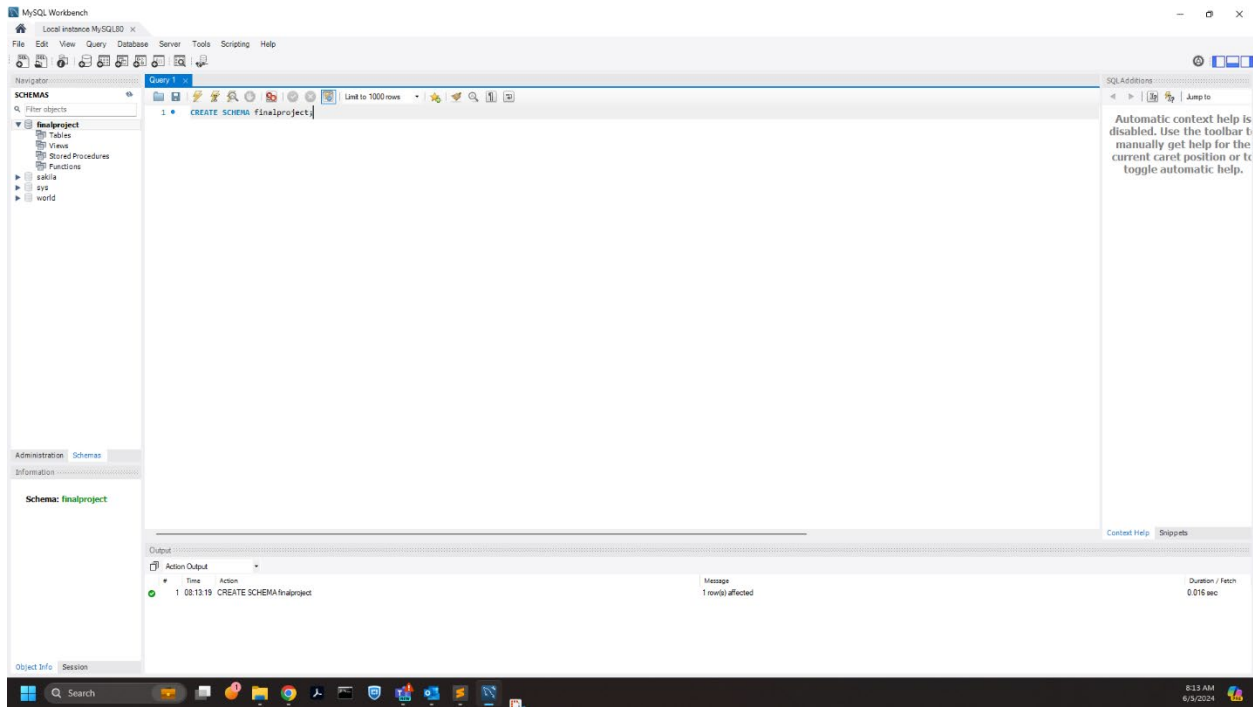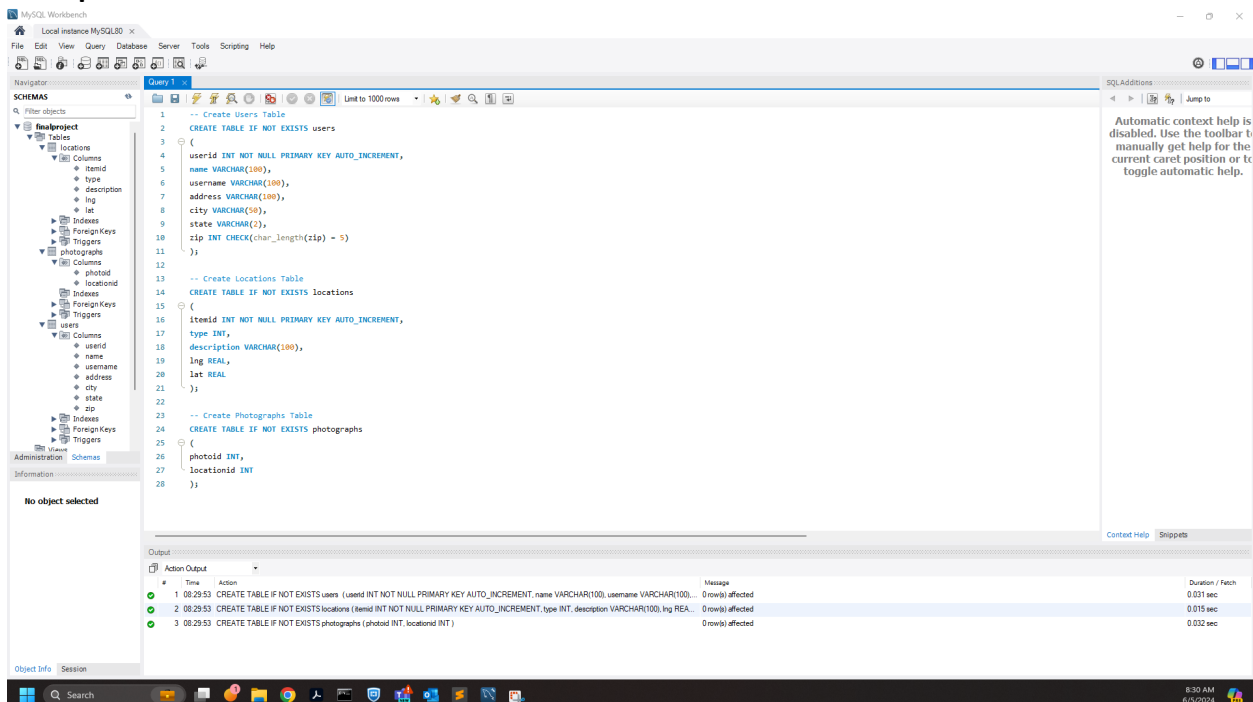Warren Spears

