

Assignment-2

| | |
|---------------------|-------------------|
| Assignment Date | 15 September 2022 |
| Student Name | Kalaivani T V |
| Student Roll Number | 913319104026 |
| Maximum Marks | 2 Marks |

Question:

- 1) Create a Flask App

App.py

```
from flask import Flask,
url_for,render_template,request,config,session,make_response,abort,redirect,flash;
from markupsafe import escape;
import ibm_db;
conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=32328;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=qzb03901;PWD=fcw2xGW5LDoXVqBI",'','')

app = Flask(__name__);
# app.config['SECRET_KEY'] = ' the secret key'
app.secret_key="Kalaivani";

@app.route("/")
def index():
    return render_template("index.html");

@app.route("/signup/")
def signup():
    return render_template("signup.html")

@app.route("/signin/")
def signin():
    return render_template("signin.html")

@app.route("/about/")
def about():
    return render_template("about.html")
```

```

messages=[{'title':'Message One',
            'content':'Message one content' },
            {'title':'Message Two',
            'content':'Message Two content' },
            ]
@app.route("/chat")
def chat():
    return render_template("chat.html", messages=messages);

@app.route('/create/' , methods=('GET', 'POST'))
def create():
    if request.method=="POST":
        title = request.form['title']
        content = request.form['content']

        if not title:
            flash('Title is required!')
        elif not content:
            flash('Content is required!')
        else:
            messages.append({'title':title,'content':content})
            name = "Bala G"
            return redirect(url_for("chat", messages=name ))
    return render_template("create.html")

@app.route("/database")
def database():
    return render_template("database.html");

@app.route("/addstudent")
def new_student():
    return render_template("add_student.html");

@app.route("/addrec", methods=['GET','POST'])
def addrec():
    if request.method == 'POST':

        name=request.form['name'];
        addr=request.form['address'];
        city=request.form['city'];
        pin=request.form['pin'];

        sql="SELECT * FROM students WHERE name=?"
        stmt=ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,name)
        ibm_db.execute(stmt)
        account=ibm_db.fetch_assoc(stmt)

```

```

        print(account)
        #return "Success..." + name

    if account:
        return render_template("list.html", msg="You are already a
member")
    else:
        insert_sql="INSERT INTO students VALUES(?,?,?,?)"
        prep_stmt=ibm_db.prepare(conn,insert_sql)
        ibm_db.bind_param(prepare_stmt,1,name)
        ibm_db.bind_param(prepare_stmt,2,addr)
        ibm_db.bind_param(prepare_stmt,3,city)
        ibm_db.bind_param(prepare_stmt,4,pin)
        ibm_db.execute(prepare_stmt)
        return render_template("list.html",msg="Student data saved
successfully")

```

```

@app.route('/list')
def list():
    students = []
    sql = "SELECT * FROM Students"
    stmt = ibm_db.exec_immediate(conn, sql)
    dictionary = ibm_db.fetch_both(stmt)
    while dictionary != False:
        # print ("The Name is : ", dictionary)
        students.append(dictionary)
        dictionary = ibm_db.fetch_both(stmt)

    if students:
        return render_template("list.html", students = students)

```

```

@app.route('/delete/<name>')
def delete(name):
    sql = f"SELECT * FROM Students WHERE name='{escape(name)}'"
    print(sql)
    stmt = ibm_db.exec_immediate(conn, sql)
    student = ibm_db.fetch_row(stmt)
    print ("The Name is : ", student)
    if student:
        sql = f"DELETE FROM Students WHERE name='{escape(name)}'"
        print(sql)
        stmt = ibm_db.exec_immediate(conn, sql)

    students = []
    sql = "SELECT * FROM Students"

```

```

stmt = ibm_db.exec_immediate(conn, sql)
dictionary = ibm_db.fetch_both(stmt)
while dictionary != False:
    students.append(dictionary)
    dictionary = ibm_db.fetch_both(stmt)
if students:
    return render_template("list.html", students = students, msg="Delete
successfully")

return "success..."

