MySQL

Not your SQL, understand?
MySQL

- Relational Database Management System
  - RDBMS
- Stores stuff in Tables
- Tables have named columns
- Tables have multiple rows with the same columns for each row
- Tables can be related to each other
Connecting

- AWS VM
- From your command line:

```
$ mysql
```
Databases

• `show databases`;

• Lists all the databases on this server

• `use <database>;`

• Select a database to send commands to
Looking At Tables

• `show tables;`
• Lists all tables in the database
• `describe <tablename>;`
• Print out the column structure of the given table
SQL

- SQL - **Structured Query Language**
- An english like syntax to interact with a databases
- Basic Verbs initiate Commands
  - SELECT
  - INSERT
  - UPDATE
  - DELETE
CREATE TABLE

• Make a new table to hold stuff
• Think about the columns you want to have in your table
• Data Modeling

```
CREATE TABLE `staff` (
    `id` int(11) NOT NULL auto_increment,
    `name` varchar(1024) default NULL,
    `phone` varchar(1024) default NULL,
    `email` varchar(1024) default NULL,
    PRIMARY KEY (`id`) ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```
MySQL Datatypes


• Several ways to hold a string
  • CHAR and VARCHAR
  • Also BLOB and TEXT

• Numbers
  • INT, SMALLINT, BIGINT etc
  • DECIMAL, NUMERIC, FLOAT, DOUBLE, BIT

• Dates & Times
  • DATE, TIME, TIMESTAMP, DATETIME
CRUD

• You’ll hear people mention CRUD in connection with databases

• **Create**

• **Retrieve**

• **Update**

• **Delete**
## SQL

<table>
<thead>
<tr>
<th>CRUD</th>
<th>SQL Verb</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create</td>
<td>INSERT</td>
</tr>
<tr>
<td>Retrieve</td>
<td>SELECT</td>
</tr>
<tr>
<td>Update</td>
<td>UPDATE</td>
</tr>
<tr>
<td>Delete</td>
<td>DELETE</td>
</tr>
</tbody>
</table>
• Getting data out of tables

SELECT <fields> FROM <tables> [WHERE <conditions>];
select

- SQL is case in-sensitive
- These all work the same
- The Asterisk "*' means "All the fields in the tables"
- Can select just specific fields by specifying which ones

```
select * from staff;
SELECT * FROM staff;
Select * From Staff;
```
selecting specific things

• The WHERE clause for a SELECT statement allows us to limit the rows selected from a set of tables

```sql
SELECT * FROM staff WHERE name='Mark';
```
selecting specific things

- Doesn’t have to be an *exact* match – LIKE

- % is our wildcard match character for strings in SQL

```sql
SELECT * FROM staff WHERE name LIKE 'M%';
```
insert

• Adding new rows to a table

• Values must match positions with their field names

• Values must be correct for the datatype of the field

• Strings must be surrounded by single quotes – 'some string'

```
INSERT INTO <table>
(field1, field2, ...) VALUES (value1, value2, ...);
```
mysql> select * from staff;
<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
<th>phone</th>
<th>email</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mark</td>
<td>626-1541</td>
<td><a href="mailto:fischerm@email.arizona.edu">fischerm@email.arizona.edu</a></td>
</tr>
<tr>
<td>2</td>
<td>Margrit</td>
<td>626-1541</td>
<td><a href="mailto:memcinto@email.arizona.edu">memcinto@email.arizona.edu</a></td>
</tr>
<tr>
<td>3</td>
<td>Tracey</td>
<td>626-1541</td>
<td><a href="mailto:thummel@email.arizona.edu">thummel@email.arizona.edu</a></td>
</tr>
<tr>
<td>4</td>
<td>Cindy</td>
<td>626-1541</td>
<td><a href="mailto:ccamp@email.arizona.edu">ccamp@email.arizona.edu</a></td>
</tr>
<tr>
<td>5</td>
<td>Jan</td>
<td>626-1541</td>
<td><a href="mailto:jknight@email.arizona.edu">jknight@email.arizona.edu</a></td>
</tr>
<tr>
<td>6</td>
<td>Danielle</td>
<td>626-1541</td>
<td><a href="mailto:danistil@email.arizona.edu">danistil@email.arizona.edu</a></td>
</tr>
<tr>
<td>7</td>
<td>Michael</td>
<td>626-1541</td>
<td><a href="mailto:martelle@email.arizona.edu">martelle@email.arizona.edu</a></td>
</tr>
</tbody>
</table>

7 rows in set (0.00 sec)

mysql> INSERT INTO staff (name, phone, email) VALUES ('Adam', '621-1541', 'adam@email.arizona.edu');
Query OK, 1 row affected (0.01 sec)

mysql> select * from staff;
<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
<th>phone</th>
<th>email</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mark</td>
<td>626-1541</td>
<td><a href="mailto:fischerm@email.arizona.edu">fischerm@email.arizona.edu</a></td>
</tr>
<tr>
<td>2</td>
<td>Margrit</td>
<td>626-1541</td>
<td><a href="mailto:memcinto@email.arizona.edu">memcinto@email.arizona.edu</a></td>
</tr>
<tr>
<td>3</td>
<td>Tracey</td>
<td>626-1541</td>
<td><a href="mailto:thummel@email.arizona.edu">thummel@email.arizona.edu</a></td>
</tr>
<tr>
<td>4</td>
<td>Cindy</td>
<td>626-1541</td>
<td><a href="mailto:ccamp@email.arizona.edu">ccamp@email.arizona.edu</a></td>
</tr>
<tr>
<td>5</td>
<td>Jan</td>
<td>626-1541</td>
<td><a href="mailto:jknight@email.arizona.edu">jknight@email.arizona.edu</a></td>
</tr>
<tr>
<td>6</td>
<td>Danielle</td>
<td>626-1541</td>
<td><a href="mailto:danistil@email.arizona.edu">danistil@email.arizona.edu</a></td>
</tr>
<tr>
<td>7</td>
<td>Michael</td>
<td>626-1541</td>
<td><a href="mailto:martelle@email.arizona.edu">martelle@email.arizona.edu</a></td>
</tr>
<tr>
<td>8</td>
<td>Adam</td>
<td>621-1541</td>
<td><a href="mailto:adam@email.arizona.edu">adam@email.arizona.edu</a></td>
</tr>
</tbody>
</table>

8 rows in set (0.00 sec)
Why didn’t we specify the id field?

Where does the 8 come from?
AUTO INCREMENT

• When defining a table, you can specify a PRIMARY KEY field be AUTO_INCREMENT

• This does pretty much what it sounds like

• Anytime a new row is inserted into the table, MySQL will automatically assign a new value, incrementing an internal counter

