQL, 338, 350–351  
 Oracle, 46  
 PostgreSQL, 46, 377, 386  
 standard identifier, 41  
 Sybase, 46, 249, 259
- SUBSTRING\_INDEX function (MySQL), 338, 351**
- SUM function**  
 ANSI, 61, 65  
 IBM DB2, 46  
 Microsoft SQL Server, 46  
 MySQL, 317, 319  
 Oracle, 46, 91, 96  
 PostgreSQL, 46, 371, 374–375  
 standard identifier, 42  
 Sybase, 46, 273, 275
- superaggregate, 94**
- SUSER\_ID function**  
 Microsoft SQL Server, 134  
 Sybase, 134, 288, 294–295
- SUSER\_NAME function (Sybase), 288, 295**
- SUSER\_SID function (Microsoft SQL Server), 213, 214**
- SUSER\_SNAME function (Microsoft SQL Server), 213, 214**
- SWITCH statement, 524**
- Sybase database platform. See specific function and topic**
- SYSCAT.COLUMNS view, 641**
- SYSCAT.DBAUTH view, 641**
- SYSCAT.FUNCTIONS view, 474, 641–642**
- SYSCAT.INDEXCOLUSE view, 641**
- SYSCAT.PROCEDURES view, 474, 641**
- SYSCAT.ROUTINES view, 474**
- SYSCAT.SCHEMATA view, 641, 642–643**
- SYSCAT.TABAUTH view, 641**
- SYSCAT.TABLES view, 641, 643–644**
- SYSCAT.VIEWS view, 641, 644–645**
- SYSCHEMSETS system table, 256**
- SYSCOLUMNS system table, 648, 649, 653–654**
- SYSCOMMENTS system table, 506–507, 648, 649–650**
- SYSCONSTRAINTS system table, 506**
- SYSDATABASES system table, 205, 292**
- SYSDATE function**  
 MySQL, 355, 362  
 Oracle, 115, 120–121
- SYSDEPENDS system table, 506**
- SYSDOMAIN system table, 653, 654**
- SYSFILES system table, 206**
- SYSFUN schema, 137, 449**
- SYSIBM schema, 137, 140, 173, 449**
- SYSLOGINS system table, 294, 298**
- SYSOBJECTS system table, 290, 293, 505–506, 648, 650–651**
- SYSSTAT.ROUTINES view, 474**
- SYSTABLE system table, 653, 654–655**
- System Catalog. See also specific system table and view**  
 Column catalog, 636  
 IBM DB2, 641–645  
 introduced, 635  
 Microsoft SQL Server, 648–652  
 MySQL, 655  
 Oracle, 636–640  
 PostgreSQL, 645–647  
 Privileges catalog, 636  
 Sybase, 652–655  
 table information, returning from, 636–639, 641, 646, 653–655  
 UDF information, returning from, 641–642  
 view information, returning from, 636, 639–640, 644–645, 647, 654–655
- system function, 492–493, 718, 719**
- system resource, 718**
- system UDF, 492–495**
- SYSTEMTIMESTAMP function, 117**
- SYSUSERS system table, 216, 648, 651–652**