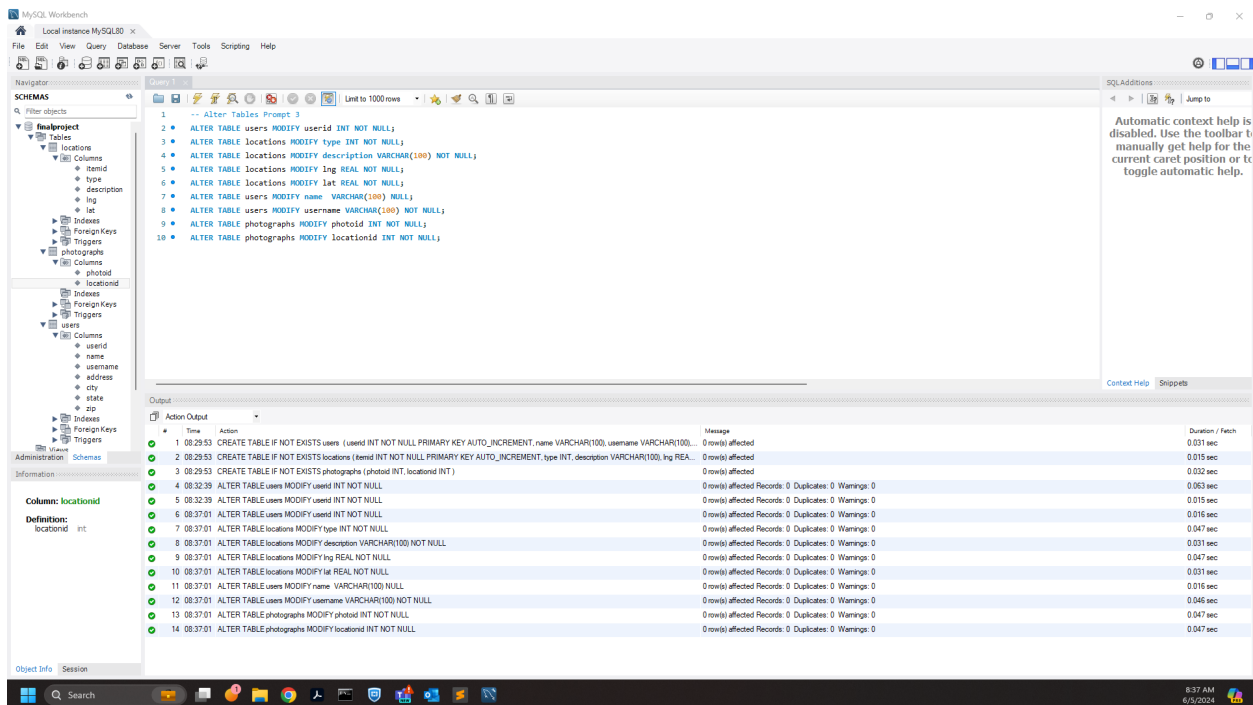Computer Science 303 Database Management

6/5/2024

Prompt 1

Prompt 2

-- Create Tables Prompt 2

CREATE TABLE IF NOT EXISTS users (userid INT NOT NULL PRIMARY KEY AUTO_INCREMENT, name VARCHAR(100), username VARCHAR(100), address VARCHAR(100), city VARCHAR(50), state VARCHAR(2), zip INT CHECK(char_length(zip) = 5));

CREATE TABLE IF NOT EXISTS locations (itemid INT NOT NULL PRIMARY KEY AUTO_INCREMENT, type INT, description VARCHAR(100), lng REAL, lat REAL);

CREATE TABLE IF NOT EXISTS photographs (photoid INT, locationid INT);

Prompt 3

-- Alter Tables Prompt 3

ALTER TABLE users MODIFY userid INT NOT NULL;

ALTER TABLE users MODIFY userid INT NOT NULL;

