INDEX

Symbols

+ (addition operator), 79
& (ampersand operator), 270
:= (assignment operator), 359
* (asterisk)
    as multiplication operator, 79
    as wildcard in SELECT statement, 30
\ (backslash), 249
    escaping characters with, 255
, (comma), 60
||/ (cube root operator), 81
{} (curly brackets), 249
    denoting an array in query output, 90
<-> (distance operator), 271
@@ (double at sign match operator), 266
:: (double-colon CAST operator), 56
$$ (double-dollar quoting), 351
|| (double-pipe concatenation operator), 172, 261, 329
" (double quote), 61, 118
! (exclamation point)
    as factorial operator, 81
    as negation, 250, 270
^ (exponentiation operator), 80
/ (forward slash)
    as division operator, 80
    in macOS file paths, 44
> (greater-than comparison operator), 35
>= (greater-than or equals comparison operator), 35
- (hyphen subtraction operator), 79, 331
@> and <@ (JSON containment operators), 315
? and ?| (JSON existence operators), 315
-> and - (JSON field and element extraction operators), 310, 310–311
#- (JSON path deletion operator), 331
#> and ##> (JSON path extraction operators), 310, 314
< (less-than comparison operator), 35
<= (less-than or equals comparison operator), 35
<> (not-equal comparison operator), 35
!= (not-equal comparison operator), 35
() (parentheses)
    to designate order of operations, 38
    to specify columns for importing, 70
% (percent sign)
    as modulo operator, 80
    wildcard for pattern matching, 37
| (pipe character)
    as delimiter, 44, 63, 72
    to redirect output, 295
; (semicolon), 19
|/ (square root operator), 81
~* (tilde-asterisk case-insensitive matching operator), 250
~ (tilde case-sensitive matching operator), 250
_ (underscore wildcard for pattern matching), 37

A

adding numbers, 79
    across columns, 82
addition operator (+), 79
aggregate functions, 86, 142
    avg(), 86, 200
    binary (two-input), 186
    count(), 143, 159
    filtering with HAVING, 153

aggregate functions *(continued)*
GROUP BY requirement, 146
interviewing data, 159
max(), 145
min(), 145
PostgreSQL documentation, 143
sum(), 86, 149
aliases for table names, 107, 151
ALTER COLUMN statement, 133
ALTER SYSTEM command, 394
ALTER TABLE statement, 165
ADD COLUMN, 165, 169, 286
ADD CONSTRAINT, 133
ALTER COLUMN, 165
DROP COLUMN, 165, 176
RENAME TO, 180
to restart sequence, 127
table constraints, adding and removing, 132
American National Standards Institute (ANSI), 305
SQL standard, xxiv
ampersand operator (&), 270
Amtrak trip data, 218
ANALYZE keyword
with EXPLAIN statement, 135
with VACUUM, 391
AND logical operator combining comparison operators, 38, 114
antimeridian, 66
API (application programming interface), 318
array, 90, 256
array_length() function, 252
constructor example, 90
denoted by curly brackets in query output, 90, 256
extracting element with JSON operators, 312
functions, 91
index positions, 313
in JSON, 306, 332
notation in query, 260
passing into ST_MakePoint(), 284
returned from regexp_match(), 255
unnest() function, 91
array_length() function, 252
ASC keyword in ORDER BY clause, 32
AS keyword
declaring table aliases with, 107, 113
renaming columns in query results with, 82, 106
assignment operator (:=), 359
asterisk (*)
as multiplication operator, 79
as wildcard in SELECT statement, 30
attribute, 22
auto-incrementing integers, 23, 31, 46
gaps in sequence, 47, 126
identity column SQL standard, 46
overriding values, 127
restarting sequence, 127
as surrogate primary key, 125
using IDENTITY, 46, 122, 125
autovacuum, 390
editing server setting, 393
time of last vacuum, 391
avg() function, 86, 200, 226

B

backslash (\), 249
escaping characters with, 255
backups
column, 168
improving performance when updating tables, 179
restoring values from another table, 170
tables, 167
bell curve. *See* normal distribution
BETWEEN comparison operator, 35, 229
inclusive property, 36
BINARY file format, 63
birth data, US, 404
Boolean expression
in constraint, 131
in table join, 94
Boolean value, 94
used in table join, 94
B-tree index, 134
Bureau of Labor Statistics, 401