## T

**table. See also specific table**

- alias, assigning, 181
- column
  - adding, 6
  - data type, assigning, 6
  - data warehouse column input, filtering, 569
  - function, column, 35, 37, 621
  - IDENTITY column, 6, 228–229, 234, 304
  - length, returning, 204–205, 289–290
  - name, assigning, 6
  - name, returning, 290
  - random value, populating with using generated SQL, 594–598
  - removing, 6
  - updating, 10
  - value within, returning minimum, 42
- constraint, returning, 643
- creating, 5–6
- data warehouse implementation, in
  - designing table for, 566–567, 571
  - filtering column input, 569
  - joining tables, 568
- dual, 73, 425
- ESQL, updating in, 580–581
- function, table, 37
- information about, returning from System Catalog, 636–639, 641, 646, 653–655
- joining
  - data warehouse implementation, in, 568
  - information about join, returning, 637
  - Microsoft SQL Server, 485
  - outer, 717
  - SELECT statement, using, 316, 370, 568
  - view, using, 623
- name, returning, 638
- refreshing, 174
- row
  - adding, 9
  - aggregate, 94
  - count, returning in ANSI, 63–64
  - count, returning in IBM DB2, 46, 643
  - count, returning in Microsoft SQL Server, 231–232
  - count, returning in MySQL, 318
  - count, returning in Oracle, 93–94
  - count, returning in PostgreSQL, 373
  - count, returning in Sybase, 274, 308–309
  - count, using to verify database migration success, 563
  - ESQL single-row function, 582–583
  - function, rowset, 717
  - interpolation, 44
  - number of rows affected by statement, returning, 231–232
  - rank, returning, 45
  - removing, 10–11
  - TOP N analysis, 8, 185
  - UDF row function, 536–537
  - values, returning individually versus returning as row, 606
  - summing all values in, 65, 96
  - temporary, 298, 316, 508
  - timestamp, returning, 643
  - transaction processing, locking during, 628
  - UDF, table-valued
    - FROM clause, optimizing using, 632
    - IBM DB2, 451, 454–456
    - Microsoft SQL Server, 480, 484–487
    - Oracle, 427, 430–431
    - PostgreSQL, 536–537
    - view, replacing using, 621–622
  - view
    - join, using in, 623
    - virtual table, as, 617
- TABLE\_NAME function (IBM DB2), 177, 181**
- TAN function**
  - ANSI, 72, 83
  - MySQL, 322, 333
  - Sybase, 277, 286
- tangent, returning, 72, 83, 279, 286, 333**
- TANH function (ANSI), 72, 83–84**
- telephone number, formatting, 630**
- text. See also character set**
  - binary value, converting to/from string, 339, 380
  - case, converting
    - ANSI, 68
    - IBM DB2, 66
    - Microsoft SQL Server, 190, 196–197
    - MySQL, 344, 352
    - Oracle, 66
    - PostgreSQL, 66, 380–381, 387
    - Sybase, 66
  - case sensitivity
    - query, 8
    - statement, 7–8
  - comma-delimited string set, 341, 342, 346
  - comparing strings in Sybase, 250–252
  - compressing/uncompressing string, 339–340, 352–353
  - concatenating strings
    - ANSI, 67–68
    - IBM DB2, 142
    - MySQL, 340–341
    - Sybase, 266

