PAP Assignment

Aayush Senapati
PES1UG21CS015

1.Database Name:

database = "company.db"

2.Short Description about the Database:

- Number of Tables: 4 (department, employee, project, projectassign)
- Attributes for Each Table:
  - **Department Table:**
    - id (Primary Key)
    - name
    - location
  - **Employee Table:**
    - id (Primary Key)
    - name
    - deptid (Foreign Key referencing department.id)
    - salary
  - **Project Table:**
    - id (Primary Key)
    - name
    - startdate
    - enddate

        - projectmanagerid (Foreign Key referencing employee.id)
    - **Project Assign Table:**
        - employee_id (Foreign Key referencing employee.id)
        - project_id (Foreign Key referencing project.id)
        - assignment_date
        - end_date
- Primary and Foreign Key Constraints:
    - `department` table: `id` is the primary key.
    - `employee` table: `id` is the primary key, and `deptid` is a foreign key referencing `department.id`.
    - `project` table: `id` is the primary key, and `projectmanagerid` is a foreign key referencing `employee.id`.
    - `projectassign` table: Composite primary key `(employee_id, project_id)`, and both are foreign keys referencing `employee.id` and `project.id`, respectively.

3.

```
import sqlite3

from sqlite3 import Error

import pandas as pd




def create_connection(db_file):

    conn = None
```

```
    try:
        conn = sqlite3.connect(db_file) # creates a persistent database
        conn.execute("PRAGMA foreign_keys = ON") # enable foreign key
constraint
        return conn
    except Error as e:
        print(e)


def display_table(conn, table_name):
    df = pd.read_sql_query(f"SELECT * from {table_name}", conn)
    print(df)
def create_table(conn, create_table_sql):
    try:
        c=conn.cursor()
        c.execute(create_table_sql)
    except Error as e:
        print(e)


def drop_tables(conn):
    c = conn.cursor()
    c.execute("DROP TABLE IF EXISTS projectassign")
    c.execute("DROP TABLE IF EXISTS project")
    c.execute("DROP TABLE IF EXISTS employee")
    c.execute("DROP TABLE IF EXISTS department")
```

```
def insert_data(conn):
    c=conn.cursor()
    for i in range(1, 11):
        c.execute(f"INSERT INTO department (id, name, location) VALUES ({i}, 'Department{i}', 'Location{i}')")

    for i in range(1, 11):
        c.execute(f"INSERT INTO employee (id, name, deptid, salary) VALUES ({i}, 'Employee{i}', {i}, {i*1000.0})")

    for i in range(1, 11):
        c.execute(f"INSERT INTO project (id, name, startdate, enddate, projectmanagerid) VALUES ({i}, 'Project{i}', '2022-01-01', '2022-12-31', {11-i})")

    for i in range(1, 11):
        c.execute(f"INSERT INTO projectassign (employee_id, project_id, assignment_date, end_date) VALUES ({i}, {i}, '2022-01-01', '2022-12-31')")




def main():
```

```
    database = "company.db"


    sql_create_department_table = """ CREATE TABLE IF NOT EXISTS
department (
                                        id integer PRIMARY KEY,
                                        name text NOT NULL,
                                        location text
                                    ); """


    sql_create_employee_table = """ CREATE TABLE IF NOT EXISTS employee (
                                        id integer PRIMARY KEY,
                                        name text NOT NULL,
                                        deptid integer,
                                        salary real,
                                        FOREIGN KEY (deptid)
                                        REFERENCES department (id)
                                        ON DELETE SET NULL
                                        ON UPDATE CASCADE
                                    ); """


    sql_create_project_table = """CREATE TABLE IF NOT EXISTS project (
                                        id integer PRIMARY KEY,
                                        name text NOT NULL,
                                        startdate text,
```

```
                                    enddate text,
                                    projectmanagerid integer,
                                    FOREIGN KEY (projectmanagerid)
REFERENCES employee (id) ON DELETE CASCADE ON UPDATE CASCADE
                                );"""

    sql_create_projectassign_table = """CREATE TABLE IF NOT EXISTS
projectassign (
                                        employee_id integer,
                                        project_id integer,
                                        assignment_date text,
                                        end_date text,
                                        PRIMARY KEY (employee_id,
project_id),
                                        FOREIGN KEY (employee_id)
REFERENCES employee (id) ON DELETE CASCADE ON UPDATE CASCADE,
                                        FOREIGN KEY (project_id)
REFERENCES project (id) ON DELETE CASCADE ON UPDATE CASCADE
                                    );"""

    conn = create_connection(database)

    if conn is not None:
        drop_tables(conn)
        create_table(conn, sql_create_department_table)
```

```
        create_table(conn, sql_create_employee_table)

        create_table(conn, sql_create_project_table)

        create_table(conn, sql_create_projectassign_table)

        insert_data(conn)

        display_table(conn, 'department')

        display_table(conn, 'employee')

        display_table(conn, 'project')

        display_table(conn, 'projectassign')

        conn.execute("UPDATE department SET id = 11 WHERE id = 1")

        conn.execute("UPDATE employee SET id = 11 WHERE id = 2")

        conn.execute("UPDATE project SET id = 11 WHERE id = 3")


print("------------------------------------------------------------------
-------------------------------------------------------")

        print("After update")

        display_table(conn, 'department')

        display_table(conn, 'employee')

        display_table(conn, 'project')

        display_table(conn, 'projectassign')


print("------------------------------------------------------------------
-------------------------------------------------------")

        print("After delete")

        conn.execute("DELETE FROM department WHERE id = 2")

        conn.execute("DELETE FROM employee WHERE id = 3")
```

```
        conn.execute("DELETE FROM project WHERE id = 4")

        display_table(conn, 'department')

        display_table(conn, 'employee')

        display_table(conn, 'project')

        display_table(conn, 'projectassign')

        conn.commit()

        conn.close()

    else:

        print("Error! cannot create the database connection.")


if __name__ == '__main__':

    main()
```