C

CALL command, 352
camel case, 27, 118

carriage return in text files, 63
Cartesian product as result of CROSS JOIN, 103
CASCADE keyword, 129
case sensitivity
  with ILIKE operator, 37
  with LIKE operator, 37
CASE statement, 241
  in common table expression, 242
  ELSE clause, 242
  syntax, 241
  with trigger, 358
  in UPDATE statement, 262
  WHEN clause, 241, 359
CAST() function, 55
  shortcut notation, 56
categorizing data, 241
char, 42
character string types, 42
  char, 42
  functional difference from number types, 45
  performance in PostgreSQL, 43
  text, 42
  varchar, 42
char_length() function, 246
CHECK constraint, 130
classify_max_temp() user function, 358
clock_timestamp() function, 208
Codd, Edgar F., xxiv, 93
codes, distinguishing from numbers, 65
column, 22
  adding numbers in, 86
  alias, 82
  alter data type, 165
  avoiding spaces in name, 119
  deleting, 176
  indexes, 136
  populating new during backup, 179
  retrieving in queries, 31
  updating values, 166
comma (,), 60
command line, 363
  advantages of using, 364
  createdb command, 383
  psql, 370
  setup, 364
    macOS, 368
    PATH environment variable, 364
    Windows, 364
  shell programs, 368
COMMIT statement in transaction block, 177
common table expression (CTE), 234
  with CASE statement, 242
  defining, 234
comparison operators, 35
  combining with AND and OR, 38
composite primary key, 121
concatenation, 172
conditional expression, 241
constraints, 23, 120
  adding and removing, 132
  CHECK, 130, 185
  column vs. table, 121, 123
  CONSTRAINT keyword, 95, 123, 130
  foreign key, 128
  NOT NULL, 132
  primary key, 95, 121, 123
  UNIQUE, 96, 131
  violations when altering table, 166
constructor, 90
Coordinated Universal Time (UTC), 52, 206
  UTC offset, 52, 206, 219
COPY statement
  DELIMITER option, 63
  description of, 59
  exporting data, 43, 72
  FORMAT option, 63
  FROM keyword, 62
  HEADER option, 63
  importing data, 62
  JSON import, 310
  naming file paths, 44
  QUOTE option, 63
  specifying columns to import, 68
  specifying file formats, 63
  specifying rows to import, 70
  TO keyword, 72, 216

COPY statement *(continued)*
WHERE clause, 70
WITH keyword, 62
corr() function, 186
correlation does not imply causality, 189, 192
interpreting correlation coefficients, 186
correlated subquery, 224
count() function, 143, 159, 227
distinct values, 144
with GROUP BY, 147
on multiple columns, 148
values present in a column, 144
counting
distinct values, 144
missing values displayed, 161
rows, 143
using pgAdmin, 144
CREATE DATABASE statement, 19
createdb utility, 383
CREATE EXTENSION statement, 237, 276, 352
CREATE FUNCTION statement, 347
CREATE INDEX statement, 134, 136
CREATE TABLE statement, 22
backing up a table with, 167
declaring data types, 41
temporary table, 71
CREATE TRIGGER statement, 356
CREATE VIEW statement, 339
CROSS JOIN keyword, 103, 236
crosstab() function, 240
syntax example, 239
cross tabulations, 236
installing the tablefunc module, 237
CSV (comma-separated values), 60
export using COPY, 72
header row, 61
opening with text editor, 2
cube root operator (|||/), 81
curly brackets ({}), 249
denoting an array in query output, 90
current_date function, 208
current_time function, 208
current_timestamp function, 208
cut points, 88

D

data
structured and unstructured, 245
telling its story, 399
database
backup and restore, 395
connecting to, 20, 22
create from command line, 383
creating, 17, 19
maintenance, 387
database management system, 12, 19
database server, 12, 19
data dictionary, 41
data types, 41
bigint, 45
bigserial, 23, 45, 46
boolean, 55
char, 42
character string types, 42
date, 22, 51
date and time types, 51
decimal, 47
declaring with CREATE TABLE, 41
double precision, 48
full-text search, 265
geography, 280
geometry, 281
importance of using appropriate type, 41
integer, 45
interval, 51
json, 54
jsonb, 54
modifying with ALTER COLUMN, 165
number types, 44
numeric, 23, 47
real, 48
returned by math operations, 78
serial, 31, 45, 46
serial types, 46
smallint, 45
smallserial, 45, 46
text, 42
time, 51
timestamp, 51, 204
transforming values with CAST(), 55
tsquery, 266