- data type, converting to character, 108, 110–113, 165–166, 194–195
- dictionary, 251
- encryption, string, 168, 240, 382
- escaping, 114, 532
- ESQL string function, 589–590
- field string index value, returning, 341–342
- file, returning as string, 367
- function, string
  - ANSI, 65–66
  - ESQL, 589–590
  - IBM DB2, 140–141
  - Microsoft SQL Server, 37, 185–187
  - MySQL, 39, 333–338
  - PostgreSQL, 40, 375–377
  - Sybase, 37, 247–249
- generating unique string, 179
- hexadecimal representation, returning, 164, 240, 270–271, 342
- length of string, returning
  - IBM DB2, 143–144
  - Microsoft SQL Server, 189–190
  - MySQL, 68–69, 335, 344–345, 353
  - Oracle, 69
  - PostgreSQL, 68, 378–379, 381, 383
  - Sybase, 250
- multibyte, checking if character is, 348
- number expressed as string, truncating, 147–148
- padding
  - IBM DB2, 146–147
  - Microsoft SQL Server, 194, 226
  - MySQL, 346, 349
  - Oracle, 101–102
  - PostgreSQL, 381–382, 385
  - Sybase, 257
- pointer, returning, 287–288, 481
- position of pattern, returning
  - Microsoft SQL Server, 191
  - Sybase, 253–254
- position of string, returning
  - IBM DB2, 144
  - Microsoft SQL Server, 188
  - MySQL, 345
  - PostgreSQL, 383–384
  - Sybase, 249–250
- position, replacing substrings at specified
  - IBM DB2, 142–143
  - Microsoft SQL Server, 195
  - MySQL, 342–343
  - PostgreSQL, 383
  - Sybase, 255, 258–259
- position, returning string at specified
  - IBM DB2, 143, 147
  - Microsoft SQL Server, 189, 196
  - MySQL, 343, 344, 349, 351
  - Oracle, 104–105
  - PostgreSQL, 386
- query
  - case sensitivity, 8
  - process handle, returning query text based on, 240–241
- regular expression, 106, 188, 250, 386
- repeating string
  - Microsoft SQL Server, 145, 186, 192–193
  - MySQL, 348
  - PostgreSQL, 384–385
  - Sybase, 254
- reversing string
  - Microsoft SQL Server, 193
  - MySQL, 337, 349
  - Sybase, 254–255
- search result, replacing every occurrence with specified text
  - ANSI, 69
  - IBM DB2, 145–146
  - Microsoft SQL Server, 192
  - MySQL, 348
  - Oracle, 103
  - PostgreSQL, 385
- sound representation of character, working with
  - IBM DB2, 141, 146
  - Microsoft SQL Server, 186, 193–194
  - MySQL, 337, 350
  - Oracle, 98, 103–104
  - Sybase, 249, 252, 257
- template, translating from, 105–106
- trimming
  - IBM DB2, 141, 144–145
  - Microsoft SQL Server, 186, 190–191
  - MySQL, 336, 346, 350, 352
  - Oracle, 98, 102
  - PostgreSQL, 378, 382, 386, 387
  - Sybase, 248, 252–253
- TEXTPTR function**
  - Microsoft SQL Server, 481
  - Sybase, 287
- TEXTVALID function (Sybase), 287–288**
- thread server connection ID, returning, 366**
- 3GL (third-generation language)**
  - ESQL, using in, 577, 578
  - 4GL compared, 19
  - procedure, 20–21

## time

- clock, considerations when resetting, 179
- current, returning
  - IBM DB2, 175
  - Microsoft SQL Server, 202, 481
  - MySQL, 356, 362
  - Oracle, 120–121
  - PostgreSQL, 400, 402–403
  - Sybase, 264
  - system time, 27–28, 115, 120–121, 202
  - time zone, of specified, 42, 118–119
  - UTC time, 202, 481
- current, returning interval between timestamp and, 399–400
- ESQL time function, 588–589
- field, returning time value from, 117–118, 401
- finite, checking if interval is, 402
- formats supported, 198
- formatting
  - IBM DB2, 154
  - Microsoft SQL Server, 198, 200–201
  - MySQL, 358–359, 364
  - Oracle, 119, 122
  - Sybase, 267
- hour
  - adding to/subtracting from input argument, 261, 357
  - returning hour part of time value, 153–154, 200, 361
  - second value, converting to/from hours, minutes, and seconds, 363, 364
- I/O busy time, returning, 305
- ISO format, 154, 160
- microsecond, adding to/subtracting from input argument, 357
- millisecond part of time value, returning, 155, 200
- minute
  - adding to/subtracting from input argument, 357
  - returning minute part of time value, 156, 200, 361
  - second value, converting to/from hours, minutes, and seconds, 363, 364
- second
  - adding to/subtracting from input argument, 357
  - hours, minutes, and seconds, converting second value to/from, 363, 364
  - midnight, returning seconds from, 155–156
  - returning second part of time value, 157, 200, 363
- server
  - boot time, returning, 301
  - busy time, returning, 235–236, 303, 305, 481
  - date in server time zone, returning current, 42
  - idle time, returning, 235–236, 305, 481
  - system time, returning, 27–28, 115, 120–121, 202
  - tick, returning, 236, 310, 481

