CSCI-UA:0060-02

Database Design & Web Implementation

Professor Evan Sandhaus
sandhaus@cs.nyu.edu
evan@nytimes.com

Lecture #10: Open Office Base, Life on the Console, MySQL
Administrivia

- Readings:
  - Chapters 6, 7 for Tuesday

- Homework
  - New homework!
On Today’s Menu

- Lets go over the homework real quick
- Open Office Base
  - Spotify Example
- Meet the console
- MySQL
  - Databases on the command line.
SQL

- Structured Query Language
- Debuted in 1974
- Used in LOTS of relational database systems
MySQL

- Relational database engine
- Named after the creator’s daughter My
- Released in 1995
- Now owned by Oracle
- Downloaded 100 Million Times
- Installed on i6 for you to use!
Diving Right In - Connecting

> ssh <username>@i6.cims.nyu.edu
> mysql -u<username> -p<password>
My First Database

CREATE DATABASE spotify;
My First Database

USE spotify;
CREATE TABLE artist(
    artist_id int(11) NOT NULL AUTO_INCREMENT,
    artist_name varchar(255) NOT NULL,
    popularity float NOT NULL DEFAULT 0,
    PRIMARY KEY (artist_id)
) ENGINE=InnoDB;
My First Database

DESC artist;
My First Database

INSERT INTO artist(artist_name, popularity)
VALUES ('Green Day', 0.6);
My First Database

INSERT INTO artist(artist_name)
VALUES ('Lady Gaga');
My First Database

```sql
INSERT INTO artist(artist_name, popularity)
VALUES
('The Beatles', 1.0),
('LMFAO', 0.4),
('Rhianna', .9);
```
My First Database

SELECT * FROM artist;
My First Database

SELECT * FROM artist ORDER BY popularity;
My First Database

SELECT * FROM artist ORDER BY popularity DESC;
My First Database

SELECT * FROM artist WHERE popularity > 0.5;
My First Database

SELECT * FROM artist WHERE artist_id = 1;
My First Database

SELECT * FROM artist WHERE artist_name = 'Green Day';
My First Database

SELECT * FROM artist WHERE artist_name = 'Lady Gaga';
My First Database

UPDATE artist SET popularity = 0.85
WHERE artist_id = 2;
My First Database

CREATE TABLE album(
    album_id int(11) NOT NULL AUTO_INCREMENT,
    artist_id int(11) NOT NULL,
    album_name varchar(255) NOT NULL,
    release_year int(4) NOT NULL,
    PRIMARY KEY (album_id),
    CONSTRAINT FOREIGN KEY (artist_id) REFERENCES artist (artist_id)
) ENGINE=InnoDB;
My First Database

INSERT INTO album (artist_id, album_name, release_year)
VALUES (1, 'Dookie', 1993);

INSERT INTO album (artist_id, album_name, release_year)
VALUES (1, 'Uno', 2012), (1, 'Dos', 2012), (1, 'Tre!', 2013);

INSERT INTO album (artist_id, album_name, release_year)
VALUES (2, 'Born This Way', 2012);
My First Database

```
INSERT INTO album (artist_id, album_name, release_year)
VALUES (6, 'Not a valid artist', 9999);
```
My First Database

SELECT * FROM album;
My First Database

SELECT * FROM artist
JOIN album ON artist.artist_id = album.artist_id;
My First Database

SELECT * FROM artist
JOIN album ON artist.artist_id = album.artist_id
WHERE artist.artist_id = 1;
My First Database

CREATE TABLE genre(
    genre_name varchar(255),
    PRIMARY KEY (genre_name)
) ENGINE=InnoDB;
My First Database

INSERT INTO genre(genre_name) VALUES ('Rock'), ('Pop'), ('Punk'), ('Classic Rock');
My First Database

```
INSERT INTO genre(genre_name)
VALUES ('Rock');
```
My First Database

ALTER TABLE album ADD COLUMN genre_name varchar(255) DEFAULT NULL;
My First Database

ALTER TABLE album ADD CONSTRAINT FOREIGN KEY (genre_name) REFERENCES genre (genre_name);
My First Database

UPDATE album
SET genre_name = 'Electronica'
WHERE artist_id = 2;
My First Database

```
UPDATE album
SET genre_name = 'Pop'
WHERE artist_id = 2;
```
My First Database

UPDATE artist JOIN album ON artist.artist_id = album.artist_id
SET genre_name = 'Punk'
WHERE artist_name = 'Green Day';