ALTER TABLE users MODIFY userid INT NOT NULL;

ALTER TABLE locations MODIFY type INT NOT NULL;

ALTER TABLE locations MODIFY description VARCHAR(100) NOT NULL;

ALTER TABLE locations MODIFY lng REAL NOT NULL;

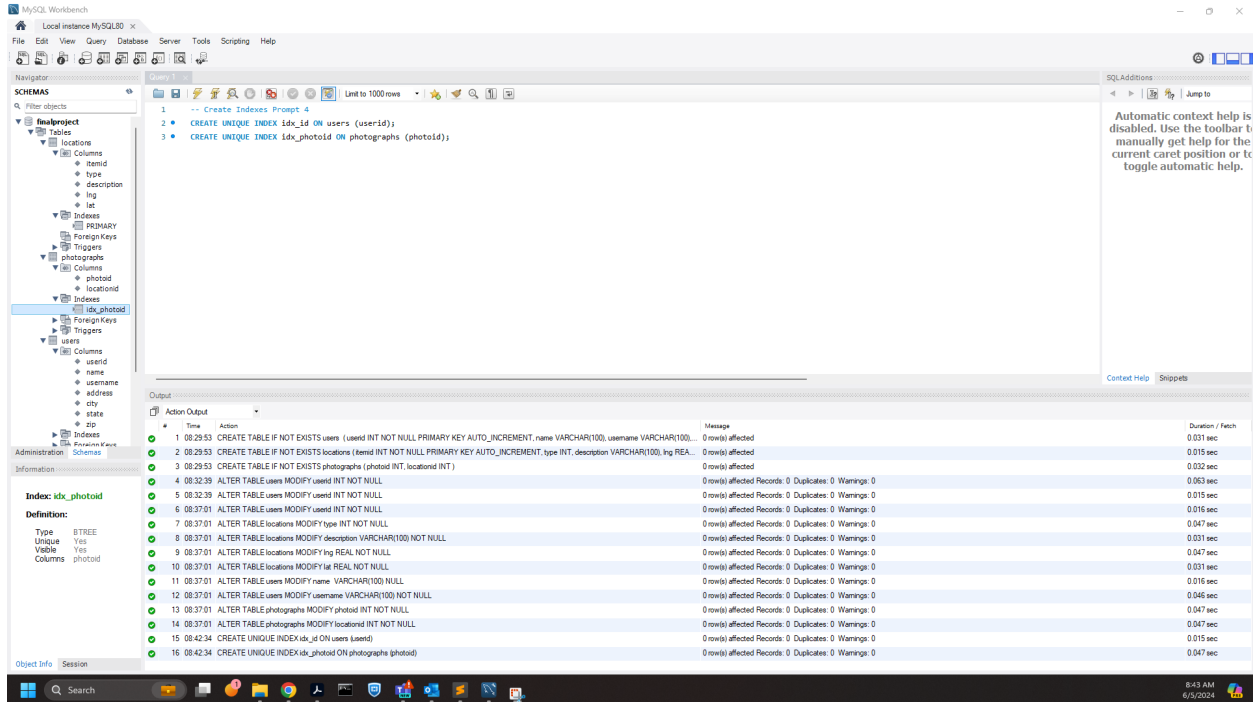ALTER TABLE locations MODIFY lat REAL NOT NULL;

ALTER TABLE users MODIFY name VARCHAR(100) NULL;

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

ALTER TABLE photographs MODIFY photoid INT NOT NULL;

ALTER TABLE photographs MODIFY locationid INT NOT NULL;

Prompt 4

-- Create Indexes Prompt 4

CREATE UNIQUE INDEX idx_id ON users (userid);

CREATE UNIQUE INDEX idx_photoid ON photographs (photoid);

