**Class: Computer Science 303: Database Management**

**Name: Myranda Brandt**

**Assignment 1**

**Prompt 1 - Create the Schema/Database**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Prompt 2 - Create Tables**

USE finalproject;

/\*Use zero fill on zip to make sure leading zeros do not disappear. \*/

CREATE TABLE users

(userid INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100),

username VARCHAR(20),

address VARCHAR(300),

city VARCHAR(100),

state VARCHAR(2),

zip MEDIUMINT(5) ZEROFILL)

;

CREATE TABLE locations

(itemid INT PRIMARY KEY AUTO\_INCREMENT,

type INT,

description VARCHAR(500),

lng REAL,

lat REAL)

;

CREATE TABLE photographs

(photoid INT PRIMARY KEY,

locationid INT)

;

A screenshot of a computer

Description automatically generated

**Prompt 3 - Alter Tables**

ALTER TABLE locations MODIFY type INT NOT NULL;

ALTER TABLE locations MODIFY description VARCHAR(500) NOT NULL;

ALTER TABLE locations MODIFY lng REAL NOT NULL;

ALTER TABLE locations MODIFY lat REAL NOT NULL;

ALTER TABLE users MODIFY name VARCHAR(100) NOT NULL;

ALTER TABLE users MODIFY username VARCHAR(20) NOT NULL;

ALTER TABLE photographs MODIFY photoid INT NOT NULL;

ALTER TABLE photographs MODIFY locationid INT NOT NULL;

A screenshot of a computer

Description automatically generated

**Prompt 4 – Create Index**

CREATE UNIQUE INDEX id ON users (userid)

;

CREATE UNIQUE INDEX photoid ON photographs (photoid)

;

A screenshot of a computer

Description automatically generated

**Prompt 5 - Enter Data**

INSERT INTO users (name, username, address, city, state, zip)

VALUES ('Bonnie Buntcake', 'bbunt', '6709 Wonder Street', 'Wonderbread', 'OH', 46106),

('Sam Smarf', 'ssmarf', '356 A Street', 'Beefy', 'PA', 19943),

('Wendy Grog', 'wgrog', '900 Star Street', 'Mary', 'MD', 21340),

('Joe Jogger', 'jjogger', '183713 N North Street', 'Norther', 'WV', 51423)

;

A screenshot of a computer

Description automatically generated

**Prompt 6 - Count Rows**

SELECT COUNT(\*) FROM users

;

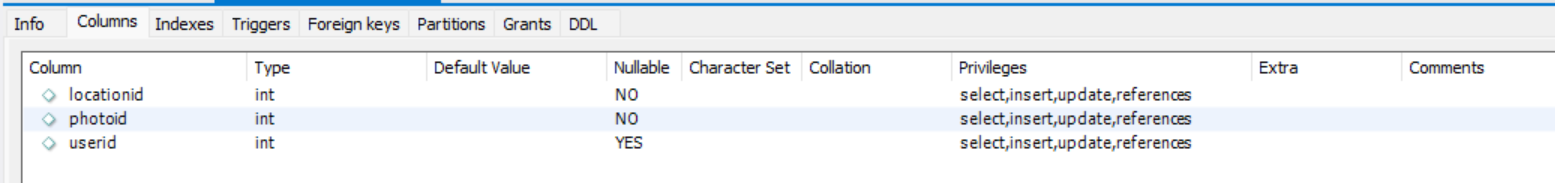
A screenshot of a computer program

Description automatically generated

**Prompt 7 - Add Column**

ALTER TABLE photographs ADD COLUMN userid INT AFTER locationid

;



**Prompt 8** **- Issue with New Column**

ALTER TABLE photographs MODIFY userid INT NOT NULL;

A screenshot of a message

Description automatically generated

In a previous step we updated multiple columns to have the constraint NOT NULL. This ensures that the columns will not have null values. If we allow the new userid column in the photographs table to accept null values we will not be able to identify the user who took the photo. To resolve this we need to make the userid column have a NOT NULL constraint so we are able to join the photographs table to the users table and identify the taker of the picture. This will allow for better data integrity because the tables will be able to be joined together, staying consistent with the idea that all foreign keys should be able to join to a primary key in another table.

**Prompt 9 - Location and Photograph Table Updates**

INSERT INTO locations (type, description, lng, lat)

VALUES (1, 'Independence Hall', 794.35, 651.43),

(2, '6709 Wonder Street', 323.41, 412.22),

(1, 'Sunrise', 221.45, 132.43),

(2, '356 A Street', 123.32, 222.43),

(1, 'Mountains', 34.12, 87.99),

(2, '900 Star Street', 1071.9, 206.45),

(1, 'Moonrise', 816.2, 111.2),

(2, '183714 N North Street', 176.11, 11.176)

;

/\*In the last query where I inserted data into the locations table

itemid numbers were automatically created because I gave that column

an auto\_increment constraint. Since we entered 8 rows of data, there

are itemids made from 1 to 8, I can use any of those for the locationid

in the next insert statement \*/

INSERT INTO photographs (photoid, locationid, userid)

VALUES (1, 4, 1),

(2, 2, 1),

(3, 6, 3),

(4, 1, 4)

;

A screenshot of a computer

Description automatically generated

SELECT \* FROM locations

;

A screenshot of a computer

Description automatically generated

SELECT \* FROM photographs

;

A screenshot of a computer

Description automatically generated

**Prompt 10 - Users**

SELECT name FROM users

;

A screenshot of a computer

Description automatically generated

**Prompt 11- Who's Taking Pictures?**

SELECT name FROM USERS us, photographs pgh WHERE us.userid = pgh.userid

;

A screenshot of a computer

Description automatically generated

**Prompt 12 - Unique Names**

SELECT DISTINCT name FROM USERS us, photographs pgh WHERE us.userid = pgh.userid

;

A screenshot of a computer

Description automatically generated