## timestamp

- calculating difference between two timestamps, 159
- comparing two timestamps, 243, 295–296
- current, returning, 175–176, 225, 295, 402
- database instance timestamp, returning, 225
- date/time value, returning from, 42, 117, 158, 400–401
- finite, checking if, 402
- IBM DB2, 159
- interval between current time and, returning, 399–400
- ISO format, 160
- Microsoft SQL Server, 159
- schema timestamp, returning, 642
- Sybase, 159
- table timestamp, returning, 643
- truncating, 401
- UNIX timestamp, returning, 361, 365
- truncating time value, 115, 121–122, 401
- UTC, 117, 176, 202, 481
- zone
  - current, returning, 176
  - current time at specified zone, returning, 42, 118–119
  - database time zone, returning, 116–117
  - server time zone, returning current date in, 42
  - session time zone, returning, 114, 116–117
  - session time zone, returning current date in, 42
  - session time zone, returning current time in, 42
  - timestamp, extracting from, 117
  - UTC, calculating difference from, 176

**TIME function (IBM DB2), 149, 157–158**

**TIME\_DIFF function (MySQL), 118**

**TIME\_FORMAT function (MySQL), 356, 364**

**TIMEOFDAY function (PostgreSQL), 399, 403**

**TIMESTAMP function (IBM DB2), 149, 158**

**TIMESTAMPDIFF function (IBM DB2), 149, 159**

**TIMESTAMP\_FORMAT function (IBM DB2), 150, 160**

**TIMESTAMP\_ISO function (IBM DB2), 150, 160**

**@@TIMETICKS function**

- Microsoft SQL Server, 235, 236, 481
- Sybase, 300, 310

**TIME\_TO\_SEC function (MySQL), 356, 364**

**TOAD integrated development environment, 718**

**TO\_CHAR function (Oracle), 106, 110–113**

**TO\_DAYS function (MySQL), 356, 364–365**

**TO\_NCHAR function (Oracle), 113**

**Tools ⇨ Object Browser ⇨ Show/Hide (Query Analyzer), 497**

**TOO\_MANY\_ROWS error, 434**

**TOP N analysis, 8, 185**

**@@TOTAL\_ERRORS function**

- Microsoft SQL Server, 235, 236–237, 481
- Sybase, 300, 310–311

**@@TOTAL\_READ function**

Microsoft SQL Server, 235, 237, 481  
 Sybase, 300, 311

**@@TOTAL\_WRITE function**

Microsoft SQL Server, 235, 237–238, 481  
 Sybase, 300, 311

**ToUpper function (UNIX), 14****@@TRANCHAINED function (Sybase), 300, 311–312****@@TRANCOUNT function**

Microsoft SQL Server, 218, 232–233  
 Sybase, 300, 312

**transaction**

application, transaction heavy, 605  
 chained mode, 311–312  
 committing, 447, 503, 628  
 database, transactional, 628  
 locking table during processing, 628  
 nesting level, returning, 312  
 performance, optimizing, 627–628  
 read, dirty, 628  
 rolling back, 447, 628  
 session, returning number of pending transactions for, 232–233  
 statement grouping within, 628  
 status, returning, 312–313  
 UDF, using  
   IBM DB2, 473  
   Microsoft SQL Server, 502, 507  
   MySQL, 522  
   Oracle, 440, 447  
 workstation ID of current session, tracking using, 227

**Transact-SQL (T-SQL) language, 3, 54–55, 226, 245.**

*See also specific Microsoft SQL Server function*

**transform group, returning, 171****TRANSLATE function**

IBM DB2, 46, 165–166  
 Microsoft SQL Server, 46  
 Oracle, 46, 99, 105–106, 107, 113  
 PostgreSQL, 46  
 REPLACE function compared, 103, 165  
 standard identifier, 41  
 Sybase, 46  
 wildcard, using, 165

**@@TRANSTATE function (Sybase), 300, 312–313****TRIM function**

MySQL, 338, 352  
 Oracle, 99, 102, 145  
 PostgreSQL, 377, 387  
 standard identifier, 41

**TRUNC function**

ANSI, 72, 84  
 IBM DB2, 141, 147–148

number usage, 72, 84, 123, 127  
 Oracle, 115, 121–122, 123, 127  
 PostgreSQL, 389, 398