```
mysql> desc staff;
+--------+----------------+--------+----------+------------+------------+-----------------------------+
| Field  | Type           | Null   | Key      | Default    | Extra      |
+--------+----------------+--------+----------+------------+------------+-----------------------------+
| id     | int(11)        | NO     | PRI      | NULL       | auto_increment |
| name   | varchar(1024)  | YES    |          | NULL       |             |
| phone  | varchar(1024)  | YES    |          | NULL       |             |
| email  | varchar(1024)  | YES    |          | NULL       |             |
+--------+----------------+--------+----------+------------+------------+-----------------------------+
4 rows in set (0.00 sec)
```
update

• Change a value for a field or set of fields.

```
UPDATE <table> SET field1=value1, field2=value2
WHERE [conditions];
```

• WATCH OUT!

• If you don’t specify any conditions, you will update EVERY ROW!
update

```
UPDATE staff SET phone='626-TECH' WHERE id=1;
```

```
mysql> select * from staff where name='Mark';
+----+-------+-----------+--------------------------+
<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
<th>phone</th>
<th>email</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mark</td>
<td>626-1541</td>
<td><a href="mailto:fischerm@email.arizona.edu">fischerm@email.arizona.edu</a></td>
</tr>
</tbody>
</table>
+----+-------+-----------+--------------------------+
1 row in set (0.00 sec)

mysql> UPDATE staff SET phone='626-TECH' WHERE id=1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from staff where name='Mark';
+----+-------+-----------+--------------------------+
<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
<th>phone</th>
<th>email</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mark</td>
<td>626-TECH</td>
<td><a href="mailto:fischerm@email.arizona.edu">fischerm@email.arizona.edu</a></td>
</tr>
</tbody>
</table>
+----+-------+-----------+--------------------------+
1 row in set (0.00 sec)

