

MySQL

Not *your* SQL, understand?

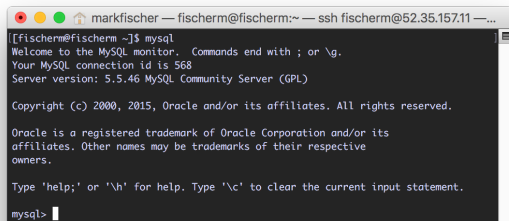
MySQL

- 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
```



```
markfischer — fischer@fischer:~ — ssh fischer@52.35.157.11 — ...
[[fischer@fischer ~]$ mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 568
Server version: 5.5.46 MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

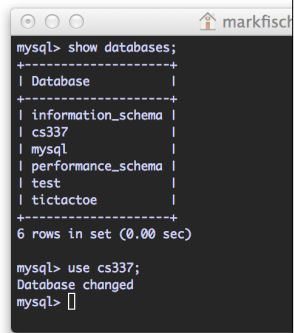
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Databases

- `show databases;`
- Lists all the databases on this server
- `use <database>;`
- Select a database to send commands to

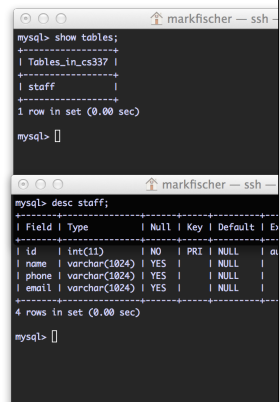


```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| cs337 |
| mysql |
| performance_schema |
| test |
| tictactoe |
+-----+
6 rows in set (0.00 sec)

mysql> use cs337;
Database changed
mysql>
```

Looking At Tables

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



```
mysql> show tables;
+-----+
| Tables_in_cs337 |
+-----+
| staff |
+-----+
1 row in set (0.00 sec)

mysql>

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)

mysql>
```

SQL

- SQL - **S**tructured **Q**uery **L**anguage
- 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

<http://dev.mysql.com/doc/en/data-types.html>

- 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
 - **C**reate
 - **R**etrieve
 - **U**ppdate
 - **D**eflete

SQL

CRUD	SQL Verb
Create	INSERT
Retrieve	SELECT
Update	UPDATE
Delete	DELETE

select

- Getting data out of tables

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

```
mysql> select * from staff;
+----+-----+-----+-----+
| id | name  | phone | email |
+----+-----+-----+-----+
| 1  | Mark  | 626-1541 | fischer@email.arizona.edu |
| 2  | Margrit | 626-1541 | memcinto@email.arizona.edu |
| 3  | Tracey | 626-1541 | thummel@email.arizona.edu |
| 4  | Cindy  | 626-1541 | ccamp@email.arizona.edu |
| 5  | Jan    | 626-1541 | jknight@email.arizona.edu |
| 6  | Danielle | 626-1541 | danistil@email.arizona.edu |
| 7  | Michael | 626-1541 | martelle@email.arizona.edu |
+----+-----+-----+-----+
7 rows in set (0.00 sec)

mysql>
```

select

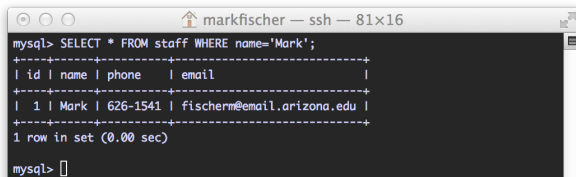
- SQL is case in-sensitive
- These all work the same
- The Asterisk '*' means "All the fields in the tables"
- Can select just specific fields by specifying which ones

```
mysql> select name, email from staff;
+-----+-----+
| name  | email |
+-----+-----+
| Mark  | fischer@email.arizona.edu |
| Margrit | memcinto@email.arizona.edu |
| Tracey | thummel@email.arizona.edu |
| Cindy  | ccamp@email.arizona.edu |
+-----+-----+
```

selecting specific things

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

```
SELECT * FROM staff WHERE name='Mark';
```