```

Output:

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  JUPYTER
PS D:\Flask\Flask_with_Database_Connectivity(DB2)> flask --debug run
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 125-669-832

```

2. Add the Home page, About Page

Index.html:

```

{% extends 'base.html' %}
{% block head %}
    <title>Home page</title>
{% endblock %}
{% block body %}
<div class="p-5 mb-4 bg-light rounded-3">
    <div class="container-fluid py-5">
        <h1 class="display-5 fw-bold">Home Page</h1>
        <p class="col-md-8 fs-4">Hii there, This is a Home page.</p>
        <p class="col-md-8 fs-4">All the CSS properties are built using
Bootstrap</p>
        <p class="col-md-8 fs-4">Page routes are built by Jinja2 Template</p>
        <p class="col-md-8 fs-4">Click on about to see about us</p>
        <p class="col-md-8 fs-4">Click on Sign in to enter into the Sign in
form.</p>
        <p class="col-md-8 fs-4">Click on Sign up to enter into the Sign up
form.</p>
        <p class="col-md-8 fs-4">Click on Database Connectivity to perform
database actions</p>
        <p class="col-md-8 fs-4">Click on Create to create a new message.</p>
    </div>
</div>

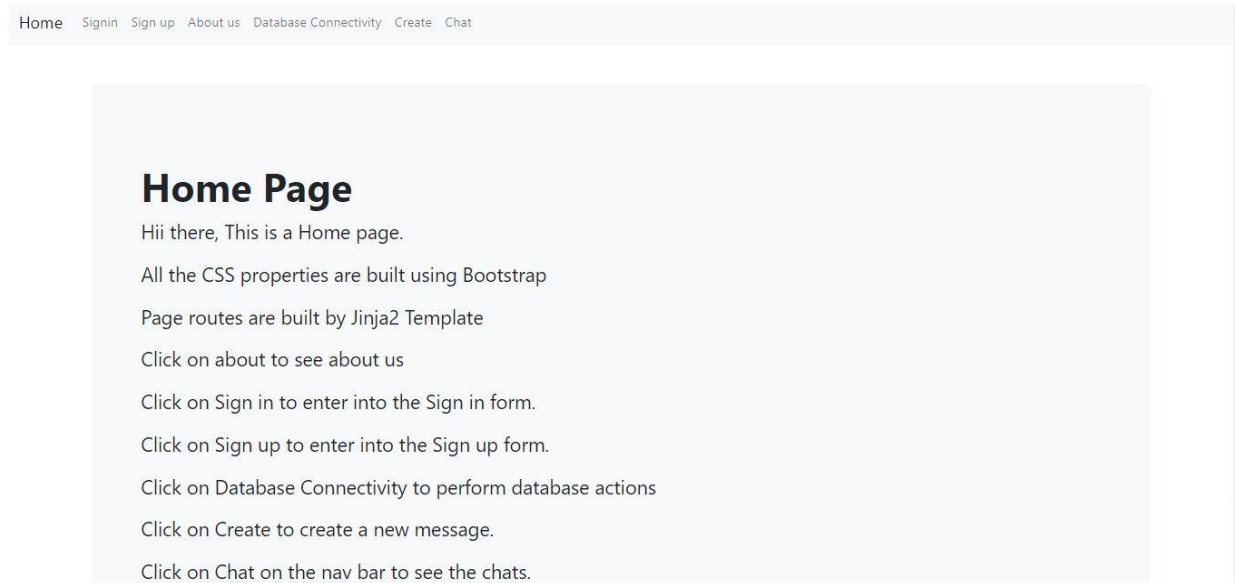
```

```

        <p class="col-md-8 fs-4">Click on Chat on the nav bar to see the
chats.</p>
    </div>
</div>
{% endblock %}

```

Output:



About.html

```

{% extends "base.html" %}
{% block title %}Aboutpage{% endblock %}
{% block body %}
<div class="about-section">
<h1>About Us Page</h1>
<p>Smart Fashion Recommender Application</p>
<p>Balamurugan G(lead)</p>
</div>
<h2 style="text-align:center">Our Team</h2>
<div class="row">
<div class="column">
<div class="card">
<div class="container">
<h2>Catherine S</h2>
<p class="title">designer</p>
<p>teammember1.</p>
<p>catherine@gmail.com</p>
<p><button class="button">Contact</button></p>
</div>
</div>
</div>
<div class="column">
<div class="card">

```