4.

Create Queries:

1. Create `department` Table:

sql

```
CREATE TABLE IF NOT EXISTS department (

    id integer PRIMARY KEY,

    name text NOT NULL,

    location text

);
```

2. Create `employee` Table:

sql

```
CREATE TABLE IF NOT EXISTS employee (
    id integer PRIMARY KEY,
    name text NOT NULL,
    deptid integer,
    salary real,
    FOREIGN KEY (deptid) REFERENCES department (id)
ON DELETE SET NULL ON UPDATE CASCADE
);
```

3. Create `project` Table:

sql

```
CREATE TABLE IF NOT EXISTS project (
    id integer PRIMARY KEY,
    name text NOT NULL,
    startdate text,
    enddate text,
    projectmanagerid integer,
```

```
    FOREIGN KEY (projectmanagerid) REFERENCES
employee (id) ON DELETE CASCADE ON UPDATE CASCADE

);
```

4. Create `projectassign` Table:

sql

```
CREATE TABLE IF NOT EXISTS projectassign (

    employee_id integer,

    project_id integer,

    assignment_date text,

    end_date text,

    PRIMARY KEY (employee_id, project_id),

    FOREIGN KEY (employee_id) REFERENCES employee
(id) ON DELETE CASCADE ON UPDATE CASCADE,

    FOREIGN KEY (project_id) REFERENCES project
(id) ON DELETE CASCADE ON UPDATE CASCADE

);
```

Insert Queries:

5. Insert Data into Tables:

python

```
# (Within the insert_data function)

# Inserts data into the 'department', 'employee',
'project', and 'projectassign' tables.
```

Drop Queries:

6. Drop Tables:

sql

```
-- Drops tables if they exist

DROP TABLE IF EXISTS projectassign;

DROP TABLE IF EXISTS project;

DROP TABLE IF EXISTS employee;

DROP TABLE IF EXISTS department;
```

Update Queries:

7. Update `department` Table:

sql

```
-- Updates the 'department' table, setting the ID
to 11 where ID is currently 1

UPDATE department SET id = 11 WHERE id = 1;
```