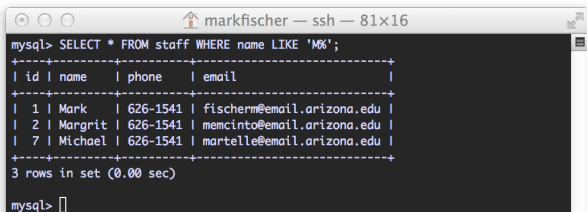
```
mysql> SELECT * FROM staff WHERE name='Mark';
+----+-----+-----+-----+
| id | name | phone | email |
+----+-----+-----+-----+
| 1  | Mark | 626-1541 | fischer@email.arizona.edu |
+----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

selecting specific things

- Doesn't have to be an *exact* match – LIKE
- % is our wildcard match character for strings in SQL

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



```
mysql> SELECT * FROM staff WHERE name LIKE 'M%';
+----+-----+-----+-----+
| id | name | phone | email |
+----+-----+-----+-----+
| 1  | Mark | 626-1541 | fischer@email.arizona.edu |
| 2  | Margrit | 626-1541 | memcinto@email.arizona.edu |
| 7  | Michael | 626-1541 | martelle@email.arizona.edu |
+----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

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;
+----+-----+-----+-----+
| id | name  | phone | email |
+----+-----+-----+-----+
| 1  | Mark  | 626-1541 | fischer@email.arizona.edu |
| 2  | Margrit | 626-1541 | mncinto@email.arizona.edu |
| 3  | Tracey | 626-1541 | thummel@email.arizona.edu |
| 4  | Cindy | 626-1541 | ccamp@email.arizona.edu |
| 5  | Jan   | 626-1541 | jknight@email.arizona.edu |
| 6  | Danielle | 626-1541 | danistill@email.arizona.edu |
| 7  | Michael | 626-1541 | martelle@email.arizona.edu |
+----+-----+-----+-----+
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;
+----+-----+-----+-----+
| id | name  | phone | email |
+----+-----+-----+-----+
| 1  | Mark  | 626-1541 | fischer@email.arizona.edu |
| 2  | Margrit | 626-1541 | mncinto@email.arizona.edu |
| 3  | Tracey | 626-1541 | thummel@email.arizona.edu |
| 4  | Cindy | 626-1541 | ccamp@email.arizona.edu |
| 5  | Jan   | 626-1541 | jknight@email.arizona.edu |
| 6  | Danielle | 626-1541 | danistill@email.arizona.edu |
| 7  | Michael | 626-1541 | martelle@email.arizona.edu |
| 8  | Adam  | 621-1541 | adam@email.arizona.edu |
+----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

insert

```
INSERT INTO staff
(name, phone, email) VALUES
('Adam', '621-1541', 'adam@email.arizona.edu');
```

- Why didn't we specify the `id` field?
- Where does the **8** come from?

```
mysql> select * from staff;
+----+-----+-----+-----+
| id | name  | phone | email |
+----+-----+-----+-----+
| 1  | Mark  | 626-1541 | fischer@email.arizona.edu |
| 2  | Margrit | 626-1541 | mncinto@email.arizona.edu |
| 3  | Tracey | 626-1541 | thummel@email.arizona.edu |
| 4  | Cindy | 626-1541 | ccamp@email.arizona.edu |
| 5  | Jan   | 626-1541 | jknight@email.arizona.edu |
| 6  | Danielle | 626-1541 | danistill@email.arizona.edu |
| 7  | Michael | 626-1541 | martelle@email.arizona.edu |
| 8  | Adam  | 621-1541 | adam@email.arizona.edu |
+----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

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

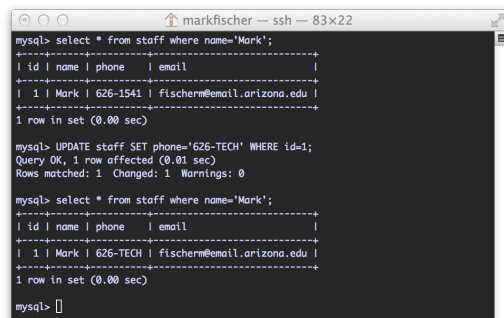
- Change a value for a field or set of fields.