Prompt 5

-- Insert Data Into Users Table Prompt 5

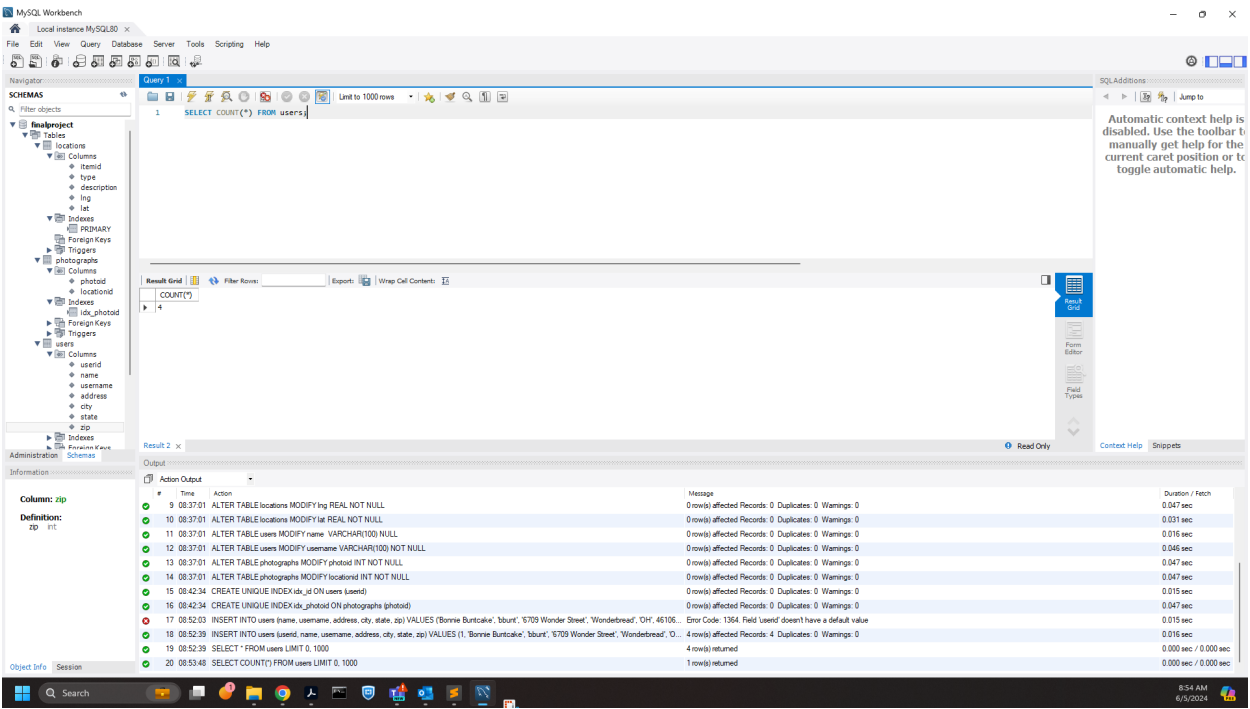INSERT INTO users (userid, name, username, address, city, state, zip) VALUES

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

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

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

(4, 'Joe Jogger', 'jjogger', '183713 N North Street', 'Norther', 'WV', '51423');
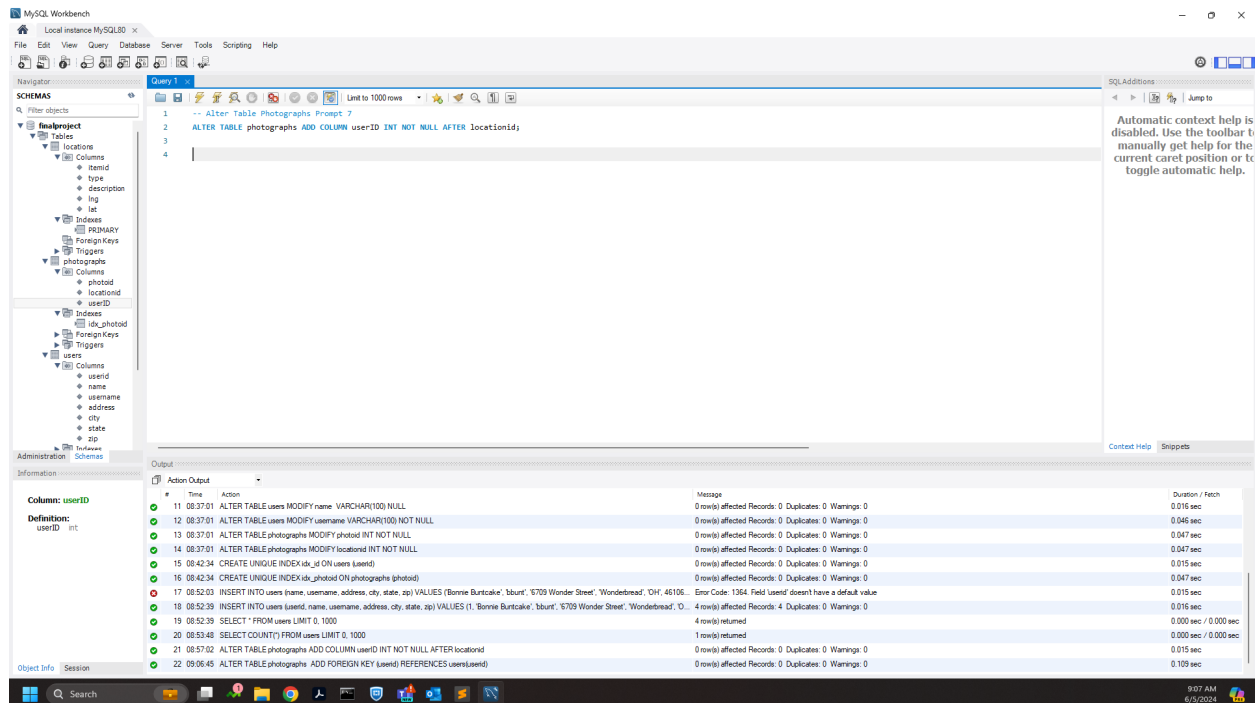
Prompt 6