8. Update `employee` Table:

sql

```
-- Updates the 'employee' table, setting the ID to
11 where ID is currently 2

UPDATE employee SET id = 11 WHERE id = 2;
```

9. Update `project` Table:

sql

```
-- Updates the 'project' table, setting the ID to
11 where ID is currently 3

UPDATE project SET id = 11 WHERE id = 3;
```

Delete Queries:

10. Delete from `department` Table:

sql

```
-- Deletes the record from the 'department' table
where ID is 2

DELETE FROM department WHERE id = 2;
```

11. Delete from `employee` Table:

sql

```
-- Deletes the record from the 'employee' table
where ID is 3
```

```
DELETE FROM employee WHERE id = 3;
```

12. Delete from `project` Table:

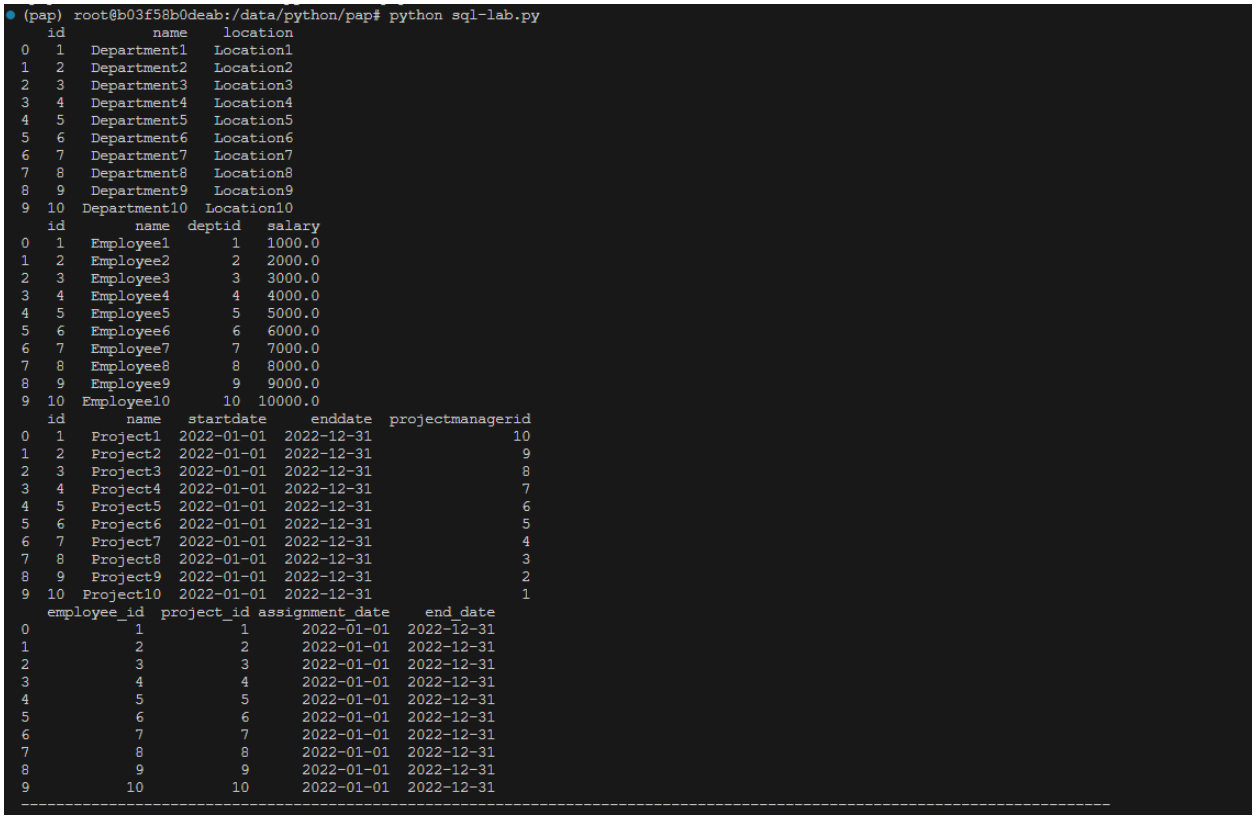
sql

-- Deletes the record from the 'project' table where ID is 4

DELETE FROM project WHERE id = 4;