```

<div class="container">
<h2>Kalaivani T V</h2>
<p class="title">frontend developer</p>
<p>team member2</p>
<p>kalai@example.com</p>
<p><button class="button">Contact</button></p>
</div>
</div>
</div>
<div class="column">
<div class="card">
<div class="container">
<h2>Kumaragurumoorthy R</h2>
<p class="title">Backend developer</p>
<p>team member3</p>
<p>guru@example.com</p>
<p><button class="button">Contact</button></p>
</div>
</div>
</div>
</div>
</div>
{% endblock %}

```

Output:

About Us Page

Smart Fashion Recommender Application

Balamurugan G(lead)

Our Team

Catherine S

designer

teammember1.

catherine@gmail.com

Contact

Kalaivani T V

frontend developer

team member2

kalai@example.com

Contact

Kumaragurumoorthy R

Backend developer

team member3

guru@example.com

Contact

3. Add the Bootstrap

base.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <!-- CSS only -->
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
iYQeCzEYFbKjA/T2uDLTPkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT"
crossorigin="anonymous">
  <!-- JavaScript Bundle with Popper -->
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.
js" integrity="sha384-
u10knCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTP00mMi466C8"
crossorigin="anonymous"></script>
  {% block head %} {% endblock %}
</head>
<body>
  <nav class="navbar navbar-expand-lg bg-light">
    <div class="container-fluid">
      <a class="navbar-brand" href="{{url_for('index')}}">Home</a>
      <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target="#navbarSupportedContent" aria-
controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
navigation">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav me-auto mb-2 mb-lg-0">
          <li class="nav-item">
            <a class="nav-link" href="{{url_for('signin')}}">Signin</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="{{url_for('signup')}}">Sign up</a>
```

```

        </li>
        <li class="nav-item">
            <a class="nav-link" href="{{url_for('about')}}">About us</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="{{url_for('database')}}">Database
Connectivity</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="{{url_for('create')}}">Create</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="{{url_for('chat')}}">Chat</a>
        </li>

    </div>
</div>
</nav>

<div class="container mt-5">
    {% block body %} {% endblock%}

</div>
</body>
</html>

```

Output:

[Home](#)
[Signin](#)
[Signup](#)
[About us](#)
[Database Connectivity](#)
[Create](#)
[Chat](#)

Add a New Message

Title

Message Content

Messages

Message One

Message one content

Message Two

Message Two content

hii

how are you

4. Add the Sign in page and Add the Signup Page + database connectivity

Signin.html

```
{% extends 'base.html' %}
{% block head %}
  <title>Signin page</title>
{% endblock %}

{% block body %}
  <h2>Signin forms</h2>
  <form action="/signup">
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" class="form-control"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" class="form-control"><br><br>
    <input type="submit" value="Submit" class="btn btn-primary">
  </form>
{% endblock %}
```

Output:

Signin forms

First name:

Last name:

Submit

Signup.html

```
{% extends 'base.html' %}
<html>
  {% block head %}
    <title>Signup page</title>
  {% endblock %}

  {% block body %}
    <h2>Signup forms</h2>
    <form action="/signup">
      <label for="fname">First name:</label><br>
      <input type="text" id="fname" name="fname" class="form-control"><br>
      <label for="lname">Last name:</label><br>
      <input type="text" id="lname" name="lname" class="form-control"><br><br>
      <label for="pswd">Password:</label><br>
      <input type="text" id="pswd" name="pswd" class="form-control"><br><br>
      <label for="Rpswd">Re-type Password</label><br>
      <input type="text" id="Rpswd" name="Rpswd" class="form-control"><br><br>
      <input type="submit" value="Submit" class="btn btn-primary">
    </form>
  {% endblock %}
</html>
```

Output:

[Home](#) [Signin](#) [Sign up](#) [About us](#) [Database Connectivity](#) [Create](#) [Chat](#)

Signup forms

First name:

Last name:

Password:

Re-type Password

Create Your Account

| | |
|--------------------------|--|
| Enter Name : | <input type="text" value="Enter user here"/> |
| Enter Email ID : | <input type="text" value="Enter Email ID here"/> |
| Enter Username : | <input type="text" value="Enter Username here"/> |
| Enter Password : | <input type="password" value="Enter Password here"/> |
| Enter Confirm Password : | <input type="password" value="Enter Password here"/> |

[Clear Form](#) [Create Account](#)

Database connectivity:

App.py(Database part)

```
from flask import Flask,
url_for,render_template,request,config,session,make_response,abort,redirect,fla
sh;
from markupsafe import escape;
import ibm_db;
conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2d46b6b4-cbf6-40eb-bbce-
6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=32328;SECURITY
=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=qzb03901;PWD=fcw2xGW5LDo
XVqBI",'','')

app = Flask(__name__);
# app.config['SECRET_KEY'] = ' the secret key'
app.secret_key="Kalaivani";

@app.route("/")
def index():
    return render_template("index.html");

@app.route("/signup/")
def signup():
    return render_template("signup.html")

@app.route("/signin/")
def signin():
    return render_template("signin.html")

@app.route("/about/")
def about():
    return render_template("about.html")

messages=[{'title':'Message One',
            'content':'Message one content' },
            {'title':'Message Two',
            'content':'Message Two content' },
            ]

@app.route("/chat")
```

```

def chat():
    return render_template("chat.html", messages=messages);

@app.route('/create/' , methods=('GET', 'POST'))
def create():
    if request.method=="POST":
        title = request.form['title']
        content = request.form['content']

        if not title:
            flash('Title is required!')
        elif not content:
            flash('Content is required!')
        else:
            messages.append({'title':title,'content':content})
            name = "Bala G"
            return redirect(url_for("chat", messages=name ))
    return render_template("create.html")

@app.route("/database")
def database():
    return render_template("database.html");

@app.route("/addstudent")
def new_student():
    return render_template("add_student.html");

@app.route("/addrec", methods=['GET','POST'])
def addrec():
    if request.method == 'POST':

        name=request.form['name'];
        addr=request.form['address'];
        city=request.form['city'];
        pin=request.form['pin'];

        sql="SELECT * FROM students WHERE name=?"
        stmt=ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,name)
        ibm_db.execute(stmt)
        account=ibm_db.fetch_assoc(stmt)
        print(account)
        #return "Success..." +name

        if account:
            return render_template("list.html",msg="You are already a
member")