-- Prompt 6

SELECT count(*) from users;

Prompt 7

-- Alter Table Photographs Prompt 7

ALTER TABLE photographs ADD COLUMN userID INT NOT NULL AFTER locationid;

Prompt 8

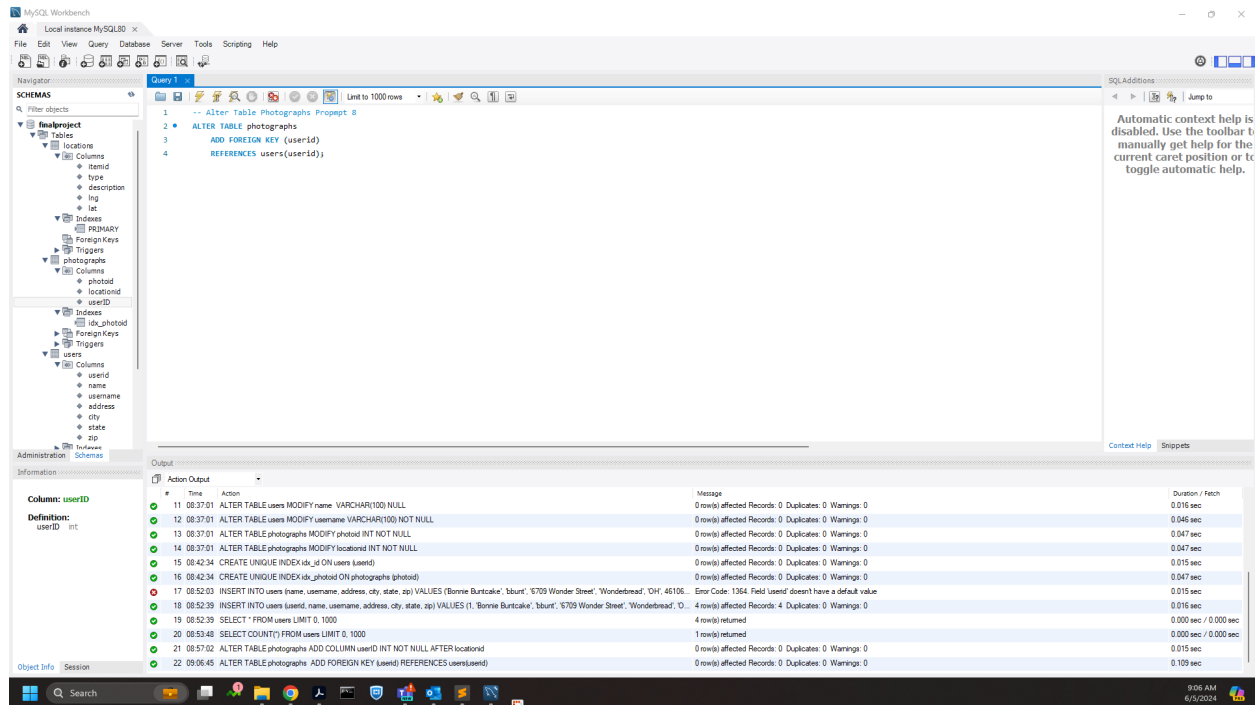The new column created in the the Photographs table called userid has no reference or relationship to the rest of the database. It is a field without context.

