Relational Databases with MySQL Week 3 Coding Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

Instructions: Using a text editor of your choice, write the queries that accomplishes the objectives listed below. Take screenshots of the queries and results and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your Java project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Coding Steps:

You have been asked to create a database for a new social media application that your company is developing.

The database must store user data such as username, email, password, etc...

Users are able to post and comment. So, your database must also store post and comment data.

We need to know which user made which posts.

We also need to know which user made which comments, and which post a comment is on.

Posts and comments should both include the time they were created, and what the content of the post or comment is.

Create an Entity Relationship Diagram (ERD) using draw.io to model the database you will create. Insert a screenshot of the ERD in the screenshots section below.

Write a SQL script to create the database. Insert a screenshot of the SQL in your script.

Hints:

Database: social_media

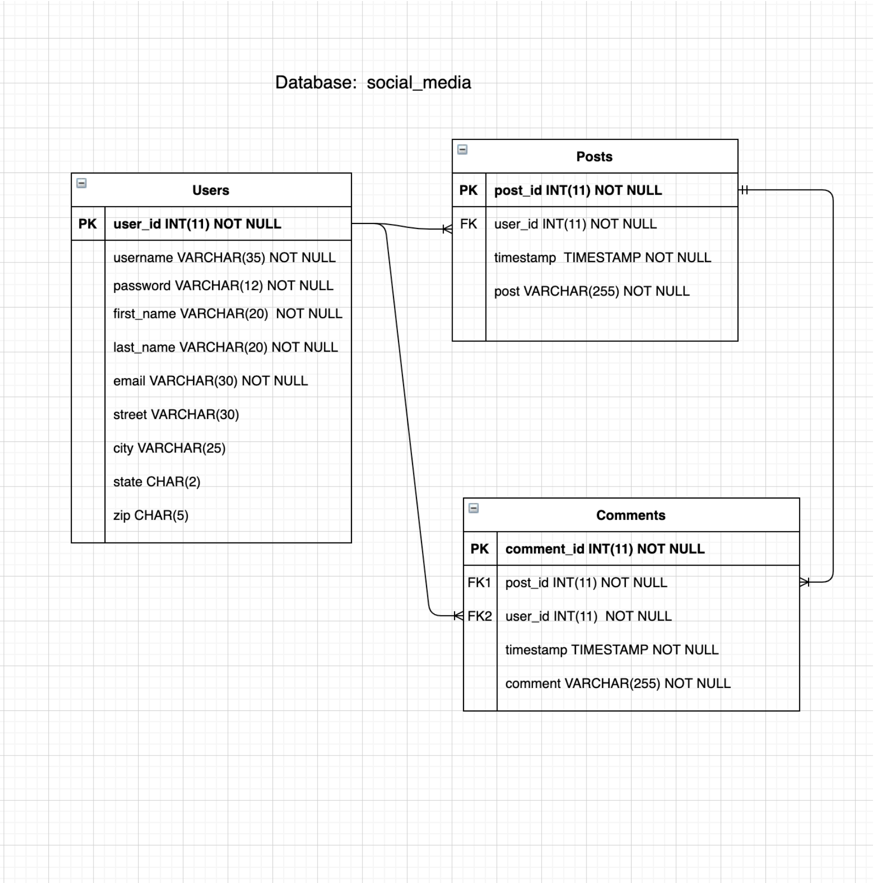
You will only need three tables:

1. users : (user_id (PK), username, password, first_name, last_name,
   email, street, city, state, zip)
2. posts : (post_id (PK), user_id (FK), timestamp, post)
3. comments : (comment_id (PK), post_id (FK), user_id (FK), timestamp, comment)

Two tables will have foreign key references. (tables: posts & comments)

One table will have two foreign key references. (table: comments)

Screenshots:

