CS 50011: Introduction to Systems II

Prof. Jeff Turkstra

Computer Science Department
Purdue University

Copyright 2017

SQL
and Relational Databases

Relational Databases

- You have learned how to store data persistently into files.
- However, when using plain files, you have to write functions to parse the information, to search it etc.
- Relational databases (often simply “databases”) store the information in tables.
- A table is made of many rows and columns, where one row stores a record and a column stores an attribute of the record.
- Databases provide functions to add, delete, and search records in a table.
- SQL (Structured Query Language) is the standard language used to manipulate databases.

Relational Databases

- Predominant database idea since 1980s
- Organizes data into sets of two-dimensional tables
- Edgar “Ted” Codd (1923-2003)
- Introduced concept c1970
- Turing Award 1981

Available SQL databases

MariaDB

https://mariadb.org/

Very popular free database

Oracle

http://www.oracle.com

Very popular database for enterprise use

Microsoft SQL

http://www.microsoft.com/sqlserver

Also very popular.

Structured Query Language (SQL)

All major languages let you communicate with a database using SQL.

There are often GUIs available that help with database administration

You can think of a database as a group of named tables with rows and columns.

Each column has a name and each row contains a set of related data.

Anatomy

- Database consists of a set of named *relations* (tables)
- Relation
  - Set of *tuples* (rows) with identical column structure
  - Values often uniform “type” (integer, text, date, etc)
- All rows should be distinct
  - Not necessarily enforced

Single Table

Title	ISBN	FName	LName	Price	Publisher	URL
A Guide to the SQL Standard	0-201-96426-0	Alexander	Christopher	47.95	Addison-Wesley	www.aw-bc.com
A Pattern Language: Towns, Buildings, Construction	0-19-501919-9	Frederick P.	Brooks	65.00	John Wiley & Sons	www.wiley.com
A Pattern Language: Towns, Buildings, Construction	0-19-501919-9	Smith	Peter	65.00	John Wiley & Sons	www.wiley.com

Structuring your information in Tables

- As part of the design of a project, you need to design the tables in a database.
- You may store your information in multiple tables to reduce the amount of repeated information.
- In the example before we could store, all the information in a single table but that would lead to many repeated entries.

Schema vs. Instance

- Structure of the database (tables, columns, types, etc) is the *schema*
  - Usually fixed over time
- Instance is a specific collection of data conforming to the schema
  - “Database” usually refers to the latter (“the grades database”)

SQL Schema

Authors Table

Author_ID	Name	Fname
ALEX	Alexander	Christopher
BROO	Brooks	Frederick P.
SMITH	Smith	Peter

Books Table

Title	ISBN	Publisher_ID	Price
A Guide to the SQL Standard	0-201-96426-0	0201	47.95
A Pattern Language: Towns, Buildings, Construction	0-19-501919-9	0407	65.00

SQL by Example (from textbook)

BookAuthors Table

ISBN	Author_ID	Seq_No
0-201-96426-0	ALEX	1
0-201-96426-0	BROO	2
0-19-501919-9	DAR	1

Publishers Table

Publisher_ID	Name	URL
0201	Addison-Wesley	www.aw-bc.com
0407	John Wiley & Sons	www.wiley.com

Primary Keys

- One or more of the columns of the table are defined as “Primary Keys”.
- The value in each column of a primary key must be unique.
- A separate data structure called “B-tree” is created for this “Primary Key” for quick access.
- Searches on this binary key will be at least O(log n), compared to searches in other columns that will be linear.

Foreign Keys

- Uniquely identifies row of another table
  - Refers to primary key in another table
- Must be NULL or refer to a row in the other table
  - Referential integrity constraint
- Database normalization
  - Organizing columns and tables to reduce data redundancy

Structured Query Language (SQL)

- Computer language for expressing management, manipulation, and querying of databases
- Supported by almost all RDBs
- Sometimes there are differences
  - Underlying implementation may also be inconsistent

SQL by Example

By convention SQL keywords are written in uppercase.

SELECT * FROM Books

This query returns all rows in the Books table.

SQL statements always require FROM

SELECT ISBN, Price, Title

FROM Books

This query returns a table with only the ISBN, price and title columns from the Books table.

SQL by Example

SELECT ISBN, Price, Title

FROM Books

WHERE Price <=29.95

This query returns a table with the ISBN, Price and Title from the Books table but only for the books where the price is less or equal to 29.95.

SQL by Example

SELECT ISBN, Price, Title

FROM Books

WHERE Title NOT LIKE “%n_x%”

Returns a table with ISBN, Price and Title as columns excluding the books that contain Linux or UNIX in the title.

The “%” character means any zero or more characters. “_” means any single character.

SQL by Example

```
SELECT  Title, Name, URL
   FROM Books, Publishers
   WHERE Books.Publisher_ID=Publishers.Publisher_ID
```

It returns a table with the Title, Name of publisher, and URL from Books and Publishers.

Title	Name	URL
A Guide to the SQL Standard	Addison-Wesley	www.aw-bc.com
A Pattern Language: Towns, Buildings, Construction	John Wiley & Sons	www.wiley.com

SQL by Example

You can also use SQL to change data inside a database.

```
UPDATE Books
   SET Price = Price – 5.00
   WHERE Title Like “%C++%”
```

This reduces the price by $5.00 for all books that have C++ in the title.

SQL by Example (from textbook)

You can also delete rows with SQL

DELETE FROM Books

WHERE Title Like “%C++%”

This deletes all books that have C++ in the title.

SQL by Example

Use INSERT to insert a new row in the table.

INSERT INTO Books

VALUES (‘A Guide to the SQL Standard’, ‘0-201-96426-0’, ‘0201’, 47.95)

This inserts a new book in the Books table.

SQL by Example

You can also create a new table using SQL

```
CREATE TABLE Books
   (
      TITLE CHAR(60),
      ISBN CHAR(13),
      Publisher_ID CHAR(6),
      Price DECIMAL(10,2)
)
```

JOIN

- Combine columns from one or more tables using values common to each
- Different types
  - INNER JOIN common
  - LEFT (OUTER) JOIN
    - Always contain left table rows, even if no match on right
  - RIGHT JOIN
  - FULL JOIN

SQL Tutorials

For more information about SQL, see the SQL tutorial in

http://www.w3schools.com/sql/default.asp

You can also run some SQL examples there.

Running SQL in the SSLAB machines

- The explanation of how to run mysql inside your account in the sslab machines is in :

  http://support.cs.purdue.edu/help/MySQL_mini-HOWTO_Linux

- The instructions allow you to run the database in the background but it will allow only to connect clients in the same machine.
- Here is the mysql tutorial. Follow the tutorial to get familiar with mysql.

  http://dev.mysql.com/doc/refman/5.1/en/tutorial.html

Running mysql in the sslab machines

To run mysql type:

```
bash> mysql -u root
mysql> CREATE DATABASE menagerie;
mysql> USE menagerie
mysql> SHOW TABLES;
mysql> CREATE TABLE pet (name VARCHAR(20), owner VARCHAR(20),
    -> species VARCHAR(20), sex CHAR(1), birth DATE, death DATE);
mysql> SHOW TABLES;
mysql> DESCRIBE pet;
mysql> INSERT INTO pet
    -> VALUES ('Puffball','Diane','hamster','f','1999-03-30',NULL);
mysql> select * from pet;
```