# Relational Databases with MySQL Week 9 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 to the repository. Additionally, push an .sql file with all your queries and your ERD to the same 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:

You will only need three tables.

Two tables will have foreign key references.

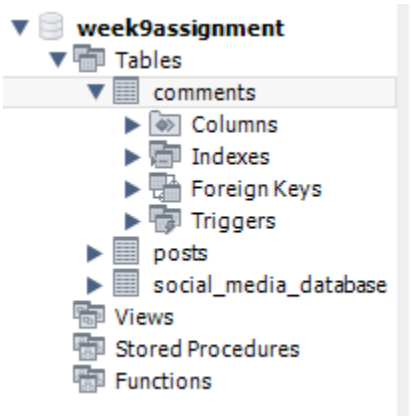
One table will have two foreign key references.

**Screenshots:**

**Query:**

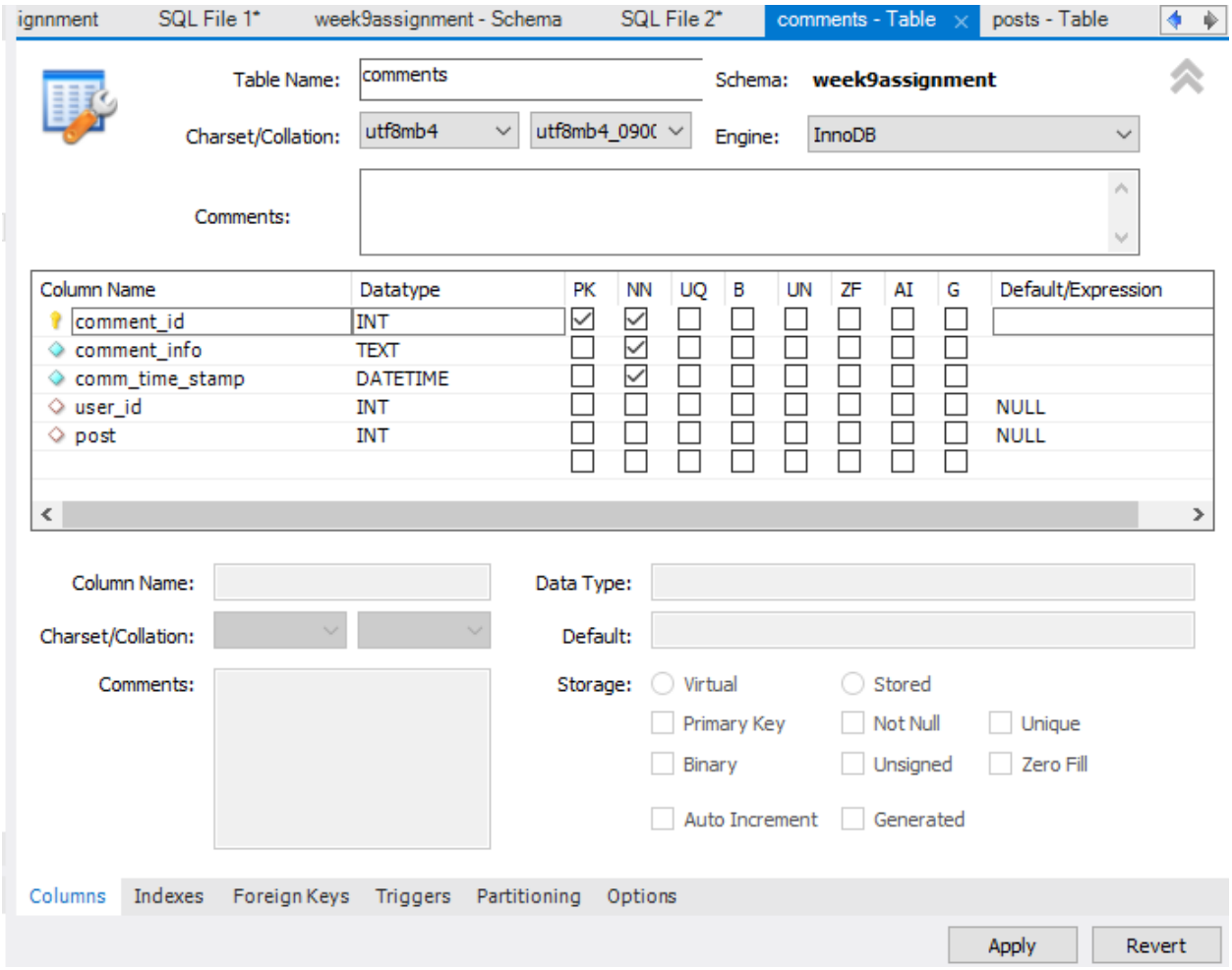
```
create table Social_Media_Database (
user_id int primary key, username varchar(100), email varchar(100),
passwor varchar(20)
);

create table Posts (
post int primary key, post_content text not null,  time_stamp datetime not null,
user_id int,
foreign key(user_id) references social_media_database(user_id)
);

 create table Comments (
comment_id int primary key, comment_info text not null,
comm_time_stamp datetime not null,
user_id int, post int,
foreign key(user_id) references social_media_database(user_id),
foreign key(post) references posts(post)
);
```

