Introduction to MySQL

Dr. Justin F. Brunelle
jbrunelle@cs.odu.edu
http://www.cs.odu.edu/~jbrunelle/cs518

Relational Databases

- Data organized in tables
  - Can be created, retrieved, deleted, and updated
  - Uses keys for manipulation
- Primary key - unique identifier for the table
- Foreign key - matches primary key of another table

First Normal Form

- No repeating columns containing same data
- All columns contain a single value
- Primary key to uniquely identify each row

name	real_name	power 1	power 2	power 3	lair_address	city	state	zip
Clean Freak	John Smith	Strength	X-ray vision	Flight Invisibility	123 Poplar Ave	Townsburg	OH	45293
Soap Stud	Efram Jones	Speed			123 Poplar Ave	Townsburg	OH	45293
The Dustmite	Dustin Huff	Strength	Dirtiness	Laser Vision	452 Elm St. #3D	Burgtown	OH	45201

First Normal Form

- Eliminate repeating columns
- Add primary key to tables
- Each attribute is atomic

id	name	real_name	power	lair_address	city	state	zip
1	Clean Freak	John Smith	Strength	123 Poplar Ave	Townsburg	OH	45293
1	Clean Freak	John Smith	X-ray vision	123 Poplar Ave	Townsburg	OH	45293
1	Clean Freak	John Smith	Flight	123 Poplar Ave	Townsburg	OH	45293
1	Clean Freak	John Smith	Invisibility	123 Poplar Ave	Townsburg	OH	45293
2	Soap Stud	Efram Jones	Speed	123 Poplar Ave	Townsburg	OH	45293
3	The Dustmite	Dustin Huff	Strength	452 Elm St. #3D	Burgtown	OH	45201
3	The Dustmite	Dustin Huff	Dirtiness	452 Elm St. #3D	Burgtown	OH	45201
3	The Dustmite	Dustin Huff	Laser Vision	452 Elm St. #3D	Burgtown	OH	45201

Second Normal Form

- Separate tables for duplicate data

id	lair_id	name	real_name
1	1	Clean Freak	John Smith
2	1	Soap Stud	Efram Jones
3	2	The Dustmite	Dustin Huff

id	lair_address	city	state	zip
1	123 Poplar Ave	Townsburg	OH	45293
2	452 Elm St. #3D	Burgtown	OH	45201

id	power
1	Stength
2	X-Ray vision
3	Flight
4	Invisibility
5	Speed
6	Dirtiness
7	Laser Vision

char_id	power_id
1	1
1	2
1	3
1	4
2	5
3	1
3	6
3	7

Third Normal Form

- Create separate tables for any transitive or partial dependencies

id	lair_id	name	real_name
1	1	Clean Freak	John Smith
2	1	Soap Stud	Efram Jones
3	2	The Dustmite	Dustin Huff

id	zip_id	lair_address
1	45293	123 Poplar Ave
2	45201	452 Elm St. #3D

id	power
1	Stength
2	X-Ray vision
3	Flight
4	Invisibility
5	Speed
6	Dirtiness
7	Laser Vision

char_id	power_id
1	1
1	2
1	3
1	4
2	5
3	1
3	6
3	7

id	city	state
45293	Townsburg	OH
45201	Burgtown	OH

MySQL Structure

- Servers have databases
- Databases have tables
- Tables have rows
- Rows have values
- MyDB.myTable.

Useful commands

- /usr/bin/mysql -p
  - -p will prompt for password
- show tables
- use [db]
- select *

phpMyAdmin

- Access your MySQL database through a GUI
- Change your password
- Create, edit, delete tables
- Run (and test) queries
- View and print table structures
- Bulk backup and restore

Table Commands

- CREATE - create new databases, tables
- ALTER - modify existing tables
- DELETE - erase data from tables
- DESCRIBE - show structure of tables
- INSERT INTO tablename VALUES - put data in table
- UPDATE - modify existing data in tables
- DROP - destroy table or database (values + structures)

Query Syntax

SELECT [fieldnames] FROM [tablenames] WHERE [criteria] ORDER BY [fieldname to sort on] LIMIT [offset, maxrows]

Query Example

id	lair_id	name	real_name
1	1	Clean Freak	John Smith
2	1	Soap Stud	Efram Jones
3	2	The Dustmite	Dustin Huff

```
SELECT name, real_name FROM heroes
```

name	real_name
Clean Freak	John Smith
Soap Stud	Efram Jones
The Dustmite	Dustin Huff

Joins

SELECT * FROM TableA INNER JOIN TableB on TableA.name = TableB.name

id	name
1	Pirate
2	Monkey
3	Ninja
4	Spaghetti

id	name
1	Rutabaga
2	Pirate
3	Darth Vader
4	Ninja

Joins

SELECT * FROM TableA LEFT OUTER JOIN TableB ON TableA.name = TableB.name

id	name	id	name
1	Pirate	2	Pirate
2	Monkey	null	null
3	Ninja	4	Ninja
4	Spaghetti	null	null

Inserts

```
INSERT INTO `lecture4`.`heroes`
(`id` ,`lair_id` ,`name` ,`real_name`)
VALUES
(NULL , '4', 'General Grime', 'Phillip
Grimaldi');
```

Deletes

```
DELETE FROM
`lecture4`.`heroes` WHERE
`id`='1'
```

Update data

```
UPDATE `lecture4`.`heroes`
SET `name`='Admiral Grime',
`real_name`='Phillip J. Grimaldi' WHERE
`id` = '4'
```

PHP and MySQL

- Connect to a MySQL server
  - $mysqli = new mysqli ("hostname", "user", "pass", "db")
- Send MySQL commands/query to server
  - $results = $mysqli->query("query")
- Shows error message generated by the MySQL server
  - $mysqli->error
- Release results array
  - $results->free()
  - Note: result object is only created for SELECT, SHOW, DESCRIBE or EXPLAIN queries, so for CREATE, INSERT, UPDATE there is nothing to "free".
- Close connection
  - $mysqli->close();

Notes

- General Note
  - If two or more columns of the result have the same field (column) names, the last column will take percedence and overwrite the earlier data.
  - Can use alias in select statement (AS keyword)
  - Can use numeric array index
- $results->fetch_array()
  - stores data in both the numeric indices of the result array AND associate indices using the field (column) name as the key

fetch_array

```
$query = "SELECT * FROM heroes";
$results = $mysqli->query($query)
  or die($mysql->error.__LINE__);
$row = $results->fetch_array();
print_r($row);
```

id	lair_id	name	real_name
1	1	Clean Freak	John Smith
2	1	Soap Stud	Efram Jones
3	2	The Dustmite	Dustin Huff

Array ([0] => Clean Freak [name] => Clean Freak [1] => John Smith [real_name] => John Smith)

Array ([0] => Soap Stud [name] => Soap Stud [1] => Efram Jones [real_name] => Efram Jones)

Array ([0] => The Dustmite [name] => The Dustmite [1] => Dustin Huff [real_name] => Dustin Huff)

For your releases...

- Commit a MySQL dump of your DB...
  - https://dev.mysql.com/doc/refman/5.7/en/mysqldump-sql-format.html

    mysqldump --databases myCS518DB > milestone1dump.sql
- Docker will import your DB
  - http://www.cyberciti.biz/faq/import-mysql-dumpfile-sql-datafile-into-my-database/
  - mysql -uroot --force --verbose < milestone1dump.sql