Creating Database Tables

Jon Flanders

@jonflanders www.jonflanders.com

DATA DEFINITION LANGUAGE (DDL)

SQL subset for creating databases and tables

Most tools have a visual method

Good to have an idea of what they are doing

CREATE DATABASE

Oddly not part of the SQL Standard

Is supported by most implementations

USE DATABASE to scope future queries

Can also fully qualify table name to database

```
CREATE DATABASE Contact;
```

- CREATE DATABASE COMMAND

```
USE DATABASE Contact;
SELECT * FROM person p;
```

- USE DATABASE COMMAND
- ALL QUERIES WILL BE IN THIS DATABASE

```
SELECT * FROM
Contact.person p;
```

- FULLY QUALIFIED TABLE NAME

CREATE TABLE

Is part of SQL Standard

Followed by table name

Then list of column definitions

At minimum column name and type

```
CREATE TABLE
email_address
(
email_address_id
INTEGER,
email_address_person_id
INTEGER,
email_address
VARCHAR(55)
);
```

- CREATE TABLE
- TABLE NAME

- COLUMNS

Standard SQL Data Types

Data Type	Value Space
CHARACTER	Can hold N character values – set to N statically
CHARACTER VARYING	Can hold N character values – set to N dynamically – storage can be less than N
BINARY	Hexadecimal data
SMALLINT	-2^15 (-32,768) to 2^15-1 (32,767)
INTEGER	-2^31 (-2,147,483,648) to 2^31-1 (2,147,483,647)
BIGINT	-2^63 (-9,223,372,036,854,775,808) to 2^63-1 (9,223,372,036,854,775,807)
BOOLEAN	TRUE or FALSE
DATE	YEAR, MONTH, and DAY in the format YYYY-MM-DD
TIME	HOUR, MINUTE, and SECOND in the format HH:MM:SS[.sF] where F is the fractional part of the
TIMESTAMP	Both DATE and TIME

NULL VALUES

NULL is a special value in SQL

Indicates a lack of a value

Columns can be required or not required

Required is NOT NULL

Not required is NULL

NULL or NOT NULL

NULL

Default for a column definition

Inserting NULL value ok

NOT NULL

Must be specified on column definition

Inserting NULL value is an error

```
CREATE TABLE
email_address
(
email_address_id
INTEGER NOT NULL,
email_address_person_id
INTEGER,
email_address
VARCHAR(55) NOT NULL);
```

- CREATE TABLE COMMAND
- NOT NULL
- NULL
- NOT NULL

PRIMARY KEY

Must be a unique value per row

Must be NOT NULL

Can be a multiple columns (compound key)

```
CREATE TABLE
email_address
(
email_address_id
INTEGER PRIMARY KEY,
email_address_person_id
INTEGER,
email_address
VARCHAR(55) NOT NULL
);
```

- CREATE TABLE
- TABLE NAME

- PRIMARY KEY

- COLUMNS

CONSTRAINT

Way to add keys in one grouping

Primary or foreign keys

```
CREATE TABLE phone_number
(
phone_number_id
INTEGER NOT NULL,
phone_number_person_id
INTEGER NOT  NULL,
phone_number
VARCHAR(55) NOT NULL,
CONSTRAINT
PK_phone_number
PRIMARY KEY
(phone_number_id)
);
```

◀ CREATE TABLE COMMAND

◀ CONSTRAINT

ALTER TABLE

Used to change an existing table

Add/remove column

Change column data type

Change column constraints

Must comport with current data

```
ALTER TABLE
email_address
ADD CONSTRAINT
FK_email_address_person
FOREIGN KEY
(email_address_person_id)
REFERENCES
person
(person_id);
```

- ALTER TABLE
- TABLE NAME
- ADD KEYWORD

- FOREIGN KEY

DROP TABLE

Removes table and all data from database

BE CAREFUL!

Error if table is a foreign key to another table

```
DROP TABLE person;
```

DROP TABLE COMMAND

Summary

Understanding DDL is a good foundation for working with SQL, even if you use it rarely

CREATE TABLE is the command to configure your columns and relations

ALTER TABLE lets you change existing definitions

DROP TABLE removes the table and all its rows from the database