**Results:**

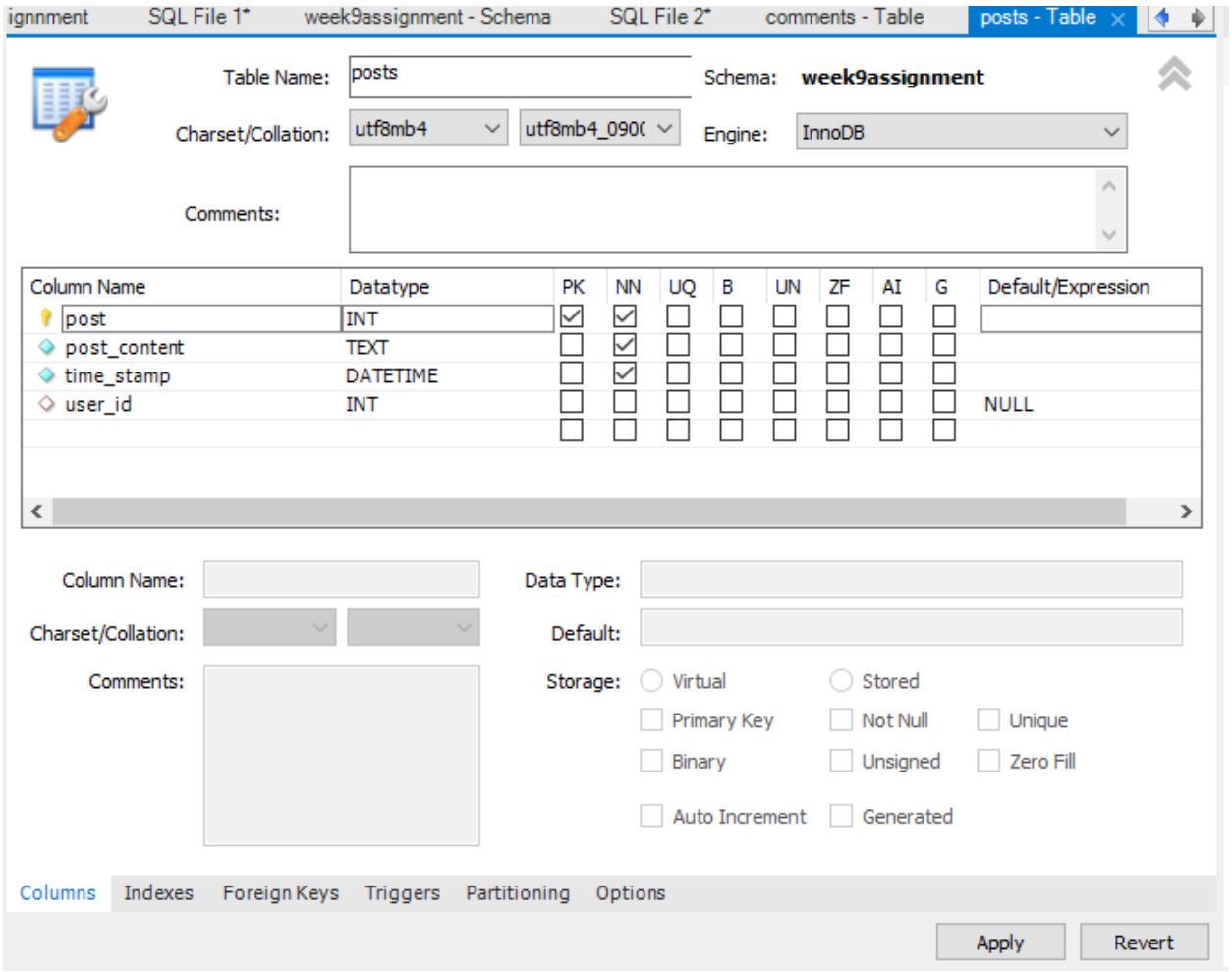


```
65  15:32:23  create table Social_Media_Database ( user_id int primary key, username varchar(100), email varchar(100), passwor varchar(20) )   0 row(s) affected
66  15:32:23  create table Posts ( post int primary key, post_content text not null, time_stamp datetime not null, user_id int, foreign key(user_id) references social_media_database(user_id) )   0 row(s) affected
67  15:32:23  create table Comments ( comment_id int primary key, comment_info text not null, comm_time_stamp datetime not null, user_id int, post int, foreign key(user_id) references social_media_database(user_id), foreign key(p...   0 row(s) affected
```

