Query Schemas and Webpage Results

Query to Update a Flower

Schema:

UPDATE FLOWERS
SET GENUS = "New Genus Name if Specified", SPECIES = "New Species Name if Specified"
WHERE COMNAME = "Queried Common Name";

Webpage:

Query for Ten Most Recent Sightings of a Flower

Schema:

SELECT SIGHTED, LOCATION, PERSON FROM SIGHTINGS
WHERE NAME="Queried Flower Name"
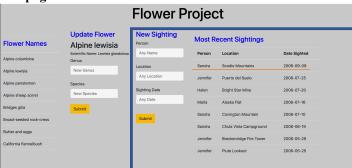ORDER BY SIGHTED DESC LIMIT 10;

Webpage:

Query to Insert a New Sighting of a Flower

Schema:

INSERT INTO SIGHTINGS(NAME, PERSON, LOCATION, SIGHTED)
VALUES("Inputted Flower", "Inputted Person", "Inputted Location", "Inputted Date");

Webpage:

Extra Credit for Back End

Sightings Indexes: 5 Points
CREATE index sightings_name_index on SIGHTINGS(name);
CREATE index sightings_person_index on SIGHTINGS(person);
CREATE index sightings_location_index on SIGHTINGS(location);
CREATE index sightings_sighted_index on SIGHTINGS(sighted);

Log Triggers for Events on Tables: 15 Points
Log Table:
CREATE TABLE TABLE_LOGS(EVENT_LOG TEXT, TIME_STAMP TEXT);

Flowers Table Log Triggers:
```
CREATE TRIGGER flowers_insert_log AFTER INSERT ON FLOWERS
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
    VALUES('(' || NEW.GENUS || ', ' || NEW.SPECIES || ', ' || NEW.COMNAME || ')' ||
        ' INSERTED INTO FLOWERS AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;I
CREATE TRIGGER flowers_update_log AFTER UPDATE ON FLOWERS
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
    VALUES('(' || OLD.GENUS || ', ' || OLD.SPECIES || ', ' || OLD.COMNAME || ')' ||
        ' UPDATED TO ' || '(' || NEW.GENUS || ', ' || NEW.SPECIES || ', ' || NEW.COMNAME || ')' ||
        ' IN FLOWERS AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;
CREATE TRIGGER flowers_delete_log AFTER DELETE ON FLOWERS
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
    VALUES('(' || OLD.GENUS || ', ' || OLD.SPECIES || ', ' || OLD.COMNAME || ')' ||
        ' DELETED FROM FLOWERS AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;
```

Features Table Log Triggers:
```
CREATE TRIGGER features_insert_log AFTER INSERT ON FEATURES
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
    VALUES('(' || NEW.LOCATION || ', ' || NEW.CLASS || ', ' || NEW.LATITUDE || ', ' || NEW.LONGITUDE || ', ' || NEW.MAP || ', ' ||
NEW.ELEV || ')' ||
        ' INSERTED INTO FEATURES AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;
CREATE TRIGGER features_update_log AFTER UPDATE ON FEATURES
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
    VALUES('(' || OLD.LOCATION || ', ' || OLD.CLASS || ', ' || OLD.LATITUDE || ', ' || OLD.LONGITUDE || ', ' || OLD.MAP || ', ' ||
OLD.ELEV || ')' ||
        ' UPDATED TO ' ||
        '(' || NEW.LOCATION || ', ' || NEW.CLASS || ', ' || NEW.LATITUDE || ', ' || NEW.LONGITUDE || ', ' || NEW.MAP || ', ' ||
NEW.ELEV || ')' ||
        ' IN FEATURES AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;
CREATE TRIGGER features_delete_log AFTER DELETE ON FEATURES
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
    VALUES('(' || OLD.LOCATION || ', ' || OLD.CLASS || ', ' || OLD.LATITUDE || ', ' || OLD.LONGITUDE || ', ' || OLD.MAP || ', ' ||
OLD.ELEV || ')' ||
        ' DELETED FROM FEATURES AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;
```

Sightings Table Log Triggers:

```
CREATE TRIGGER sightings_insert_log AFTER INSERT ON SIGHTINGS
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
   VALUES('(' || NEW.NAME || ', ' || NEW.PERSON || ', ' || NEW.LOCATION || ', ' || NEW.SIGHTED || ')' ||
      ' INSERTED INTO SIGHTINGS AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;
CREATE TRIGGER sightings_update_log AFTER UPDATE ON SIGHTINGS
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
   VALUES('(' || OLD.NAME || ', ' || OLD.PERSON || ', ' || OLD.LOCATION || ', ' || OLD.SIGHTED || ')' ||
      ' UPDATED TO ' ||
      '(' || NEW.NAME || ', ' || NEW.PERSON || ', ' || NEW.LOCATION || ', ' || NEW.SIGHTED || ')' ||
      ' IN SIGHTINGS AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;
CREATE TRIGGER sightings_delete_log AFTER DELETE ON SIGHTINGS
BEGIN
        INSERT INTO TABLE_LOGS(event_log, time_stamp)
   VALUES('(' || OLD.NAME || ', ' || OLD.PERSON || ', ' || OLD.LOCATION || ', ' || OLD.SIGHTED || ')' ||
 ' DELETED FROM SIGHTINGS AT TIME ', STRFTIME('%Y-%m-%d %H:%M:%f', 'now'));
END;
```

Queries Used to Test Log Triggers

```
-- test set one for sightings
INSERT INTO SIGHTINGS(name, person, location, sighted) VALUES('A','A','A','A');
UPDATE SIGHTINGS SET name='A' where name='A';
DELETE FROM SIGHTINGS WHERE name='A';
-- test set two for sightings
INSERT INTO SIGHTINGS(name, person, location, sighted) VALUES('B','A','A','A');
UPDATE SIGHTINGS SET name='D' where name='B';
DELETE FROM SIGHTINGS WHERE name='D';
-- test set three for flowers
INSERT INTO FLOWERS(genus, species, comname) VALUES('A','A','A');
UPDATE FLOWERS SET comname='A' where comname='A';
DELETE FROM FLOWERS WHERE comname='A';
-- test set four for flowers
INSERT INTO FLOWERS(genus, species, comname) VALUES('A','A','B');
UPDATE FLOWERS SET comname='D' where comname='B';
DELETE FROM FLOWERS WHERE comname='D';
-- test set five for features
INSERT INTO FEATURES(location, class, latitude, longitude, map, elev) VALUES('A','A','A','A','A','A');
UPDATE FEATURES SET location='A' where location='A';
DELETE FROM FEATURES WHERE location='A';
-- test set six for features
INSERT INTO FEATURES(location, class, latitude, longitude, map, elev) VALUES('B','A','A','A','A','A');
UPDATE FEATURES SET location='D' where location='B';
DELETE FROM FEATURES WHERE location='D';
-- show the executed logs
SELECT * from TABLE_LOGS;
```

Initiated Input done as Transaction: 5 Points

We ran our webpage using the flask library in python. In this library any execution of a query is run as a transaction. So, our webpage runs every user-initiated input as a transaction.