tsvector, 265
varchar, 23, 42
date data types
date, 51
interval, 51
date_part() function, 205, 215, 241
dates
calculations with, 212
input format, 22, 25, 36, 52, 204, 205
matching with regular expressions, 253
setting default style, 394
Davis, Robert, 401
daylight saving time, 210
deciles, 90
decimal data types, 47
decimal degrees, 66
DELETE CASCADE statement with foreign key constraint, 129
DELETE statement, 70
removing rows matching criteria, 175
with subquery, 225
delimited text files, 59
text qualifiers, 61
delimiter, comma as most common, 60
DELIMITER keyword with COPY statement, 63
dense_rank() function, 193
derived table, 225
joining, 226
DESC keyword in ORDER BY clause, 32
dirty data, 29, 157
cleaning, 157
foreign keys help to avoid, 129
when to discard, 165
distance operator (<->), 271
DISTINCT keyword, 33
with count(), 144
in IS DISTINCT FROM clause, 169
on multiple columns, 34
division, 79
finding the remainder, 80
integer vs. decimal, 79, 80
documenting code, 41
double at sign match operator (@@), 266
double-colon CAST operator (::), 56
double-dollar quoting ($$), 351
double-pipe concatenation operator (||), 172, 261, 329
double quote ("), 61, 118
DROP statement
COLUMN, 176
INDEX, 137
TABLE, 71, 176
duplicate data created by spelling variations, 160

E

Eastern Standard Time (EST), 52
eliminating duplicate values in query, 34
ELSE clause in CASE statement, 242
entity, 19
environment variables, 364
epoch, 206, 320
equals comparison operator (=), 35
error messages
CSV import failure, 67, 69
foreign key violation, 129
out of range, 46
primary key violation, 124
relation already exists, 119
UNIQUE constraint violation, 132
when using CAST(), 56
escaping characters, 255
EST (Eastern Standard Time), 52
exclamation point (!)
as factorial operator, 81
as negation, 250, 270
EXISTS operator, 229
with subquery, 230
in WHERE clause, 167
EXPLAIN statement, 135
output of command, 135
exponentiation operator (^), 80
exporting data
all data in table, 72
header row, including, 63
limiting columns, 73
from query results, 73, 216
using COPY statement, 59, 72, 216
using pgAdmin wizard, 74
using the command line, 380

expressions, 53, 224
conditional, 241
subquery, 229
extract() datetime function, 206

F

factorial() function, 81
factorials, 80
false as Boolean value, 94
Federal Information Processing Standards (FIPS), 65, 295
field, 22
file paths
import and export file locations, 62
naming conventions for operating systems, 44, 62
filtering rows
HAVING clause, 153, 160
with subquery, 224
WHERE clause, 35, 224
findstr Windows command, 162
FIPS (Federal Information Processing Standards), 65, 295
fixed-point numbers, 47
floating-point numbers, 48
inexact math calculations, 49
Food Safety Inspection Service, 158
foreign key
creating with REFERENCES keyword, 128
definition, 96, 128
formatting SQL for readability, 27
forward slash (/)
as division operator, 80
in macOS file paths, 44
FROM keyword with COPY, 62
FULL OUTER JOIN keywords, 102
full-text search, 265
adjacent words, locating, 271
data types, 265
highlighting terms, 269
language configurations, 266
lexemes, 265
multiple terms in query, 270
querying, 268
ranking results, 271
setting default language, 394
table and column setup, 267
to_tsquery() function, 266
to_tsvector() function, 265
ts_headline() function, 269
ts_rank() function, 272
ts_rank_cd() function, 272
using GIN index with, 268
functions, 337
difference with procedures, 349
creating, 346
full-text search, 265
IMMUTABLE keyword, 348
RAISE NOTICE keywords, 351
RETURNS keyword, 348
specifying language, 348
string, 246
structure of, 348

G

generalized inverted index (GIN), 134, 268, 310
generalized search tree (GiST), 134, 286
generate_series() function, 209, 241, 389
GeoJSON, 276, 319
getting help when code goes bad, 26
GIN (generalized inverted index), 134, 268, 310
GIS (geographic information systems), 276
decimal degrees, 66
GiST (generalized search tree), 134, 286
GitHub, downloading code resources from, 3
graphical user interface (GUI), 292
list of tools, 407
greater-than comparison operator (>), 35
greater-than or equals comparison operator (>=), 35
grep Linux command, 162
GROUP BY clause
with aggregate functions, 146
eliminating duplicate values, 146
on multiple columns, 147

H

HAVING clause, 153, 160
HEADER keyword, 63
header row
  in CSV file, 61, 64
  ignoring during import, 61
hyphen subtraction operator (-), 79, 331

I

identifiers
  avoiding reserved keywords, 119
  enabling mixed case, 118
  naming, 27, 118
  quoting, 119
IDENTITY keyword, 46, 125
ILIKE comparison operator, 35, 37
  case-insensitive search, 37
importing data, 59, 62
  adding default column value, 71
  choosing a subset of columns, 68
  choosing a subset of rows, 70
  ignoring header row in text files, 61, 63
  from non-text sources, 60
  from text file format, 62
  using COPY statement, 59
  using pgAdmin wizard, 74
  using the command line, 380
IN comparison operator, 35, 172, 229
  with subquery, 230
indexes, 133
  and ANSI SQL standard, 133
  B-tree, 134
  considerations before adding, 137
  creating on columns, 136
  dropping, 137
  effect on performance, 135
  GIN, 134
  GiST, 134, 286
  not included with table backups, 168
  syntax for creating, 134
initcap() function, 246
inserting rows into a table, 25
INSERT statement, 25
Institute of Museum and Library Services, 140
integer data types, 45
  auto-incrementing, 46
  basic math operations, 79
  bigint, 45
  bigserial, 46
  difference in integer type capacities, 45
  integer, 45
  serial, 46
  smallint, 45
  smallserial , 46
International Date Line, 66
International Organization for Standardization (ISO), 52, 277
  SQL standard, xxiv
  time format, 204
interval data type
  calculations with, 53, 219
  cumulative, 220
  value options, 53
interviewing data, 29, 159
  across joined tables, 151
  artificial values as indicators, 146, 150
  checking for missing values, 32, 160
  correlations, 185
  counting rows and values, 143
  determining correct format, 32
  finding inconsistent values, 162
  malformed values, 162
  maximum and minimum values, 145
  rankings, 193
  rates calculations, 196
  statistics, 183
  summing grouped values, 149
  unique combinations of values, 34
ISO (International Organization for Standardization), 52, 277
  SQL standard, xxiv
  time format, 204

J

JOIN keyword, 94
  example of using, 100
  in FROM clause, 94
  with USING clause, 100

join types
CROSS JOIN, 103, 236
FULL OUTER JOIN, 102
JOIN (INNER JOIN), 100, 151
LEFT JOIN, 101
list of, 98
RIGHT JOIN, 101
joining tables, 93
derived tables, 226
multiple-table joins, 107
naming tables in column list, 106, 151
performing calculations across tables, 112
with set operators, 109
spatial joins, 300
specifying columns to query, 106
using JOIN keyword, 94, 97
json_agg() function, 329
jsonb_array_elements() function, 332
jsonb_array_elements_text() function, 332
jsonb_array_length() function, 331
jsonb_build_object() function, 329
jsonb_set() function, 330
JSON (JavaScript Object Notation), 54, 305
array, 306
considerations for using in database, 307
containment operators (@> and <@), 315
data types, 308
json, 309
jsonb, 309
existence operators (? and ?|), 315, 317
extraction operators, 310
field and element extraction operators (-> and -), 310, 310–311
functions
json_agg(), 329
jsonb_array_elements(), 332
jsonb_array_elements_text(), 332
jsonb_array_length(), 331
jsonb_build_object(), 329
jsonb_set(), 330
to_json(), 327
generating, 327
GeoJSON format, 319
importing, 309
indexing, 310, 319
json and jsonb data types, 54
key/value pairs, 54, 306
manipulating, 327
modifying key/value pairs, 329
object, 306
path extraction operators (#> and #>>), 310, 314
processing functions, 331
schema flexibility, 306
structure of, 54, 306
justify_interval() function, 221

K

key columns
foreign key, 96
primary key, 95
relating tables with, 94
key/value pairs, 306
extracting from JSON, 311
Korea standard time, 212

L

LATERAL subquery, 231
with FROM keyword, 231
with JOIN keyword, 232
latitude
in US Census data, 66
in well-known text, 278
least squares regression line, 190
left() function, 247
LEFT JOIN keywords, 101
length() string function, 163, 246
less-than comparison operator (<), 35
less-than or equals comparison operator (<=), 35
lexemes, 265
LIKE comparison operator, 35
case-sensitive search, 37
LIKE expression in UPDATE statement, 171
LIMIT clause, 67
limiting number of rows query returns, 67

linear regression, 189
least squares regression line, 190
linear relationship, 186
Linux
file path declaration, 44, 62
`grep` command, 162
system permissions, 44
Terminal setup, 370
literals, 25
`localhost`, 13, 20
`localtime` function, 208
logical operators, 38
longitude
positive and negative values, 68
in US Census data, 66
in well-known text, 278
`lower()` string function, 246

M

macOS
file path declaration, 44, 62
Terminal, 368
bash shell, 368
entering instructions, 369
setup, 368
useful commands, 369
`make_date()` function, 207
`make_time()` function, 207
`make_timestamptz()` function, 207
many-to-many table relationship, 105
map projection, 279
matching operators, 37
math
across joined table columns, 112
across table columns, 82
order of operations, 81
math operators, 78
addition (`+`), 79
division (`/`), 79
exponentiation (`^`), 80
factorial (`!`), 80
modulo (`%`), 79
multiplication (`*`), 79
square root (`|/`), 80
subtraction (`-`), 79
`max()` function, 145
median, 87
compared to average, 87, 217, 226
with `percentile_cont()` function, 89
Microsoft Access, xxv
Microsoft Excel, xxv
Microsoft SQL Server, 118, 237
`BULK INSERT` command, 60
Transact-SQL, xxiv
Microsoft Windows
Command Prompt
entering instructions, 366
setup, 364, 366
useful commands, 367
file path declaration, 43, 62
`findstr` command, 162
folder permissions, 3
`min()` function, 145
mode, 91
`mode()` function, 91
modifying data, 164
for consistency, 170
updating column values, 169
modulo, 79
testing for even numbers, 80
multiplying numbers, 79
MySQL, xxv
`LOAD DATA INFILE` statement, 60

N

NAICS (North American Industry Classification System), 197
naming conventions
camel case, 118
Pascal case, 118
snake case, 118, 120
National Center for Education Statistics, 401
National Center for Health Statistics, 404
natural primary key, 121, 159
New York City taxi data, 213
creating table, 213
exporting results, 216
finding busiest hour of day, 215
importing, 214
longest trip duration, 217
normal distribution, 192, 226
North American Industry Classification System (NAICS), 197
NoSQL databases, 307
`NOT` comparison operator, 35

not-equal comparison operator
<> syntax, 35
!= syntax, 35
NOT NULL keywords, 132
adding to column, 165
removing from column, 133, 165
now() function, 52, 208
NULL keyword
comparisons using IS DISTINCT FROM, 169
definition, 104
display in query results, 105
ordering with FIRST and LAST, 161
using in table joins, 104
using in WHERE clause, 104
number data types, 44
decimal types, 47
decimal, 47
double precision, 48
fixed-point type, 47
fixed-point vs. floating-point types, 48
floating-point types, 48
numeric, 47
real, 48
integer types, 45
bigint, 45
bigserial, 45
integer, 45
serial, 45
smallint, 45
smallserial, 45
usage considerations, 50

O

OGC (Open Geospatial Consortium), 277
ON clause used with JOIN, 94
one-to-many table relationship, 105, 128
one-to-one table relationship, 105
ON keyword used with DELETE CASCADE, 129
Open Geospatial Consortium (OGC), 277
operators
addition (+), 79
comparisons with, 35
division (/), 79
exponentiation (^), 80
factorial (!), 80
JSON, 310
modulo (%), 79
multiplication (*), 79
precedence, 81
prefix, 81
square root (|/), 80
subtraction (-), 79
suffix, 81
Oracle, xxiv
ORDER BY clause, 32
ASC, DESC options, 32
with count() function, 148
on multiple columns, 33
specifying columns to sort, 32
specifying NULLS first or last, 161
OR logical operator combining comparison operators, 38
OVER clause with window functions, 193

P

Pacific time zone, 52, 211
padding character columns with spaces, 42, 44
parentheses ()
to designate order of operations, 38
to specify columns for importing, 70
Pascal case, 118
path deletion operator, JSON (#-), 331
pattern matching
using LIKE and ILIKE, 37, 171
with wildcards, 37
Pearson correlation coefficient (*r*), 186
percentage
percent change, 85
creating user function, 347
formula, 85, 113, 347
of the whole, 84
percent_change() user function, 347
using with census data, 348
percentile, 88, 224
continuous vs. discrete values, 88
definition of, 88

percentile_cont() function, 88
finding median with, 217
in subquery, 224
using an array to enter multiple values, 90
percentile_disc() function, 88
percent sign (%)
as modulo operator, 80
wildcard for pattern matching, 37
pgAdmin, 11
alternatives to, 407
connecting to database, 20, 22
connecting to server, 12
drag and drop objects, 32
executing SQL, 20, 21
geometry viewer, 324
importing and exporting data, 74
installation
Linux, 9
macOS, 8
Windows, 4
keyword highlighting, 119
launching, 11
localhost, 13, 20
master password, 11
object browser, 12, 21, 24
preferences, 15
Query Tool, 14, 20
.sql files, 3
viewing data, 26, 31, 95, 144
viewing tables, 65
viewing table SQL statements, 24
viewing text in results grid, 254
views, 339
pg_ctl utility, 395
pg_dump utility, 395
pg_reload_conf() function, 394
pg_restore utility, 396
pg_size_pretty() function, 389
pg_total_relation_size() function, 389
pipe character (|)
as delimiter, 44, 63, 72
to redirect output, 295
PL/pgSQL, 347, 350
BEGIN ... END block, 356
IF ... THEN statement, 356
point, 66
position() string function, 246
PostGIS
creating spatial objects, 281
data types, 280, 281
displaying version, 276
functions
ST_AsText(), 295
ST_DFullyWithin(), 288
ST_Distance(), 289
ST_DWithin(), 287, 326
ST_GeogFromText(), 283, 288, 326
ST_GeometryType(), 300
ST_GeomFromText(), 281
ST_Intersection(), 301
ST_Intersects(), 301
ST_LineFromText(), 284
ST_MakeLine(), 284
ST_MakePoint(), 283, 324
ST_MakePolygon(), 284
ST_MpolyFromText(), 284
ST_PointFromText(), 283
ST_PolygonFromText(), 284
ST_SetSRID(), 286, 324
installation, 276
Linux, 11
macOS, 8
Windows, 5
loading extension, 276, 324
shapefile, 292, 295, 383
spatial joins, 300
Postgres.app, 20
PostgreSQL
backup and restore, 395
pg_dump, 395
pg_restore, 396
command line usage, 363
comparison operators, 36
configuration, 387
creating functions, 346
default postgres database, 19
description of, 12, 19
documentation, 410
functions, 337
GUI tools, 407
importing from other database managers, 60
installation, 1, 3
Linux, 9

PostgreSQL *(continued)*
macOS, 8
Windows, 4
JSON support, 308
maintenance, 387
modules, 237
news and community, 409
`pg_operator` table, 79
postgresql.conf settings file, 393
procedure, 349
recovering unused space, 388
settings, 392
spatial data analysis, 275, 287, 289
starting and stopping, 395
statistics collector, 391
table size, 388
triggers, 337, 354
utilities, tools, and extensions, 408
`version()` function, 14
views, 337
postgresql.conf settings file, 210, 393
editing, 393
location of, 393
reloading settings, 395
precision input for numeric and decimal types, 47
primary key, 18, 31
composite, 121, 124
definition of, 95, 121
natural, 96, 121, 159
surrogate, 122, 126
auto-incrementing, 125
syntax, 99, 123, 124
uniqueness, 96
using auto-incrementing serial type, 46, 122
violation, 124
Prime Meridian, 66, 280
procedural language, 347
procedures
`CALL` command, 352
difference with functions, 349
updating data with, 349
projection (map), 279
Albers, 279
`psql` command line application, 364
connecting to database, 370, 372
displaying table info, 379
editing queries, 375
executing queries from a file, 382
formatting results, 375
help commands, 372
importing and exporting files, 380
meta-commands, 379, 389
multiline queries, 374
`NULL` value display, 105
paging results, 376
parentheses in queries, 374
passing in SQL commands, 381
password file, 373
running queries, 374
saving query output, 381
setup
Linux, 370
macOS, 368
Microsoft Windows, 364
Public Libraries Survey, 140
creating tables, 140–141
Python programming language, xxv
creating PL/Python extension, 352
in PostgreSQL function, 347, 352
installation
macOS, 9
Windows, 5

Q

quantiles, 88
quartiles, 90
query
choosing order of columns, 31
definition of, 17
eliminating duplicate values, 34
execution time, 135
exporting results of, 73
limiting number of rows returned, 67
measuring performance with `EXPLAIN`, 135
order of clauses, 39
retrieving a subset of columns, 31
selecting all rows and columns, 30
quintiles, 90
quotes, single vs. double, 25

R

r (Pearson correlation coefficient), 186

rank() function, 193
ranking data, 193
    rank() and dense_rank() functions, 193
    by subgroup, 195
rates calculations, 196, 227, 235
record_if_grade_changed() user function, 356
REFERENCES keyword declaring foreign key, 128
referential integrity
    cascading deletes, 129
    foreign keys, 128
    primary key, 124
regexp_match() function, 255
    extracting text from result, 260
regexp_matches() function, 256
regexp_replace() function, 252
regexp_split_to_array() function, 252
regexp_split_to_table() function, 252
regular expressions, 247
    capture group, 249, 257
    escaping characters, 255
    examples, 250
    notation, 248
    parsing unstructured data, 253, 258
    regexp_match() function, 255
    regexp_matches() function, 256
    regexp_replace() function, 252
    regexp_split_to_array() function, 252
    regexp_split_to_table() function, 252
    with substring() function , 250
    in WHERE clause, 250
relational database, 93
    related tables, 18
relational model, 93
    reducing redundant data, 97
    table relationships, 105
replace() string function, 247
reserved keywords, 119
RETURNING clause, 170
    with UPDATE statement, 167
right() function, 247
RIGHT JOIN keyword, 101
ROLLBACK statement in transaction block, 177
roots, square and cube, 80
round() function, 86, 188
row
    constructor, 328
    counting, 143
    in CSV file, 60
    definition, 93
    deleting, 175
    generating from JSON array, 332
    inserting, 25
    recovering unused, 388
    specifying in window function, 200
    updating specific, 169
row() function, 328
R programming language, xxv
r-squared (coefficient of determination), 191

S

scalar subquery, 224
scale input for numeric and decimal types, 47
scatterplot, 186, 188
selecting all rows and columns, 30
SELECT statement
    definition, 30
    order of clauses, 39
semicolon (;), 19
server
    localhost, 20
    postgresql.conf file, 210
    setting time zone, 210
SET keyword
    clause in UPDATE statement, 166
    TIME ZONE, 211
set operators, 109
    EXCEPT, 111
    INTERSECT, 111
    UNION, 109
    UNION ALL, 109
setting up your coding environment, 1
shapefile, 291
    contents of, 292
    loading into database, 292
    shp2pgsql command line utility, 383
    US Census TIGER/Line, 292, 300

SHOW command
config_file, 393
data_directory, 395
timezone, 209
view all server settings with, 210
shp2pgsql command line utility, 383
simple feature standard, 277
slope-intercept formula, 190
snake case, 118
sorting data, 32
on aggregate results, 149
by multiple columns, 33
spatial data
area analysis, 296
building blocks of, 276
converting JSON to, 323
data types, 280
distance analysis, 287, 289
finding location, 297
geographic coordinate system, 276, 279
geometries, 277, 281, 283, 284, 319
intersection analysis, 301
joins, 300
projection, 279
shapefile, 291
simple feature standard, 277
spatial reference system identifier (SRID), 277, 279
well-known text (WKT), 278
WGS 84 coordinate system, 280
spatial reference system identifier (SRID), 277, 279
setting with ST_SetSRID(), 286
SQL
history of, xxiv
indenting code, 27
math operators, 78
relational model, 93
reserved keywords, 119
style conventions, 23, 27, 56
using with external programming languages, xxvi
value of using, xxiv, xxv
variations among databases, xxiv
sqrt() function, 81
square root operator (|/), 81
SRID (spatial reference system identifier), 277, 279
setting with ST_SetSRID(), 286
START TRANSACTION statement, 177
statistical functions, 183
correlation with corr(), 186
dependent and independent variables, 186
linear regression, 189–190
regr_intercept() function, 190
regr_slope() function, 190
rates calculations, 196
rolling average, 198
standard deviation, 192
variance, 192
string functions, 246
case formatting, 246
character information, 246
char_length(), 246
extracting and replacing characters, 247
initcap(), 246
left(), 247
length(), 246
lower(), 246
position(), 246
removing characters, 247
replace(), 247
right(), 247
to_char(), 219
trim(), 247
upper(), 246
subquery
correlated, 224
with crosstab() function, 239
definition, 223
in DELETE statement, 225
expressions, 229
generating column with, 228
IN operator expression, 230
with LATERAL, 231
scalar, 224
uncorrelated, 224
in WHERE clause, 224
substring() function, 250
subtracting numbers, 79
across columns, 82

sum() function, 86
example on joined tables, 149
grouping by column value, 152
summarizing data, 139
surrogate primary key, 122
creating, 126

T

tab character as delimiter, 63
table
add column, 165, 168
aliases, 107, 226
alter column, 165
autovacuum, 390
constraints, 23
creating, 22
dead tuples, 388
definition of, 17
deleting column, 165, 176
deleting data, 175
derived table, 225
design best practices, 117
dropping, 176
holds data on one entity, 93
indexes, 133
inserting rows, 25
key columns, 94
modifying with ALTER statement, 164
naming, 120
querying multiple tables using joins, 97
relationships, 17
size, 388
temporary tables, 71
viewing data, 26
tablefunc module, installing for crosstab(), 237
table relationships
many to many, 105
one to many, 105, 128
one to one, 105
TABLE statement, 31
telling your data's story, 399
asking why, 405
assessing the data's origins, 402
building your own database, 401
communicating your findings, 405
consulting the data's owner, 403
documenting your process, 400
gathering your data, 400
identifying trends over time, 403
interviewing the data with queries, 402
starting with a question, 400
temporary table
declaring, 71
removing with DROP TABLE, 71
text
case formatting, 246
concatenation, 172
editors, 2
escaping characters, 255
extracting and replacing characters, 247
formatting with functions, 246
matching patterns with regular expressions, 247
removing characters, 247
text data type, 42
text editors, 2
TEXT file format, 63
text qualifier
ignoring delimiters with, 61
specifying with QUOTE option in COPY, 63
tilde-asterisk case-insensitive matching operator (~*), 250
tilde case-sensitive matching operator (~), 250
time, matching with regular expression, 249
time data types
interval, 51
time, 51
timestamp, 51, 204
timestamp, 51, 204
calculations with, 212
creating from components, 207, 261
extracting components from, 205
formatting display, 219
retrieving current date and time, 208
subtracting to find interval, 219

timestamp *(continued)*
with time zone, 51
within transaction, 208
time zones
AT TIME ZONE keywords, 212
automatic conversion of, 205
including in timestamp, 51, 204, 262
setting, 210
setting server default, 394
standard name database, 52
viewing names of, 210
viewing server setting, 209
working with, 209
to_char() function, 219
to_json() function, 327
to_timestamp() function, 320
to_tsquery() function, 266
to_tsvector() function, 265
transaction block
COMMIT, 177
definition of, 177
ROLLBACK, 177
START TRANSACTION, 177
with time functions, 208
visibility to other users, 179
triggers, 337, 354
BEFORE INSERT statement, 359
CREATE TRIGGER statement, 356
EXECUTE PROCEDURE statement, 357
FOR EACH ROW statement, 356
FOR EACH STATEMENT statement, 356
NEW and OLD variables, 356
testing, 357, 360
trim_county() user function, 353
trim() function, 247
true as Boolean value, 94
TRUNCATE statement, 176
restarting identity sequence with, 176
ts_headline() function, 269
tsquery data type, 266
ts_rank() function, 272
ts_rank_cd() function, 272
tsvector data type, 265

U

uncorrelated subquery, 224
underscore wildcard for pattern matching (_), 37
UNIQUE constraint, 96, 131
universally unique identifier (UUID), 55, 122
as primary key, 122
unnest() function, 91
unstructured data, 245
parsing with regular expressions, 253, 258
update_personal_days() user function, 350
UPDATE statement
with CASE, 262
definition, 166
PostgreSQL-specific syntax, 167
with RETURNING clause, 167, 170, 262
with SET clause, 166
using across tables, 166, 173
upper() function, 246
USA Today, xxiii
US Census
American Community Survey, 2014–2018, 184
description of columns, 184
importing data, 184
County Business Patterns, 197
county population estimates, 63, 295
adding and subtracting columns, 82
description of columns, 65
finding average county population, 87
finding median county population, 89
importing data, 64
percent calculations with, 84
performing calculations across tables, 112
methodologies compared, 185, 402
trade data, 200
US Department of Agriculture, 158, 402
farmers' market data, 285
US Geological Survey, 318
UTC (Coordinated Universal Time), 52, 206
UTC offset, 52, 206, 219

UUID (universally unique identifier), 55, 122
  as primary key, 122

V

VACUUM command, 388
  ANALYZE option, 391
  autovacuum process, 390
  editing server setting, 393
  FULL option, 392
  monitoring table size, 388
  purpose of, 388
  running manually, 392
  time of last vacuum, 391
  VERBOSE option, 392
VALUES clause with INSERT, 25
varchar, 23, 42
  also specified as character varying, 42
version() function, 14
views, 337
  advantage of using, 338
  creating, 338
  deleting data with, 346
  dropping, 339
  inserting data with, 342, 344
  LOCAL CHECK OPTION, 344
  materialized, 338
  queries in, 339
  retrieving specific columns, 341
  security_barrier option, 343
  updating data with, 342, 345

W

Wall Street Journal, xxiv
well-known text (WKT), 278
  extended, 283
  order of coordinates, 278
WHEN clause in CASE statement, 241
WHERE clause, 35
  in COPY statement, 70
  with DELETE statement, 175
  with EXISTS operator, 167
  with IS NULL keywords, 161
  with JSON operators, 316
  with LIKE expression, 171
  with regular expressions, 250
  in UPDATE statement, 166
whole numbers, 45
wildcard
  percent sign (%), 37
  underscore (_), 37
window functions
  definition of, 193
  OVER clause, 193, 220
  PARTITION BY clause, 195
  specifying rows, 200
WITH keyword
  defining common table expression, 234
  options with COPY, 44, 62
WKT (well-known text), 278
  extended, 283
  order of coordinates, 278
working tables, 176

X

XML, 55

Z

ZIP codes, 163
  loss of leading zeros, 163