**TRUNCATE function**

ANSI, 72, 84  
 IBM DB2, 141, 147–148  
 MySQL, 322, 333

**try-catch block, 611, 612****TSEQUAL function**

Microsoft SQL Server, 239, 242–243  
 Sybase, 289, 295–296

**T-SQL (Transact-SQL) language, 3, 54–55, 226, 245.**

*See also specific Microsoft SQL Server function*

**TYPE\_ID function (IBM DB2), 178, 182****TYPE\_NAME function (IBM DB2), 178, 182****U****UCASE function**

IBM DB2, 197  
 MySQL, 338, 352

**UDB (Universal Database). See also specific IBM DB2 function**

ANSI compliance, 16  
 data type conversion, implicit, 138, 142  
 date formats available, 154  
 platform support, 137  
 query syntax, 138–140  
 register, special, 121, 167, 168, 169, 453  
 SYSIBM schema, 137, 140, 173, 449

**UDF (user-defined function)**

aggregate, 427, 430  
 altering  
   IBM DB2, 459  
   Microsoft SQL Server, 495–496  
   MySQL, 525  
   Oracle, 431–432  
   SQLJ, 516–517  
   Sybase, 516–517  
 argument  
   data type, 425, 510  
   Oracle, 424–425, 426  
   Sybase, 510  
 array static value support, declaring, 418  
 built-in function  
   Microsoft SQL Server UDF, using in, 481  
   overriding, 458–459  
   performance compared, 633  
   system UDF, 492–493  
 calling, 445–447, 483, 523, 543  
 compilation  
   interpreted mode, 432, 441, 443  
   native, 422, 432, 441, 443

