ASSIGNMENT 2

Assignment Date	20/09/2022
Student Name	SATHYA.P
Student Roll No	960519104073
Maximum Marks	2 Marks

1.Create User table with email, user name, roll number, password.

Program:

```
 import sqlite3
from sqlite3 import Error
def sql_connection():
  try:
    conn = sqlite3.connect('mydatabase.db')
    return conn
  except Error:
    print(Error)

def sql_table(conn):
  cursorObj = conn.cursor()
# Create the table
  cursorObj.execute("CREATE TABLE salesman(salesman_id n(5), name char(30), city char(35),
commission decimal(7,2));")
# Insert records
```

```
cursorObj.executescript("""
INSERT INTO salesman VALUES(5001,'James Hoog', 'New York', 0.15);
INSERT INTO salesman VALUES(5002,'Nail Knite', 'Paris', 0.25);
INSERT INTO salesman VALUES(5003,'Pit Alex', 'London', 0.15);
INSERT INTO salesman VALUES(5004,'Mc Lyon', 'Paris', 0.35);
INSERT INTO salesman VALUES(5005,'Paul Adam', 'Rome', 0.45);
""")
conn.commit()
cursorObj.execute("SELECT * FROM salesman")
rows = cursorObj.fetchall()
print("Agent details:")
for row in rows:
    print(row)
sqllite_conn = sql_connection()
sql_table(sqllite_conn)
if (sqllite_conn):
 sqllite_conn.close()
 print("\nThe SQLite connection is closed.")
```

Output:

```
Agent details:
(5001, 'James Hoog', 'New York', 0.15)
(5002, 'Nail Knite', 'Paris', 0.25)
(5003, 'Pit Alex', 'London', 0.15)
(5004, 'Mc Lyon', 'Paris', 0.35)
(5005, 'Paul Adam', 'Rome', 0.45)

The SQLite connection is closed.
```

2. Perform UPDATE, DELETE Queries with user table.

Program:

Source Code

```
from tabulate import tabulate
import mysql.connector


con = mysql.connector.connect(host="localhost", user="root", password="root",
database="python_db")



def insert(name, age, city):
    res = con.cursor()
    sql = "insert into users (name,age,city) values (%s,%s,%s)"
    user = (name, age, city)
    res.execute(sql, user)
    con.commit()
    print("Data Insert Success")


def update(name, age, city,id):
    res = con.cursor()
    sql = "update users set name=%s,age=%s,city=%s where id=%s"
    user = (name, age, city,id)
    res.execute(sql, user)
    con.commit()
    print("Data Update Success")
```

```
def select():
    res = con.cursor()
    sql = "SELECT ID,NAME,AGE,CITY from users"
    res.execute(sql)
    # result=res.fetchone()
    # result=res.fetchmany(2)
    result = res.fetchall()
    print(tabulate(result, headers=["ID", "NAME", "AGE", "CITY"]))



def delete(id):
    res = con.cursor()
    sql = "delete from users where id=%s"
    user = (id,)
    res.execute(sql, user)
    con.commit()
    print("Data Delete Success")



while True:
    print("1.Insert Data")
    print("2.Update Data")
    print("3.Select Data")
    print("4.Delete Data")
    print("5.Exit")
    choice = int(input("Enter Your Choice : "))
```

```
if choice == 1:
    name = input("Enter Name : ")
    age = input("Enter Age : ")
    city = input("Enter City : ")
    insert(name, age, city)
elif choice == 2:
    id = input("Enter The Id : ")
    name = input("Enter Name : ")
    age = input("Enter Age : ")
    city = input("Enter City : ")
    update(name, age, city,id)
elif choice == 3:
    select()
elif choice == 4:
    id = input("Enter The Id to Delete : ")
    delete(id)
elif choice == 5:
    quit()
else:
    print("Invalid Selection . Please Try Again !")
```

Output:

1.Insert Data

2.Update Data

3.Select Data

4.Delete Data

5.Exit

Enter Your Choice : 1

Enter Name : Priya

Enter Age : 21

Enter City : Hosur

Data Insert Success

1.Insert Data

2.Update Data

3.Select Data

4.Delete Data

5.Exit

Enter Your Choice : 2

Enter The Id : 1

Enter Name : Harish

Enter Age : 22

Enter City : Salem

Data Update Success

1.Insert Data

2.Update Data

3.Select Data

4.Delete Data

5.Exit

Enter Your Choice : 4

Enter The Id to Delete : 3

Data Delete Success

1.Insert Data

2.Update Data

3.Select Data

4.Delete Data

5.Exit

Enter Your Choice : 3

```
 ID  NAME     AGE  CITY
---- ------ ----- -------
   1  Harish    22  Salem
   2  Pooja     23  Chennai
   4  Ram       21  Namakkal
   6  Priya     21  Hosur
1.Insert Data
2.Update Data
3.Select Data
4.Delete Data
5.Exit
Enter Your Choice : 5

Process finished with exit code 0
```

3.Connect python code to db2.

```python
import pyodbc
import os

pw = os.environ.get("DB2PW")
user = os.environ.get("DB2USER")

con = pyodbc.connect("DSN=testpython;UID="+user+";PWD="+pw)

print("connected")


cur = con.cursor()

cur.execute("SELECT PROD_NAME from prod")

data = cur.fetchall()

print(data)
```

Output:

```
import pyodbc
import os

pw = os.environ.get("DB2PW")
user = os.environ.get("DB2USER")

con = pyodbc.connect("DSN=testpython;UID="+user+";PWD="+pw)

print("connected")


cur = con.cursor()
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.

Lernen Sie das neue plattformübergreifende PowerShell kennen – https://aka.ms/pscore6

PS C:\Users\Moham\OneDrive\Desktop\brahamcreation2go> python brahamcreation2go.py
connected
[('banana', ), ('apple', ), ('pomegranate', )]
PS C:\Users\Moham\OneDrive\Desktop\brahamcreation2go>
```

4.Create a flask app with registration page, login page and welcome page. By default load the registration page once the user enters all the fields store the data in database and navigate to login page. Authenticate user with username and password. If the user is valid show the welcome page.