The photograph table stores a photo take by a 'user'. The V1.0 of the database, the photographs table was stand along and had no relationship to any other table and no way to discern what information in the table was to a specific individual. To correct this we create a 'FOREIGN KEY' reference from the 'userid' in the photographs table to the 'userid' in the 'users' table. This will create a meaningful relationship between the two tables and allow for correct response from queries on the database as a whole.
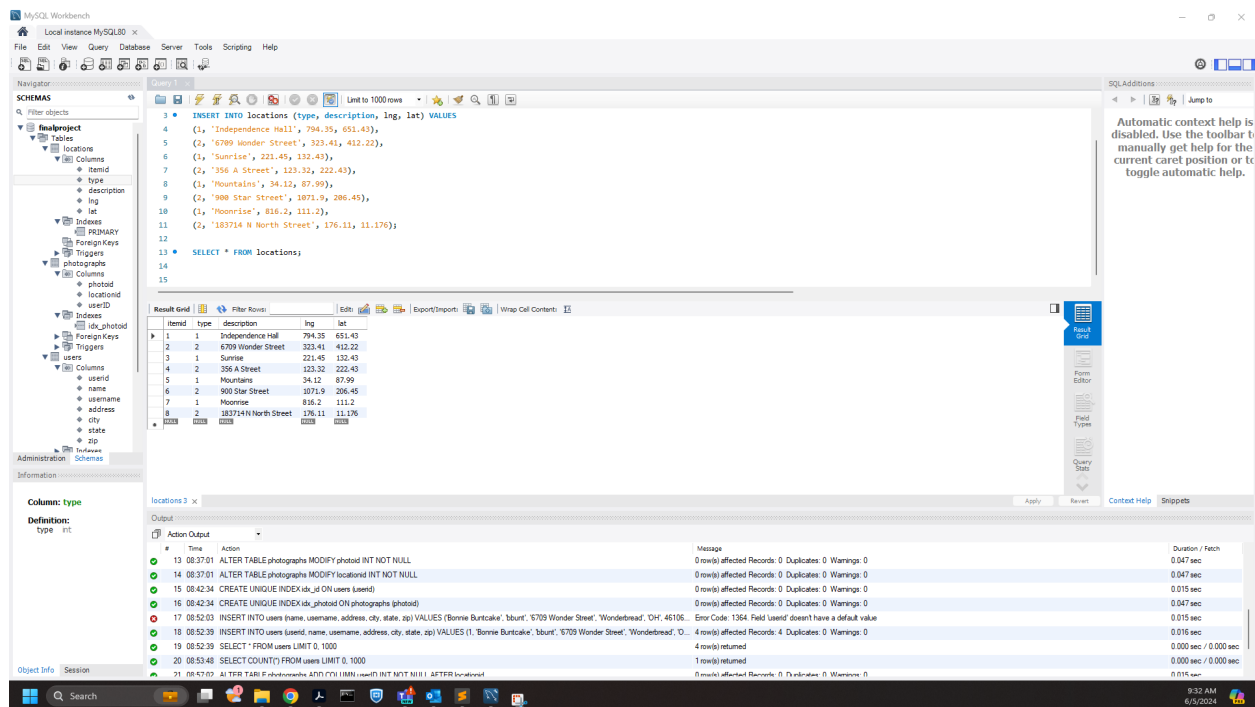
-- Alter Table Photographs Propmpt 8

ALTER TABLE photographs

      ADD FOREIGN KEY (userid)

      REFERENCES users(userid);

Prompt 9

-- Insert Data Into Locations and Photographs Tables Prompt 9

-- Step 1: Insert data into locations table

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

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

(2, '6709 Wonder Street', 323.41, 412.22),
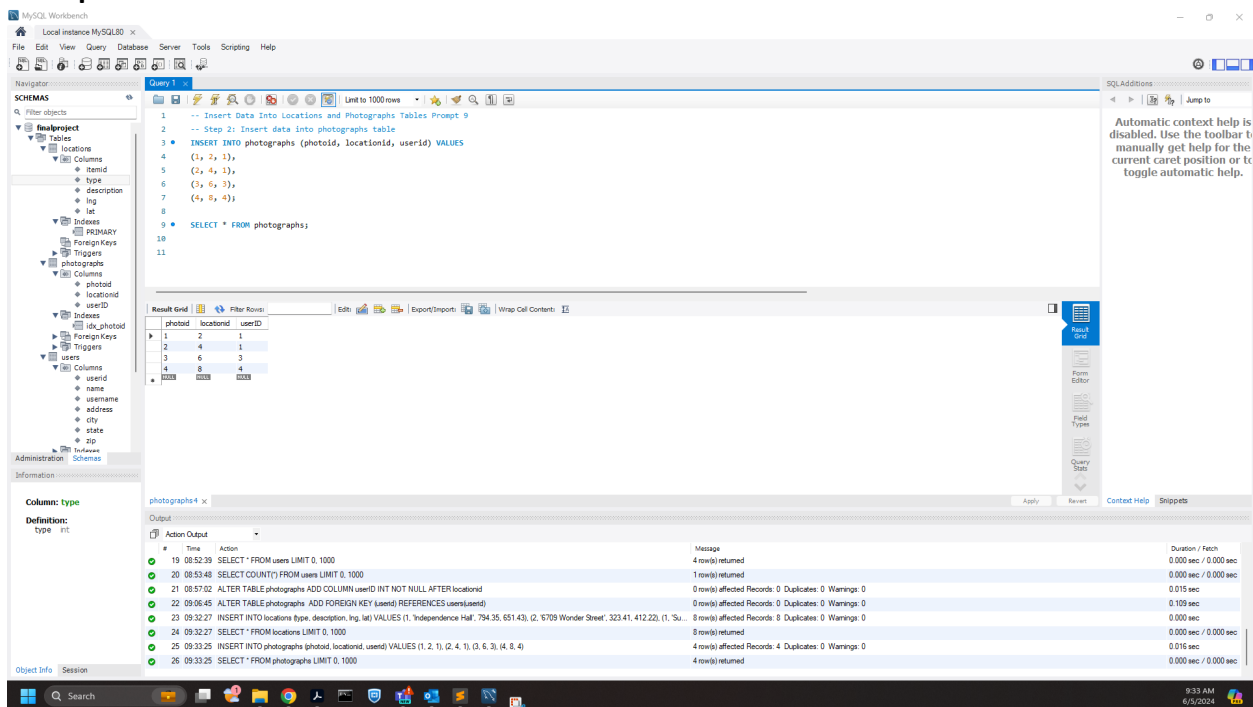
(1, 'Sunrise', 221.45, 132.43),

(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);

SELECT * FROM locations;

Prompt 9 Cont.

