Learning SQL: Chapter 1

Jimmy Nguyen

Francis I. Proctor Foundation

hieu.nguyen@ucsf.edu

September 4, 2018

Overview

Relational model

Big Idea

Redundant data is used to link records in different tables

Typically, redundant data are some kind of identifier

- Patient ID
- OD or OS
- Study site number

Example of relational database

Customer

cust_id	fname	lname
1	George	Blake
2	Sue	Smith

Account

account_id	product_cd	cust_id	balance
103	CHK	1	$75.00
104	SAV	1	$250.00
105	CHK	2	$783.64
106	MM	2	$500.00
107	LOC	2	0

Product

product_cd	name
CHK	Checking
SAV	Savings
MM	Money market
LOC	Line of credit

Transaction

txn_id	txn_type_cd	account_id	amount	date
978	DBT	103	$100.00	2004-01-22
979	CDT	103	$25.00	2004-02-05
980	DBT	104	$250.00	2004-03-09
981	DBT	105	$1000.00	2004-03-25
982	CDT	105	$138.50	2004-04-02
983	CDT	105	$77.86	2004-04-04
984	DBT	106	$500.00	2004-03-27

Figure: Notice redundant columns linking each table...

"Key" Terms ;)

primary key column(s) of unique identifiers for rows

foreign key column(s) together that identify a single row in another table

result set a nonpersistent table, generally an SQL query result

Important: Primary keys should never be allowed to change!

Redundant information typically just identifiers

Normalization

Each independent piece of information is in only one place

Example: a patient's address changes

- Ideally, we change address entry for patient with `patient_id` 105 in the table **addresses**
- No other table should contain addresses
- Idea: carefully design database to avoid unnecessary redundancies and ensure maintainability

So what is SQL?

- Pronounced S-Q-L or *sequel*
- Used to manipulating data in relational databases
- Went from DSL/Alpha -> SQUARE -> SEQUEL -> SQL
- Apparently, SQL doesn't actually stand for anything
  - People insist on "Structured Query Language"
- SQL has ANSI standards which are periodically updated (currently SQL:2016)

SQL is non-procedural

- You define the results through queries, and queries are sent to an *optimizer* that decides how to execute the query
- SQL cannot create complete applications, and is typically integrated into other languages
- Database administrators can influence optimizer behavior through tweaking *optimizer hints*, but most end-users shouldn't care too much about this

SQL Statement Classes

schema statements define data structures stored in database

data statements manipulate data structures

transaction statements begin, end, and roll back transactions

Note: transactions statements will be covered in Ch 12; book focuses on data statements

Example: SQL Schema Statement


```
CREATE TABLE corporation
 (corp_id SMALLINT,
  name VARCHAR(30),
  CONSTRAINT pk_corporation PRIMARY KEY (corp_id)
 );
```

Figure: Create table with two columns, **corp_id** and **name**

Example: SQL Data Statement


```
INSERT INTO corporation (corp_id, name)
VALUES (27, 'Acme Paper Corporation');
```

Figure: Insert row into **corporation** table with **corp_id**=**27** and **name**=**Acme Paper Corporation**

Example: SQL `select` Statement


```
mysql< SELECT name
    -> FROM corporation
    -> WHERE corp_id= 27;
+-----------------------+
| name                  |
+-----------------------+
| Acme Paper Corporation |
+-----------------------+
```

Figure: Retrieve the name in the table **corporation** with **corp_id** 27

Basic SQL query structure


```
SELECT /* one or more things */ ...
FROM /* one or more places */ ...
WHERE /* one or more conditions apply */ ...
```

Figure: Most SQL queries will have at least three clauses

Data dictionaries

- Database elements created via SQL schema statements are stored in special tables called *data dictionaries*
- Data dictionaries can be `selected` and manipulated
- Application: can query data dictionary to determine current set of columns and dynamically generate report each time it is executed

So what is MySQL?

Fun facts:

- Many SQL versions exist with various quirks, but most databases should comply with ANSI SQL standards
- MySQL is an open-source implementation of SQL
- Statements presented in book should work with most databases

Free Tools:

- Server: MySQL Community
- Visual GUI: MySQL Workbench
- Can also work with MySQL interactively through your favorite terminal

Learning SQL Example Database Script

A script that generates the data used in the book can be found at `http://examples.oreilly.com/9780596520847/`