```
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;
```



A terminal window titled 'markfischer — ssh — 83x22' showing MySQL commands and output. The first query is 'select * from staff where name='Mark';' which returns a table with columns id, name, phone, and email, containing one row: id=1, name=Mark, phone=626-1541, email=fischer@email.arizona.edu. The second query is 'UPDATE staff SET phone='626-TECH' WHERE id=1;', which returns 'Query OK, 1 row affected (0.01 sec) Rows matched: 1 Changed: 1 Warnings: 0'. The third query is 'select * from staff where name='Mark';' which returns the same table as before, but with phone=626-TECH.

```
mysql> select * from staff where name='Mark';  
+----+-----+-----+-----+  
| id | name | phone | email |  
+----+-----+-----+-----+  
| 1 | Mark | 626-1541 | fischer@email.arizona.edu |  
+----+-----+-----+-----+  
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';  
+----+-----+-----+-----+  
| id | name | phone | email |  
+----+-----+-----+-----+  
| 1 | Mark | 626-TECH | fischer@email.arizona.edu |  
+----+-----+-----+-----+  
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;
+----+-----+-----+-----+
| id | name  | phone | email |
+----+-----+-----+-----+
| 1 | Mark  | 626-TECH | fischer@email.arizona.edu |
| 2 | Margrit | 626-1541 | memcinto@email.arizona.edu |
| 3 | Tracey | 626-1541 | thumme@email.arizona.edu |
| 4 | Cindy | 626-1541 | ccomp@email.arizona.edu |
| 5 | Jan   | 626-1541 | jknights@email.arizona.edu |
| 6 | Danielle | 626-1541 | donistill@email.arizona.edu |
| 7 | Michael | 626-1541 | martelle@email.arizona.edu |
| 8 | Adam  | 621-1541 | adam@email.arizona.edu |
+----+-----+-----+-----+
8 rows in set (0.00 sec)

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

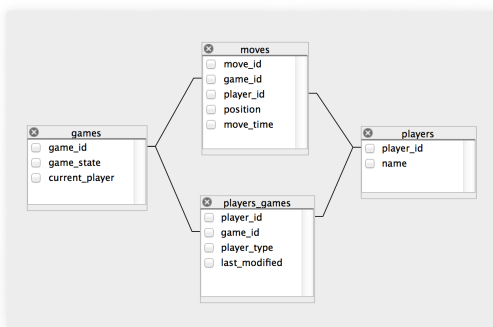
mysql> select * from staff;
+----+-----+-----+-----+
| id | name  | phone | email |
+----+-----+-----+-----+
| 1 | Mark  | 626-TECH | fischer@email.arizona.edu |
| 2 | Margrit | 626-1541 | memcinto@email.arizona.edu |
| 3 | Tracey | 626-1541 | thumme@email.arizona.edu |
| 4 | Cindy | 626-1541 | ccomp@email.arizona.edu |
| 5 | Jan   | 626-1541 | jknights@email.arizona.edu |
| 6 | Danielle | 626-1541 | donistill@email.arizona.edu |
| 7 | Michael | 626-1541 | martelle@email.arizona.edu |
+----+-----+-----+-----+
7 rows in set (0.00 sec)

mysql>
```



Joins

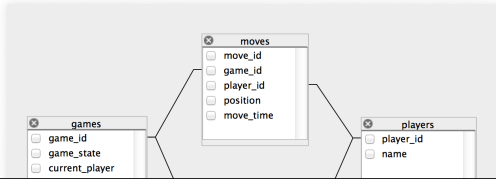
- The Relational part of RDBMS



Joins

- You can SELECT from multiple tables in a single query

```
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  
  ...
```

Table Name

Field Name

```
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;
```

```
mysql> 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;  
+-----+-----+-----+  
| game_state | game_id | player_id |  
+-----+-----+-----+  
| playing   | 60      | 57fd375464bab393065734b8d3e4cf1d |  
| playing   | 60      | d29c3e8b83e01f68e97458182e3d4039 |  
| ended     | 61      | d29c3e8b83e01f68e97458182e3d4039 |  
| ended     | 62      | d9dd132068c07304cab89c4659c80c0d |  
| ended     | 63      | 32d6df3d5e59f71a927ed5bea1a6c4bb |  
| ended     | 63      | d9dd132068c07304cab89c4659c80c0d |  
| open      | 64      | d9dd132068c07304cab89c4659c80c0d |  
+-----+-----+-----+  
7 rows in set (0.00 sec)  
  
mysql>
```

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!