## UDF (user-defined function) (continued)

---

### UDF (user-defined function) (continued)

Oracle, 421–422, 431–432, 440–443

smart, 422

data type

argument, 425, 510

UDT, using, 476, 479, 534–536

data warehouse implementation using

concatenation, 574–575

disassembling/reassembling data, 575–576

query, 568–569

standardization, 570–572

summarization, 572–574

database object, as, 443

debugging

IBM DB2, 461–463

Microsoft SQL Server, 497–501

MySQL, 526

Oracle, 432–434

PostgreSQL, 538

SQLJ, 517

Sybase, 517

deterministic/non-deterministic, declaring

ANSI, 418

IBM DB2, 452

Microsoft SQL Server, 481, 507

MySQL, 522

Oracle, 426

Sybase, 510

dropping

ANSI, 419

IBM DB2, 459–461

Microsoft SQL Server, 496–497

MySQL, 526

Oracle, 432

PostgreSQL, 537–538

SQLJ, 517

Sybase, 517

dynamic SQL, executing from, 507

embedded, 578

encryption, 480, 488

error handling

IBM DB2, 463–464

Microsoft SQL Server, 502–503

MySQL, 526

Oracle, 434–437

PostgreSQL, 538

SQLJ, 517–518

Sybase, 517–518

external

embedded UDF, 578

IBM DB2, 451–452, 453

internal versus, 417

Microsoft SQL Server, 504

MySQL, 525, 526

performance, optimizing via limiting, 453

PostgreSQL, 531

Sybase, 511, 512, 519

federated object configuration, 453

generating SQL using, 592–593, 596–598

identifier, unique, 452, 641

independent, specifying as, 427

information about, returning

IBM DB2, 474–475, 641–642

Microsoft SQL Server, 503–507

MySQL, 526–529

Oracle, 443–445

PostgreSQL, 533, 539–542

SQLJ, 518–519

Sybase, 518–519

System Catalog, from, 641–642

inheritance, 419, 457, 538

inline, 484–486

introduced, 49

Java, implementing in, 510–511, 516–519

JDBC driver, developing using, 56

language, declaring, 418, 511, 522

migrating database using, 560–564

name, assigning

IBM DB2, 452

Microsoft SQL Server, 478–480

Oracle, 424

Sybase, 510

uniqueness, 452, 472, 478

null value support, 418, 513–514

overloading

IBM DB2, 471–473

MySQL, 522

Oracle, 439–440

PostgreSQL, 538–539

ownership, 478

package, 424, 428, 437–438, 446

parallel execution environment

IBM DB2, 453

Oracle, 427

parameter

number of parameters, maximum, 478–479

value, default, 479, 483, 484

performance, optimizing, 425, 442, 453, 632

permission

IBM DB2, 450–451

Microsoft SQL Server, 477–478

MySQL, 521

- Oracle, 422–423, 426
  - PostgreSQL, 532–533
  - Sybase, 509
  - pipelined, 427, 430–431
  - predicate specification, 453
  - procedure, using, 475, 490–492, 507–508
  - query, 533–534, 549–553, 568–569
  - recursive
    - IBM DB2, 454, 476
    - Microsoft SQL Server, 488–489
    - Oracle, 428–429
  - report function, creating, 546–549
  - routine versus, 415
  - row function, 536–537
  - scalar
    - IBM DB2, 451, 452–456
    - Microsoft SQL Server, 480, 482–484
  - schema binding, 480, 487–488
  - sourced, 452, 456–457, 458–459, 473
  - SQLJ-based, 510–511
  - square root, returning using, 457, 458
  - stand-alone, 424, 437
  - SYSFUN schema, 137, 449
  - system UDF, 492–495
  - table-valued
    - FROM clause, optimizing using, 632
    - IBM DB2, 451, 454–456
    - Microsoft SQL Server, 480, 484–487
    - Oracle, 427, 430–431
    - PostgreSQL, 536–537
    - view, replacing using, 621–622
  - template
    - IBM DB2, 452, 458
    - Microsoft SQL Server, 489
  - tracing, 485, 493, 501, 588
  - transaction handling, using in
    - IBM DB2, 473
    - Microsoft SQL Server, 502, 507
    - MySQL, 522
    - Oracle, 440, 447
  - user, assigning to, 426
  - view, 443–445, 481–482, 504–507, 621–622
- UDT (user-defined type)**
- identifier, returning, 182
  - name, returning, 182
  - PostgreSQL, 534–536
  - UDF, using in, 476, 479, 534–536
- UGA (user global area) memory, 438**
- UID function (Oracle), 128, 134**
- UNCOMPRESS function (MySQL), 338, 352–353**
- UNCOMPRESSED\_LENGTH function (MySQL), 338, 353**
- underscore (\_)**
- UDF name prefix, 479
  - wildcard character, 253
- undocumented function, 37, 38**
- @@UNICHSIZE function (Sybase), 301, 313**
- Unicode character set**
- argument, disassembling, 110
  - data type Unicode representation, returning, 108
  - hexadecimal, 114
  - input argument, returning Unicode representation of, 108
  - number code
    - character corresponding to, returning, 191
    - character, returning corresponding number code, 196
  - scalar value, returning, 259
  - size of a Unicode character, returning, 313
  - trimming blank from Unicode string, 190
  - UCS2, 104, 114
  - UCS4, 104
- UNICODE function (Microsoft SQL Server), 187, 196**
- UNISTR function (Oracle), 107, 108, 113–114**
- Universal Database. See UDB**
- UNIX**
- shell function, 14
  - timestamp, returning, 361, 365
  - ToUpper function, 14
- UNIX\_TIMESTAMP function (MySQL), 356, 365**
- UPDATE statement, 10, 507**
- UPPER function**
- ANSI, 66, 68
  - IBM DB2, 46
  - Microsoft SQL Server, 46, 196–197
  - MySQL, 352
  - Oracle, 46
  - PostgreSQL, 46, 377, 387
  - standard identifier, 41
  - Sybase, 46
- US7ASCII character set, 109**
- USCALAR function (Sybase), 196, 249, 259**
- user**
- account, migrating, 557
  - database user information returning, 213, 214–216, 296–297, 411
  - server user
    - information about, returning, 294–295, 308
    - validating user ID, 298–299
  - session user, returning, 176–177, 225–226, 412
  - System Catalog user information, returning
    - introduced, 636
    - Microsoft SQL Server, 651–652

