Assignment-2

Team ID	PNT2022TMID34435
Project Name	Nutrition Assistant Application
Maximum Marks	2 Marks
Batch no.	B6-6M2E

Question:

1.Create form of type input text, email, password, radio button text Area, drop down and navigate to success page and display files of form in table (CSS, HTML).

2. For CSS create external style sheet for above task (separate CSS file and link that in html).

3. Create sample program for Flask HTTP methods (list or map and Perform operations of PUT, GET, DELETE and POST.

Apply.html

```
<html>
<body>
<form action="http://localhost:5000/login" method="POST">
<p>Enter Username:</p>
<p><input type="text" name="username" /></p>
<p>Enter Email:</p>
<p><input type="email" name="email" /></p>
<p>Enter Qualification:</p>
<p><input type="text" name="qualification" /></p>
<p>Enter Skill:</p>
<p><input type="text" name="skill" /></p>
<p>Enter Jobs:</p>
<p><input type="text" name="jobs" /></p>
<p><input type="submit" value="submit"></p>
</form>
</body>
</html>
```

Login.html

```
<html>
<body>
<form action="http://localhost:5000/login" method="POST">
<p>Enter Username:</p>
<p><input type="text" name="username" /></p>
<p>Enter Password:</p>
<p><input type="password" name="password" /></p>
<p><input type="submit" value="submit"></p>
</form>
</body>
```

```
</html>
```

Register.html

```
<html>
<body>
<form action="http://localhost:5000/login" method="POST"> <p>Enter Username:</p>
<p><input type="text" name="username" /></p>
<p>Enter Email:</p>
<p><input type="email" name="email" /></p>
<p>Enter Password:</p>
<p><input type="password" name="password" /></p>
<p><input type="submit" value="submit"></p>
</form> </body>
</html>
```

App.py

```
from flask import Flask, render_template, request, redirect, url_for, session import
ibm_db
import re

app = Flask(_name_)

app.secret_key = 'a'

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=b70af05b-76e4-4bca-a1f5-
23dbb4c6a74e.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32716;SECURITY=SSL;SSLServerC
ertificate=DigiCertGlobalRootCA.crt;UID=jzc43091;PWD=PI8VtGRvZlSVT65A",",")

@app.route('/')

def homer():
    return render_template('home.html')


@app.route('/login',methods =['GET', 'POST'])
def login():     global userid     msg = ''     if
request.method == 'POST' :
        username = request.form['username']
password = request.form['password']
        sql = "SELECT * FROM users WHERE username =? AND password=?"
stmt = ibm_db.prepare(conn, sql)         ibm_db.bind_param(stmt,1,username)
ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
print (account)         if account:
            session['loggedin'] = True
            session['id']        =        account['USERNAME']
userid=  account['USERNAME']
```

```
            session['username'] = account['USERNAME']
msg = 'Logged in successfully !'

            msg = 'Logged in successfully !'
            return render_template('dashboard.html', msg = msg)
else:
            msg = 'Incorrect username / password !'
return render_template('login.html', msg = msg)

@app.route('/register', methods =['GET', 'POST']) def
registet():
    msg = ''     if request.method == 'POST'
:          username =
request.form['username']
        email = request.form['email']
password = request.form['password']
        sql = "SELECT * FROM users WHERE username =?"
        stmt = ibm_db.prepare(conn, sql)
ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account =
ibm_db.fetch_assoc(stmt)
print(account)         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 = 'name must contain only characters and numbers !'
else:
            insert_sql = "INSERT INTO  users VALUES (?, ?, ?)"
prep_stmt = ibm_db.prepare(conn, insert_sql)
ibm_db.bind_param(prep_stmt, 1, username)
ibm_db.bind_param(prep_stmt, 2, email)
ibm_db.bind_param(prep_stmt, 3, password)
ibm_db.execute(prep_stmt)
            msg = 'You have successfully registered !'
elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('register.html', msg = msg)

@app.route('/dashboard')
def dash():

    return render_template('dashboard.html')

@app.route('/apply',methods =['GET',
'POST']) def apply():      msg = ''
```

```
     if request.method == 'POST' :
username = request.form['username']
         email = request.form['email']

         qualification= request.form['qualification']
         skills = request.form['skills']
         jobs = request.form['s']
         sql = "SELECT * FROM users WHERE username =?"
         stmt = ibm_db.prepare(conn, sql)
ibm_db.bind_param(stmt,1,username)
         ibm_db.execute(stmt)
         account =
ibm_db.fetch_assoc(stmt)
print(account)          if account:
            msg = 'there is only 1 job position! for you'
return render_template('apply.html', msg = msg)

        insert_sql = "INSERT INTO  job VALUES (?, ?, ?, ?, ?)"
prep_stmt = ibm_db.prepare(conn, insert_sql)
ibm_db.bind_param(prep_stmt, 1, username)
ibm_db.bind_param(prep_stmt, 2, email)
ibm_db.bind_param(prep_stmt, 3, qualification)
ibm_db.bind_param(prep_stmt, 4, skills)
ibm_db.bind_param(prep_stmt, 5, jobs)
ibm_db.execute(prep_stmt)
         msg = 'You have successfully applied for job !'
session['loggedin'] = True
         TEXT = "Hello,a new application for job position" +jobs+"is requested"
elif request.method == 'POST':          msg = 'Please fill out the form !'
     return render_template('apply.html', msg = msg)

@app.route('/display') def
display():
    print(session["username"],session['id'])

    cursor = mysql.connection.cursor()
    cursor.execute('SELECT * FROM job WHERE userid = % s', (session['id'],))
account = cursor.fetchone()
    print("accountdislay",account)



    return render_template('display.html',account = account)

@app.route('/logout')

def logout():
   session.pop('loggedin', None)
```

```
    session.pop('id', None)     session.pop('username', None)
    return render_template('home.html')




if _name_ == '_main_':
    app.run(host='0.0.0.0')
```