-- Step 2: Insert data into photographs table

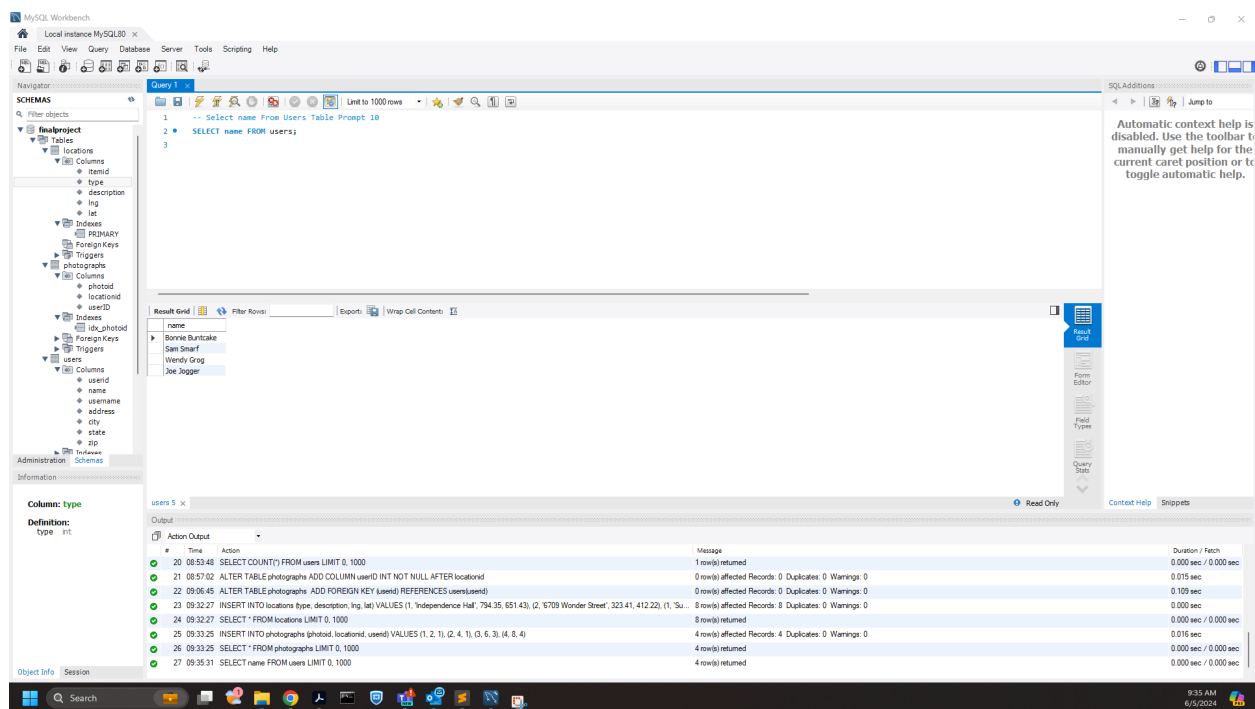INSERT INTO photographs (photoid, locationid, userid) VALUES

(1, 2, 1),

(2, 4, 1),

(3, 6, 3),

(4, 8, 4);

SELECT * FROM photographs;

