Section slides: <http://webdev.slides.com/coltsteele/mysql-106>

# Introduction to Instagram Clone Schema

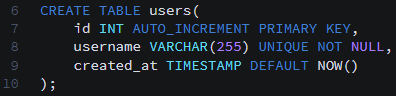
* In this section we will be cloning a portion of the Instagram database, where we’ll have multiple tables to work with and plenty of opportunities to work on our joins
* Let’s start by designing our schema, which will consist of several entities (not necessarily the same as tables)
  + Users
  + Photos/images (represented as an image URL in this case)
  + Number of likes
    - Need a way to prevent people from adding more than 1 like
  + Hashtags
  + Comments
  + Number of followers that user has
  + Number of people the user is following

# Defining the Users Schema

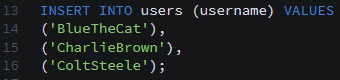
* The *users* schema will look like this



* + The user id is the primary key
  + We want username to be unique as well, but we shouldn’t use it as a primary key because long username strings can take some time to search for. We also want this to be NOT NULL
  + **created\_at** will be a timestamp that defaults the current date and time



* Let’s add a couple of users so that we can play around with them in this section (the next section will add a huge amount of users)



* Code summary

CREATE TABLE users (

id INTEGER AUTO\_INCREMENT PRIMARY KEY,

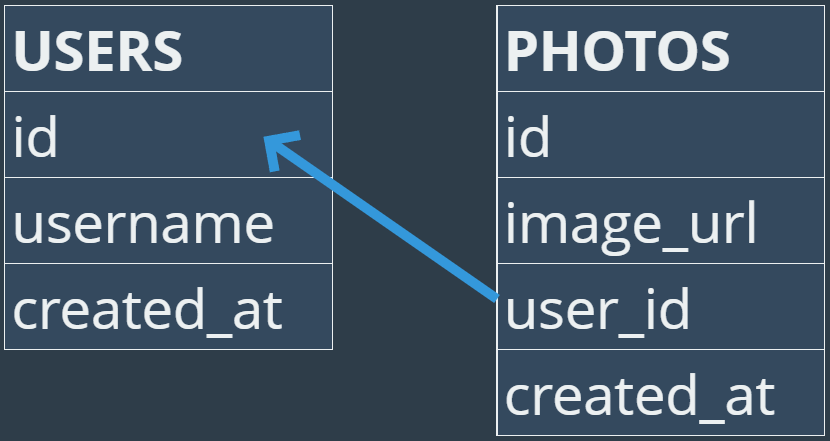
username VARCHAR(255) UNIQUE NOT NULL,

created\_at TIMESTAMP DEFAULT NOW()

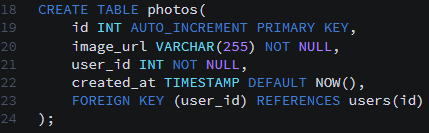
);

# Defining the Photos Schema

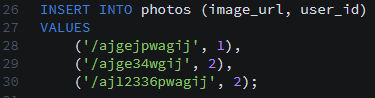
* Here is what our simplified *photos* table will look like, and how it will be connected to *users*



* + We will be using “id” as the primary key, and other tables will reference it
  + The image\_url should be NOT NULL so that we don’t have a situation where we have no photo display
  + “user\_id” must also be NOT NULL – we do not want to have orphan photos with no users



* Let’s insert a couple photos. When adding data to the *photos* table, we will need to supply the **image\_url** and **user\_id**. The others will auto-populate.



* Code summary

CREATE TABLE photos (

id INTEGER AUTO\_INCREMENT PRIMARY KEY,

image\_url VARCHAR(255) NOT NULL,

user\_id INTEGER NOT NULL,

created\_at TIMESTAMP DEFAULT NOW(),

FOREIGN KEY(user\_id) REFERENCES users(id)

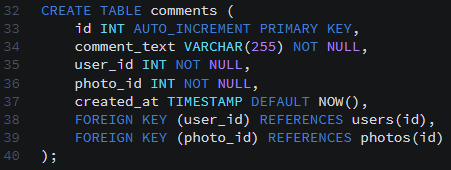
);

# Creating the Comments Schema

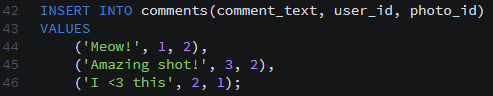
* Comments will rely on users and photos. A comment is authored by someone, and they are attributed to a specific photo
  + This means we will have two foreign keys
* The *comments* table



* + The **id** will be the primary key once again
  + **comment\_text** cannot be blank. In the real Instagram, a blank comment will not post
  + **user\_id** and **photo\_id** will both be foreign keys and will both be NOT NULL
    - user­\_id will tell us who wrote the comment
    - photo\_id will tell us which photo the comment is on



* Let’s insert some data and practice!



* Code summary

CREATE TABLE comments (

id INTEGER AUTO\_INCREMENT PRIMARY KEY,

comment\_text VARCHAR(255) NOT NULL,

photo\_id INTEGER NOT NULL,

user\_id INTEGER NOT NULL,

created\_at TIMESTAMP DEFAULT NOW(),

FOREIGN KEY(photo\_id) REFERENCES photos(id),

FOREIGN KEY(user\_id) REFERENCES users(id)

);