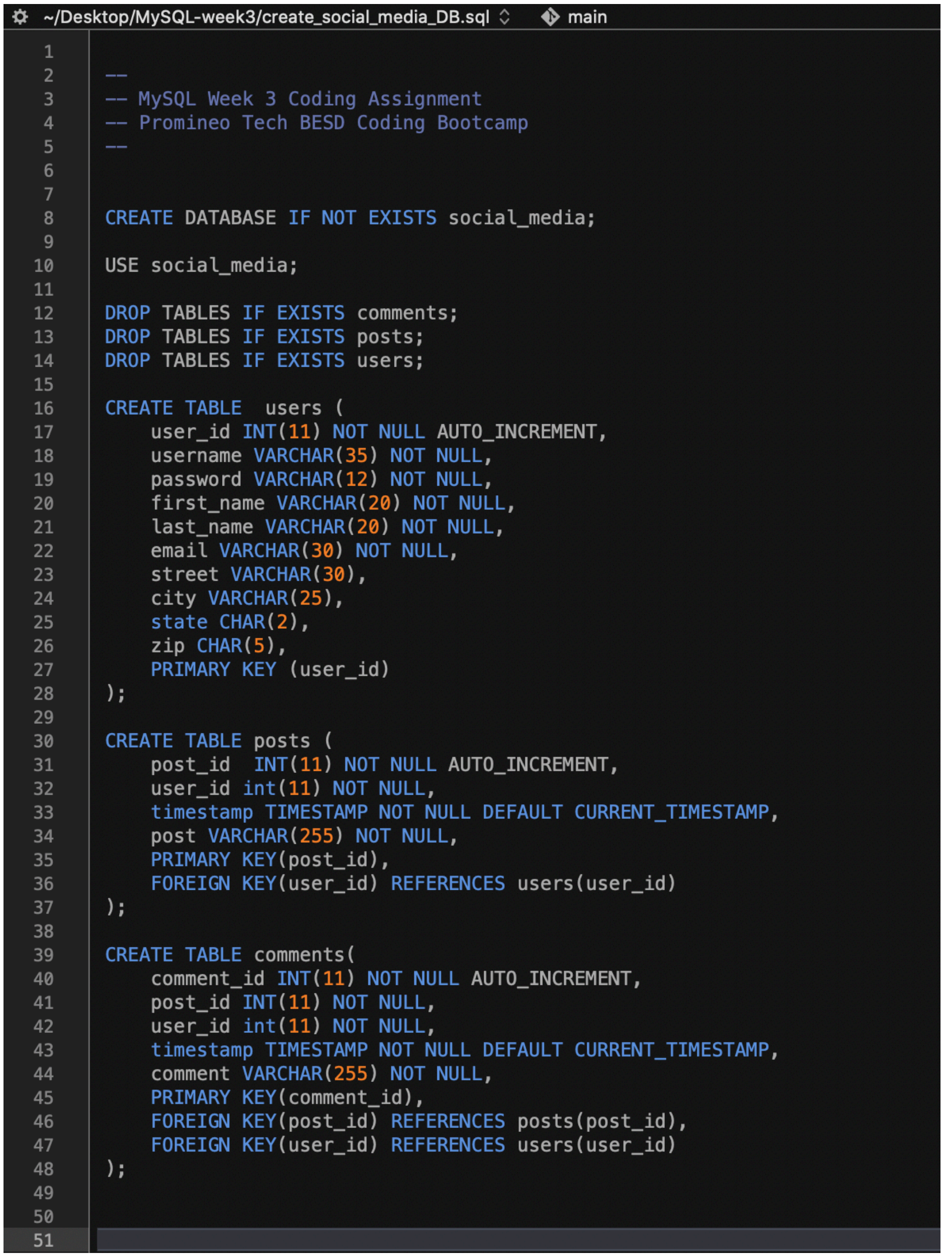


Database: social_media

	Users
PK	**user_id INT(11) NOT NULL**
	username VARCHAR(35) NOT NULL
	password VARCHAR(12) NOT NULL
	first_name VARCHAR(20) NOT NULL
	last_name VARCHAR(20) NOT NULL
	email VARCHAR(30) NOT NULL
	street VARCHAR(30)
	city VARCHAR(25)
	state CHAR(2)
	zip CHAR(5)

	Posts
PK	**post_id INT(11) NOT NULL**
FK	user_id INT(11) NOT NULL
	timestamp TIMESTAMP NOT NULL
	post VARCHAR(255) NOT NULL

	Comments
PK	**comment_id INT(11) NOT NULL**
FK1	post_id INT(11) NOT NULL
FK2	user_id INT(11) NOT NULL
	timestamp TIMESTAMP NOT NULL
	comment VARCHAR(255) NOT NULL



```sql

--
-- MySQL Week 3 Coding Assignment
-- Promineo Tech BESD Coding Bootcamp
--


CREATE DATABASE IF NOT EXISTS social_media;

USE social_media;

DROP TABLES IF EXISTS comments;
DROP TABLES IF EXISTS posts;
DROP TABLES IF EXISTS users;

CREATE TABLE  users (
    user_id INT(11) NOT NULL AUTO_INCREMENT,
    username VARCHAR(35) NOT NULL,
    password VARCHAR(12) NOT NULL,
    first_name VARCHAR(20) NOT NULL,
    last_name VARCHAR(20) NOT NULL,
    email VARCHAR(30) NOT NULL,
    street VARCHAR(30),
    city VARCHAR(25),
    state CHAR(2),
    zip CHAR(5),
    PRIMARY KEY (user_id)
);

CREATE TABLE posts (
    post_id  INT(11) NOT NULL AUTO_INCREMENT,
    user_id int(11) NOT NULL,
    timestamp TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
    post VARCHAR(255) NOT NULL,
    PRIMARY KEY(post_id),
    FOREIGN KEY(user_id) REFERENCES users(user_id)
);

CREATE TABLE comments(
    comment_id INT(11) NOT NULL AUTO_INCREMENT,
    post_id INT(11) NOT NULL,
    user_id int(11) NOT NULL,
    timestamp TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
    comment VARCHAR(255) NOT NULL,
    PRIMARY KEY(comment_id),
    FOREIGN KEY(post_id) REFERENCES posts(post_id),
    FOREIGN KEY(user_id) REFERENCES users(user_id)
);


```

URL to GitHub Repository: https://github.com/sw-dev-lisa-s-nh/MySQL-week3.git