Prompt 10

-- Select name From Users Table Prompt 10

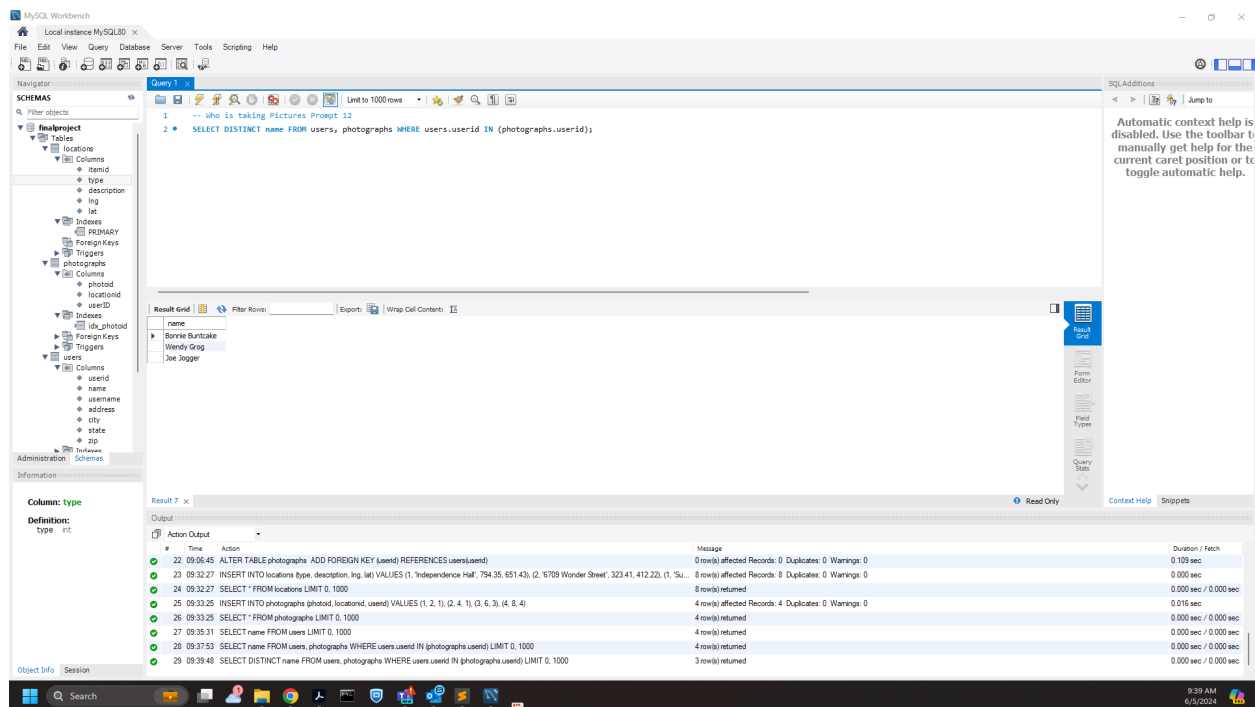SELECT name FROM users;

Prompt 11
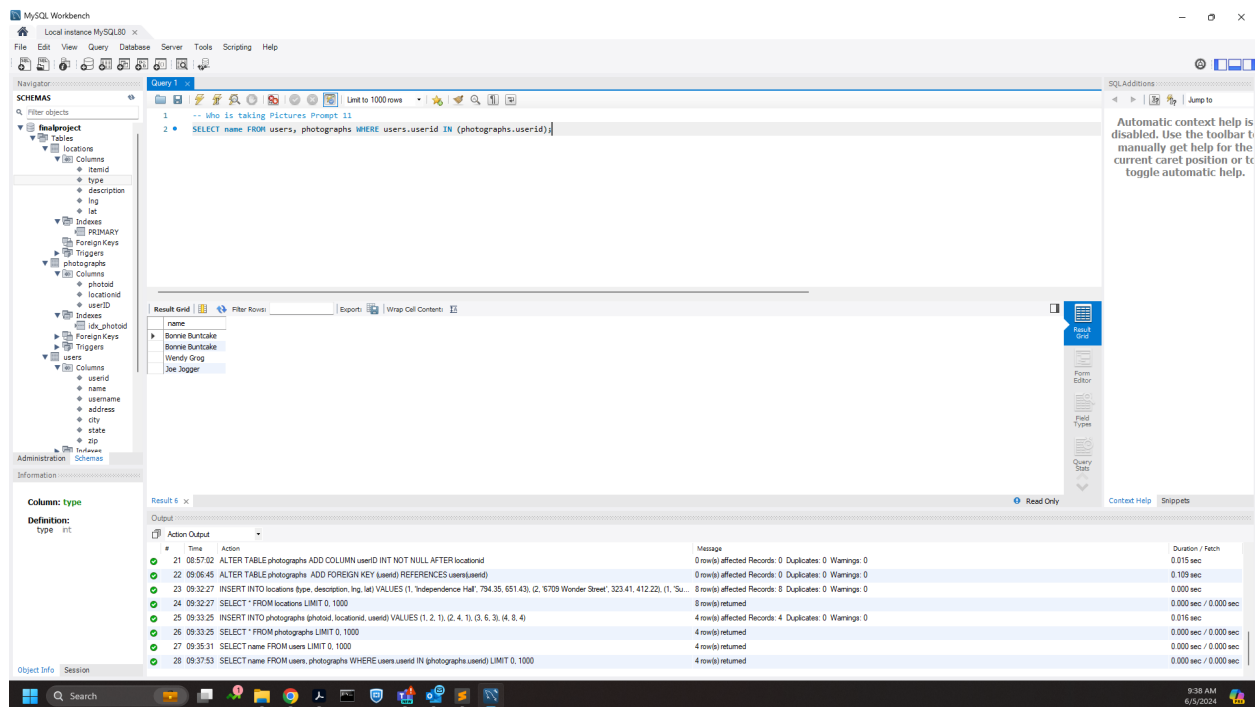
-- Who is taking Pictures Prompt 11

SELECT name FROM users, photographs WHERE users.userid IN (photographs.userid);

Prompt 12

-- Who is taking Pictures Prompt 12

SELECT DISTINCT name FROM users, photographs WHERE users.userid IN (photographs.userid);