Program:

```
# Store this code in 'app.py' file


from flask import Flask, render_template, request, redirect, url_for, session

from flask_mysqldb import MySQL

import MySQLdb.cursors

import re


app = Flask(_name_)
```

```
app.secret_key = 'your secret key'


app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = 'your password'
app.config['MYSQL_DB'] = 'geeklogin'


mysql = MySQL(app)


@app.route('/')
@app.route('/login', methods =['GET', 'POST'])
def login():
    msg = ''
    if request.method == 'POST' and 'username' in request.form and 'password' in request.form:
        username = request.form['username']
        password = request.form['password']
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM accounts WHERE username = % s AND password = % s', (username,
password, ))
        account = cursor.fetchone()
        if account:
            session['loggedin'] = True
            session['id'] = account['id']
            session['username'] = account['username']
            msg = 'Logged in successfully !'
            return render_template('index.html', msg = msg)
    else:
        msg = 'Incorrect username / password !'
```

```
    return render_template('login.html', msg = msg)


@app.route('/logout')
def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('username', None)
    return redirect(url_for('login'))


@app.route('/register', methods =['GET', 'POST'])
def register():
    msg = ''
    if request.method == 'POST' and 'username' in request.form and 'password' in request.form and
'email' in request.form :
        username = request.form['username']
password = request.form['password']
email = request.form['email']
cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
cursor.execute('SELECT * FROM accounts WHERE username = % s', (username, ))
account = cursor.fetchone()
if account:
    msg = 'Account already exists !'
elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
    msg = 'Invalid email address !'
elif not re.match(r'[A-Za-z0-9]+', username):
    msg = 'Username must contain only characters and numbers !'
elif not username or not password or not email:
    msg = 'Please fill out the form !'
else:
```

```
        cursor.execute('INSERT INTO accounts VALUES (NULL, % s, % s, % s)', (username, password, email,
))
        mysql.connection.commit()
        msg = 'You have successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('register.')


<!-- Store this code in 'login.html' file inside the 'templates' folder -->


<html>
    <head>
        <meta charset="UTF-8">
        <title> Login </title>
    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</head>
<body></br></br></br></br></br>
    <div align="center">
     <div align="center" class="border">
        <div class="header">
            <h1 class="word">Login</h1>
        </div></br></br></br>
       <h2 class="word">
            <form action="{{ url_for('login') }}" method="post">
                <div class="msg">{{ msg }}</div>
                <input id="username" name="username" type="text" placeholder="Enter Your Username"
class="textbox"/></br></br>
                <input id="password" name="password" type="password" placeholder="Enter Your
Password" class="textbox"/></br></br></br>
```

```
            <input type="submit" class="btn" value="Sign In"></br></br>
          </form>
        </h2>
        <p class="bottom">Don't have an account?  <a class="bottom" href="{{url_for('register')}}"> Sign Up here</a></p>
      </div>
    </body>
</html>


<!-- Store this code in 'register.html' file inside the 'templates' folder -->

<html>
  <head>
    <meta charset="UTF-8">
    <title> Register </title>
    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</head>
<body></br></br></br></br></br>
    <div align="center">
     <div align="center" class="border">
       <div class="header">
         <h1 class="word">Register</h1>
       </div></br></br></br>
<h2 class="word">
    <form action="{{ url_for('register') }}" method="post">
     <div class="msg">{{ msg }}</div>
```

```
            <input id="username" name="username" type="text" placeholder="Enter Your Username"
class="textbox"/></br></br>

            <input id="password" name="password" type="password" placeholder="Enter Your
Password" class="textbox"/></br></br>

            <input id="email" name="email" type="text" placeholder="Enter Your Email ID"
class="textbox"/></br></br>

                <input type="submit" class="btn" value="Sign Up"></br>

            </form>

          </h2>

          <p class="bottom">Already have an account?  <a class="bottom" href="{{url_for('login')}}"> Sign
In here</a></p>

        </div>

      </div>

   </body>

</html>
```

```
/* Store this code in 'style.css' file inside the 'static' folder*/


        .header{

            padding: 5px 120px;

            width: 150px;

            height: 70px;

            background-color: #236B8E;

        }


        .border{

            padding: 80px 50px;

            width: 400px;

            height: 450px;
```

```
    border: 1px solid #236B8E;
    border-radius: 0px;
    background-color: #9AC0CD;
}

.btn {
    padding: 10px 40px;
    background-color: #236B8E;
    color: #FFFFFF;
    font-style: oblique;
    font-weight: bold;
    border-radius: 10px;
}

.textbox{
    padding: 10px 40px;
    background-color: #236B8E;
    text-color: #FFFFFF;
    border-radius: 10px;
}

::placeholder {
    color: #FFFFFF;
    opacity: 1;
    font-style: oblique;
    font-weight: bold;
}

.word{
```

```
    color: #FFFFFF;
    font-style: oblique;
    font-weight: bold;
}

.bottom{
    color: #236B8E;
    font-style: oblique;
     font-weight: bold;
}
```

Output: