CS 2541W
Database Systems & Team Projects
Laboratory

Intro to mySQL

By Roxana Leontie
The new report form is great, but now I'm getting too many emails. I need a way to store them and easily find the ones related to Fang.

Owen's PHP form works well. Too well...
MySQL excels at storing data into TABLES

The web server processed web page request, runs PHP scripts, and returns HTML content.

The database server reads and writes data from/to the database.

Data

The database itself is often stored as files on a hard drive, but it doesn’t necessarily have to be.
Connecting to MySQL

Open MobaXterm and start local terminal
Establish a SSH connection

* Use your GW netID to connect to the gwupyterhub.seas.gwu.edu server

```bash
ssh -Y GWnetID@gwupyterhub.seas.gwu.edu
```

* Login into MySQL

```bash
mysql -u GWnetID -p
```

* Reset password

```sql
SET PASSWORD FOR 'GWNetID'@'localhost' = 'NEWPASSWORD';
```

**Note:** use your GW NetID, BUT the password CSCI2541_sp19
MySQL database

* An existing database is available for your use

```sql
show databases;
```

![Output of `show databases;` command](image)

* Use your database:

```sql
use DB_name;
```

![Output of `use DB_name;` command](image)
Creating a table:

```sql
create table deposit(
    acctno int,
    custid int,
    balance decimal,
    branch_name varchar(20),
    primary key (acctno),
    foreign key (custid) references customer(custid),
    foreign key (branch_name) references branch(branch_name)
);
```
Creating a table:

```sql
create table deposit(
    acctno int,
    custid int,
    balance decimal,
    branch_name varchar(20),
    primary key (acctno),
    foreign key (custid) references customer(custid),
    foreign key (branch_name) references branch(branch_name)
);
```

Note!

Tables customer and branch should to be created before!
### Basic data types

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>char</em>(n)</td>
<td>Fixed length character string of length n (max 255)</td>
</tr>
<tr>
<td><em>varchar</em>(n)</td>
<td>Variable length character string (max 255)</td>
</tr>
<tr>
<td><em>date</em></td>
<td>Holds a date field (28-Jan-2013)</td>
</tr>
<tr>
<td><em>decimal</em>(n,d)</td>
<td>Real numbers occupying up to n spaces with d digits after the decimal point</td>
</tr>
<tr>
<td><em>int</em>(n)</td>
<td>Integer with up to n digits</td>
</tr>
</tbody>
</table>
Constraints

* PRIMARY KEY
* FOREIGN KEY
Create Table Example

```sql
mysql> create table branch(
    -> branch_name varchar(20),
    -> assets decimal,
    -> branch_city varchar(20),
    -> primary key (branch_name)
);
Query OK, 0 rows affected (0.14 sec)
```

```sql
show tables;
```

```sql
mysql> show tables;
+--------------------------+
| Tables_in_realontie     |
|--------------------------+
| branch                  |
+--------------------------+
1 row in set (0.00 sec)
```
Tables

* **View table structure:**
  describe branch;

* **Modifying a table**
  ALTER TABLE deposit ADD(deposit_date date);
  ALTER TABLE deposit DROP column deposit_date;
  ALTER TABLE customer MODIFY COLUMN name varchar(25);

* **To delete a table:**
  DROP TABLE deposit;
Populate tables: INSERT

The SQL keywords **INSERT** and **INTO** begin the statement.

**The name of the table... in Owen's case, it will be aliens_abduction.**

The next part is a list of your table column names, separated by commas.

The single quotes are correct. Use them whenever you're inserting text, even if it's a single character!

More column names follow, with no comma after the last one.

**IMPORTANT:** these need to be in the same order as the column names.

Another SQL keyword, this one signaling that the values for the columns follow.

**VALUES**

This next part is a list of the values to be inserted, separated by commas.

More quoted values follow, with no comma after the last one.
describe table_name;

```sql
mysql> describe branch;
+-----------------+-----------------+-------+--------+----------+---------+
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Key</th>
<th>Default</th>
<th>Extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>branch_name</td>
<td>varchar(20)</td>
<td>NO</td>
<td>PRI</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>assets</td>
<td>decimal(10,0)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>branch_city</td>
<td>varchar(20)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
</tbody>
</table>
+-----------------+-----------------+-------+--------+----------+---------+
3 rows in set (0.00 sec)

mysql> insert into branch (branch_name, assets, branch_city) values ('GWU',40000,'DC');
Query OK, 1 row affected (0.09 sec)
mysql>
```
Confirm data was added: SELECT

Follow SELECT with a list of the columns you want data for.

```
SELECT columns FROM table_name
```

A SELECT always takes place with respect to a specific table, not a database in general.

The FROM part of a SELECT statement is how SELECT knows what table we'll be selecting data from.

The asterisk, or "star", tells the SELECT statement to get the data for all the columns in the table.

```
SELECT * FROM aliens_abduction
```

No list of columns is necessary because * means "get them all!"
Confirm data was added: SELECT

```
mysql> SELECT branch_name, assets, branch_city FROM branch;
+----------------+-------+-------------+
| branch_name    | assets | branch_city |
+----------------+-------+-------------+
| GWU            | 400000 | DC          |
+----------------+-------+-------------+
1 row in set (0.00 sec)
```

```
mysql> SELECT * FROM branch;
+----------------+-------+-------------+
| branch_name    | assets | branch_city |
+----------------+-------+-------------+
| GWU            | 400000 | DC          |
+----------------+-------+-------------+
1 row in set (0.00 sec)
```
* DELETE FROM branch WHERE branch_name='GWU';
In-class exercise

* Create all the tables described in the Bank Database Schema
* For each table creation record the create statements and show the description
* Run the script provided to populate the tables
* For the next part record the query and output
  * Write select statements to show the data in each table
  * Show all the loans that were made at branch GWU
  * Delete all the loans from branch Downtown
Submission

https://classroom.github.com/a/UYkuXu3B
References

* www.w3schools.com
* Head first PHP & MySQL book.