mysql> 
```
delete

• Deletes rows from a table

DELETE FROM <table> WHERE [conditions];

• WATCH OUT!

• If you don’t specify any conditions, you will DELETE EVERY ROW!
DELETE FROM staff WHERE id=8;

mysql> select * from staff;

<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
<th>phone</th>
<th>email</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mark</td>
<td>626-TECH</td>
<td><a href="mailto:fischerm@email.arizona.edu">fischerm@email.arizona.edu</a></td>
</tr>
<tr>
<td>2</td>
<td>Margrit</td>
<td>626-1541</td>
<td><a href="mailto:memcinto@email.arizona.edu">memcinto@email.arizona.edu</a></td>
</tr>
<tr>
<td>3</td>
<td>Tracey</td>
<td>626-1541</td>
<td><a href="mailto:thummel@email.arizona.edu">thummel@email.arizona.edu</a></td>
</tr>
<tr>
<td>4</td>
<td>Cindy</td>
<td>626-1541</td>
<td><a href="mailto:ccamp@email.arizona.edu">ccamp@email.arizona.edu</a></td>
</tr>
<tr>
<td>5</td>
<td>Jan</td>
<td>626-1541</td>
<td><a href="mailto:jknight@email.arizona.edu">jknight@email.arizona.edu</a></td>
</tr>
<tr>
<td>6</td>
<td>Danielle</td>
<td>626-1541</td>
<td><a href="mailto:danistil@email.arizona.edu">danistil@email.arizona.edu</a></td>
</tr>
<tr>
<td>7</td>
<td>Michael</td>
<td>626-1541</td>
<td><a href="mailto:martelle@email.arizona.edu">martelle@email.arizona.edu</a></td>
</tr>
<tr>
<td>8</td>
<td>Adam</td>
<td>621-1541</td>
<td><a href="mailto:adam@email.arizona.edu">adam@email.arizona.edu</a></td>
</tr>
</tbody>
</table>

8 rows in set (0.00 sec)

mysql> DELETE FROM staff WHERE id=8;
Query OK, 1 row affected (0.00 sec)

mysql> select * from staff;

<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
<th>phone</th>
<th>email</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mark</td>
<td>626-TECH</td>
<td><a href="mailto:fischerm@email.arizona.edu">fischerm@email.arizona.edu</a></td>
</tr>
<tr>
<td>2</td>
<td>Margrit</td>
<td>626-1541</td>
<td><a href="mailto:memcinto@email.arizona.edu">memcinto@email.arizona.edu</a></td>
</tr>
<tr>
<td>3</td>
<td>Tracey</td>
<td>626-1541</td>
<td><a href="mailto:thummel@email.arizona.edu">thummel@email.arizona.edu</a></td>
</tr>
<tr>
<td>4</td>
<td>Cindy</td>
<td>626-1541</td>
<td><a href="mailto:ccamp@email.arizona.edu">ccamp@email.arizona.edu</a></td>
</tr>
<tr>
<td>5</td>
<td>Jan</td>
<td>626-1541</td>
<td><a href="mailto:jknight@email.arizona.edu">jknight@email.arizona.edu</a></td>
</tr>
<tr>
<td>6</td>
<td>Danielle</td>
<td>626-1541</td>
<td><a href="mailto:danistil@email.arizona.edu">danistil@email.arizona.edu</a></td>
</tr>
<tr>
<td>7</td>
<td>Michael</td>
<td>626-1541</td>
<td><a href="mailto:martelle@email.arizona.edu">martelle@email.arizona.edu</a></td>
</tr>
</tbody>
</table>

7 rows in set (0.00 sec)

mysql>
Hi, this is your son’s school. We’re having some computer trouble.

Oh, dear – did he break something? In a way –

Did you really name your son Robert? Drop Table Students; --?

Oh, yes. Little Bobby Tables, we call him.

Well, we’ve lost this year’s student records. I hope you’re happy.

And I hope you’ve learned to sanitize your database inputs.
Joins

- The Relational part of RDBMS
Joins

- You can SELECT from multiple tables in a single query

```sql
SELECT games.game_state,
    games.game_id,
    players_games.player_id
FROM players_games
    INNER JOIN games
        ON players_games.game_id = games.game_id;
```
Joins

- When specifying fields to select from multiple tables, you prefix the field name by the table name

- `tablename.fieldname`

```
SELECT
    games.game_state,
    games.game_id,
    players_games.player_id
...
SELECT games.game_state,
games.game_id,
players_games.player_id
FROM players_games INNER JOIN games
ON players_games.game_id = games.game_id;

+-----------------+--------+---------+
<table>
<thead>
<tr>
<th>game_state</th>
<th>game_id</th>
<th>player_id</th>
</tr>
</thead>
<tbody>
<tr>
<td>playing</td>
<td>60</td>
<td>57fd375464bab393065734b8d3e4cf1d</td>
</tr>
<tr>
<td>playing</td>
<td>60</td>
<td>d29c3e8b83e01f68e97458182e3d4039</td>
</tr>
<tr>
<td>ended</td>
<td>61</td>
<td>d29c3e8b83e01f68e97458182e3d4039</td>
</tr>
<tr>
<td>ended</td>
<td>62</td>
<td>d9dd132068c07304cab89c4659c80c0d</td>
</tr>
<tr>
<td>ended</td>
<td>63</td>
<td>32d6df3d5e59f71a927ed5bea1a6c4bb</td>
</tr>
<tr>
<td>ended</td>
<td>63</td>
<td>d9dd132068c07304cab89c4659c80c0d</td>
</tr>
<tr>
<td>open</td>
<td>64</td>
<td>d9dd132068c07304cab89c4659c80c0d</td>
</tr>
</tbody>
</table>
+-----------------+--------+---------+
7 rows in set (0.00 sec)
Lots Of Other Stuff

• Lots of built-in functions
  • ABS, AVG, POW, RAND, SYSDATE, VARIANCE

• Standard Operators
  • + - / * = > etc

• Stored Procedures
  • Write your code directly in the database, then make SQL calls to the functions

• Can store JSON natively now

• Transactions
Great!

Now go do all that from PHP!