Screenshots(all queries are run together):

1)After create and insert



2) After update

```
After update
   id         name    location
0   2   Department2   Location2
1   3   Department3   Location3
2   4   Department4   Location4
3   5   Department5   Location5
4   6   Department6   Location6
5   7   Department7   Location7
6   8   Department8   Location8
7   9   Department9   Location9
8  10  Department10  Location10
9  11   Department1   Location1
   id        name  deptid   salary
0   1   Employee1      11   1000.0
1   3   Employee3       3   3000.0
2   4   Employee4       4   4000.0
3   5   Employee5       5   5000.0
4   6   Employee6       6   6000.0
5   7   Employee7       7   7000.0
6   8   Employee8       8   8000.0
7   9   Employee9       9   9000.0
8  10  Employee10      10  10000.0
9  11   Employee2       2   2000.0
   id        name   startdate     enddate  projectmanagerid
0   1   Project1  2022-01-01  2022-12-31                10
1   2   Project2  2022-01-01  2022-12-31                 9
2   4   Project4  2022-01-01  2022-12-31                 7
3   5   Project5  2022-01-01  2022-12-31                 6
4   6   Project6  2022-01-01  2022-12-31                 5
5   7   Project7  2022-01-01  2022-12-31                 4
6   8   Project8  2022-01-01  2022-12-31                 3
7   9   Project9  2022-01-01  2022-12-31                11
8  10  Project10  2022-01-01  2022-12-31                 1
9  11   Project3  2022-01-01  2022-12-31                 8
   employee_id  project_id assignment_date    end_date
0            1           1      2022-01-01  2022-12-31
1           11           2      2022-01-01  2022-12-31
2            3          11      2022-01-01  2022-12-31
3            4           4      2022-01-01  2022-12-31
4            5           5      2022-01-01  2022-12-31
5            6           6      2022-01-01  2022-12-31
6            7           7      2022-01-01  2022-12-31
7            8           8      2022-01-01  2022-12-31
8            9           9      2022-01-01  2022-12-31
9           10          10      2022-01-01  2022-12-31
```

3) After delete

```
After delete
   id          name    location
0   3   Department3   Location3
1   4   Department4   Location4
2   5   Department5   Location5
3   6   Department6   Location6
4   7   Department7   Location7
5   8   Department8   Location8
6   9   Department9   Location9
7  10  Department10  Location10
8  11   Department1   Location1
   id        name  deptid   salary
0   1   Employee1    11.0   1000.0
1   4   Employee4     4.0   4000.0
2   5   Employee5     5.0   5000.0
3   6   Employee6     6.0   6000.0
4   7   Employee7     7.0   7000.0
5   8   Employee8     8.0   8000.0
6   9   Employee9     9.0   9000.0
7  10  Employee10    10.0  10000.0
8  11   Employee2     NaN   2000.0
   id       name   startdate     enddate  projectmanagerid
0   1   Project1  2022-01-01  2022-12-31                10
1   2   Project2  2022-01-01  2022-12-31                 9
2   5   Project5  2022-01-01  2022-12-31                 6
3   6   Project6  2022-01-01  2022-12-31                 5
4   7   Project7  2022-01-01  2022-12-31                 4
5   9   Project9  2022-01-01  2022-12-31                11
6  10  Project10  2022-01-01  2022-12-31                 1
7  11   Project3  2022-01-01  2022-12-31                 8
   employee_id  project_id assignment_date    end_date
0            1           1      2022-01-01  2022-12-31
1           11           2      2022-01-01  2022-12-31
2            5           5      2022-01-01  2022-12-31
3            6           6      2022-01-01  2022-12-31
4            7           7      2022-01-01  2022-12-31
5            9           9      2022-01-01  2022-12-31
6           10          10      2022-01-01  2022-12-31
(pap) root@b03f58b0deab:/data/python/pap#
```