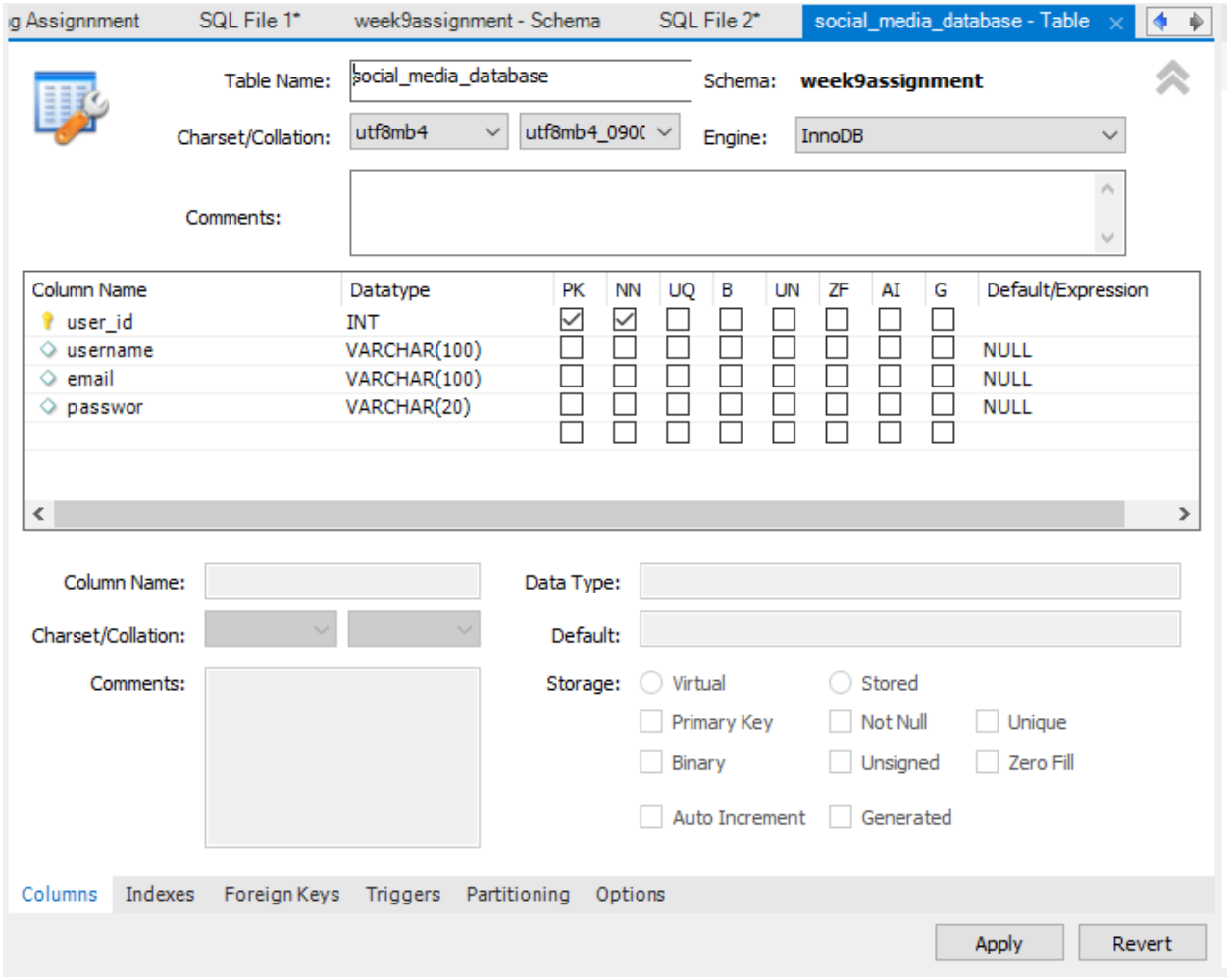
## Comments:



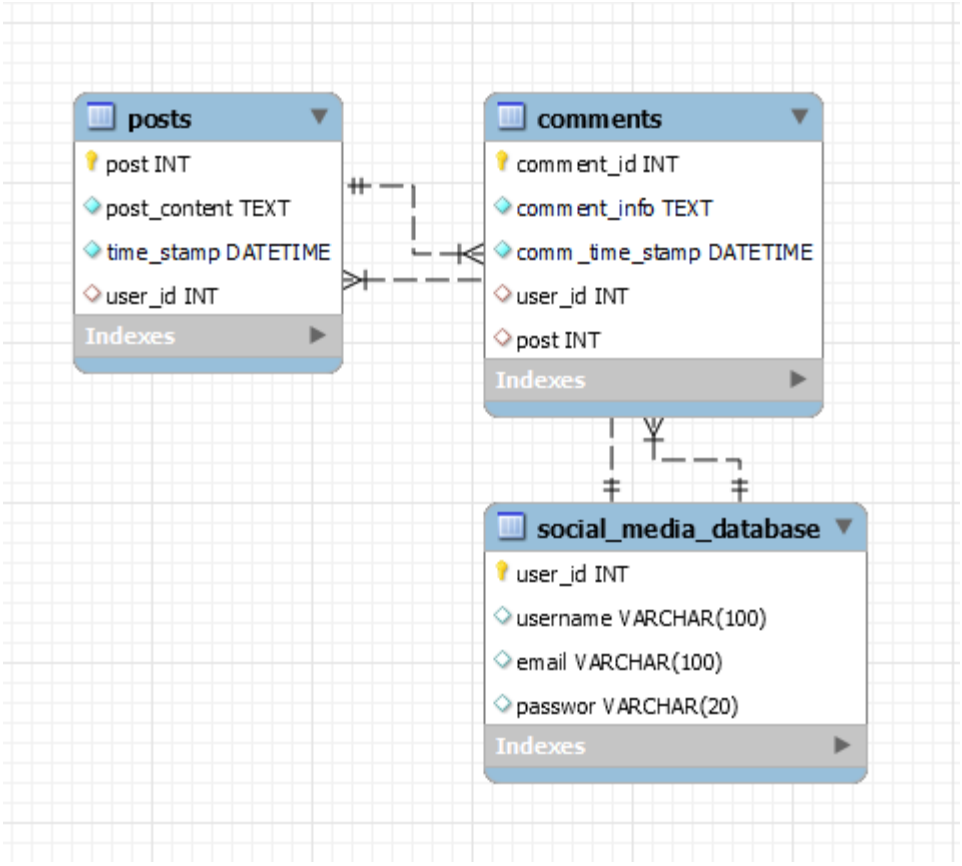
**Posts:**


ignnment
SQL File 1*
week9assignment - Schema
SQL File 2*
comments - Table
posts - Table
Table Name:
posts
Schema:
week9assignment
Charset/Collation:
utf8mb4
utf8mb4_0900
Engine:
InnoDB
Comments:
Column Name
Datatype
PK
NN
UQ
B
UN
ZF
AI
G
Default/Expression
post
INT
post_content
TEXT
time_stamp
DATETIME
user_id
INT
NULL
Column Name:
Data Type:
Charset/Collation:
Default:
Comments:
Storage:
Virtual
Stored
Primary Key
Not Null
Unique
Binary
Unsigned
Zero Fill
Auto Increment
Generated
Columns
Indexes
Foreign Keys
Triggers
Partitioning
Options
Apply
Revert

**Social media database:**



**ERD:**




**URL to GitHub Repository:**

**https://github.com/robertspwork/Week-9-Codinng-Assignment.git**