### **user (continued)**

Oracle, 637–638  
PostgreSQL, 646–647  
UDF, assigning to, 426

### **USER function**

IBM DB2, 170, 177  
Microsoft SQL Server, 213, 214–215  
PostgreSQL, 409, 412  
Sybase, 289, 296

### **user global area (UGA) memory, 438**

**USER\_CATALOG view, 638**

**user-defined function. See UDF**

**user-defined type. See UDT**

### **USER\_ID function**

Microsoft SQL Server, 213, 215  
Sybase, 289, 296–297

**USER\_MVIEW system tables, 637**

### **USER\_NAME function**

Microsoft SQL Server, 213, 215–216  
Sybase, 289, 297

**USER\_PROCEDURES view, 444**

**USER\_SOURCE view, 444**

**USER\_TABLES view, 639**

**USER\_VIEWS view, 639–640**

**UTC (Coordinated Universal Time), 117, 176, 202, 481**

**UTC\_DATE function, 117**

## **V**

**VALID\_NAME function (Sybase), 289, 297–298**

**VALID\_USER function (Sybase), 289, 298–299**

**VALUE function (IBM DB2), 178**

**VAR function (Microsoft SQL Server), 93**

**VARCHAR function (IBM DB2), 163, 166**

### **variable**

encapsulation, 715  
host, 579  
memory  
  pointer, 23  
  storage in/retrieval from, 22–23  
procedural extension implementation, 51  
scope, 24–25, 298

### **VARIANCE function**

MySQL, 93  
PostgreSQL, 371, 375

**VARP function (Microsoft SQL Server), 93**

### **VAR\_POP function**

IBM DB2, 47  
Oracle, 47  
standard identifier, 43

### **VAR\_SAMP function**

IBM DB2, 47  
Oracle, 47  
standard identifier, 43

**VB (Visual Basic).Net, creating login application using, 608–611**

### **@@VERSION function**

Microsoft SQL Server, 207, 212  
Sybase, 301, 313–314

**VERSION function (PostgreSQL), 409, 412–413**

**@@VERSION\_AS\_INTEGER function (Sybase), 301, 314**

**V\$FIXED\_VIEW\_DEFINITION system table, 637**

**view. See also specific view**

average, returning using, 620–621  
creating, 481–482, 617–618  
described, 617  
grouping, 620  
horizontal, 618  
information about, returning from System Catalog, 636, 639–640, 644–645, 647, 654–655  
materialized, 636–637  
name, returning, 638  
permission, 617  
query, filling using, 617–619, 620  
readability of data, improving using, 623  
sorting, 620  
table  
  join, using view in, 623  
  virtual table, view as, 617  
UDF, 443–445, 481–482, 504–507, 621–622  
updating, 619  
vertical, 619

**Visual Basic (VB).Net, creating login application using, 608–611**

**VSIZE function (Oracle), 128, 134–135**

**V\$TIMEZONE\_NAMES performance view, 118**

## **W**

### **WAITFOR DELAY function**

Microsoft SQL Server, 243  
Sybase, 295

**warehousing. See data warehouse**

**WEEK function (IBM DB2), 150, 160–161**

**WEEK\_ISO function (IBM DB2), 150, 161**

**WHEN statement, 219**

**WHILE statement, 52, 523**

**WIDTH function (PostgreSQL), 404, 408–409**

---

**X**

**XML (eXtensible Markup Language), returning query result as, 184**

**XOR operator, 584**

`xp_msver` **procedure, 212**

**Y**

**YEAR function**

IBM DB2, 150, 161–162

Microsoft SQL Server, 198, 203

**Yip, Paul (*DB2 SQL Procedural Language for Linux, UNIX and Windows*), 56**

**Z**

**0 (zero) division error, 435–437, 463–464, 502, 518**