```

```

        else:
            insert_sql="INSERT INTO students VALUES(?,?,?,?)"
            prep_stmt=ibm_db.prepare(conn,insert_sql)
            ibm_db.bind_param(prepare_stmt,1,name)
            ibm_db.bind_param(prepare_stmt,2,addr)
            ibm_db.bind_param(prepare_stmt,3,city)
            ibm_db.bind_param(prepare_stmt,4,pin)
            ibm_db.execute(prepare_stmt)
            return render_template("list.html",msg="Student data saved
successfully")

```

```

@app.route('/list')
def list():
    students = []
    sql = "SELECT * FROM Students"
    stmt = ibm_db.exec_immediate(conn, sql)
    dictionary = ibm_db.fetch_both(stmt)
    while dictionary != False:
        # print ("The Name is : ", dictionary)
        students.append(dictionary)
        dictionary = ibm_db.fetch_both(stmt)

    if students:
        return render_template("list.html", students = students)

```

```

@app.route('/delete/<name>')
def delete(name):
    sql = f"SELECT * FROM Students WHERE name='{escape(name)}'"
    print(sql)
    stmt = ibm_db.exec_immediate(conn, sql)
    student = ibm_db.fetch_row(stmt)
    print ("The Name is : ", student)
    if student:
        sql = f"DELETE FROM Students WHERE name='{escape(name)}'"
        print(sql)
        stmt = ibm_db.exec_immediate(conn, sql)

        students = []
        sql = "SELECT * FROM Students"
        stmt = ibm_db.exec_immediate(conn, sql)
        dictionary = ibm_db.fetch_both(stmt)
        while dictionary != False:
            students.append(dictionary)
            dictionary = ibm_db.fetch_both(stmt)
        if students:

```

```
        return render_template("list.html", students = students, msg="Delete
successfully")

    return "success..."
```

database.html

```
{% extends 'base.html' %}
{%block body%}
    <h1>Welcome to Student DB</h1>
    <a href="/addstudent">Add New Student</a>
    <a href="/list">List Student</a>
{%endblock%}
</body>
</html>
```

addstudent.html

```
{% extends 'base.html' %}
    <a href="/">Home</a>
    <a href="/addstudent">Add New Student</a>
    <a href="/list">List Student</a>
    <a href="/signup">List Student</a>
    <hr>

{%block body%}
    <form action="{{url_for('addrec')}}" method="POST">
        <h3>Student Information</h3>
        Name<br>
        <input type="text" name="name"/><br>

        Address<br>
        <input type="textarea" name="address"/><br>

        City<br>
        <input type="text" name="city"/><br>

        Pin<br>
        <input type="text" name="pin"/><br>
        <input type="submit" value="submit"/><br>
    </form>
{%endblock%}
</html>
```

list.html

```
{% extends 'base.html' %}
    {%block body%}
        <a href="/">Home</a>
```

```

<a href="/addstudent">Add New Student</a>
<a href="/list">List Student</a>
<br><hr>

{{msg}}

<table class="table table-hover">
  <thead>
    <td scope="col">Name</td>
    <td scope="col">Address</td>
    <td scope="col">City</td>
    <td scope="col">Pincode</td>
  </thead>

  {%for row in students%}
    <tr>
      <td scope="row">{{row["NAME"]}}</td>
      <td scope="row">{{row["ADDRESS"]}}</td>
      <td scope="row">{{row["CITY"]}}</td>
      <td scope="row">{{row["PIN"]}}</td>
      <td scope="row"><a
href="/delete/{{row['NAME']}}">Delete</a></td>
    </tr>
  {%endfor%}
</table>

{%endblock%}

```

Output:

[Home](#)
[Signin](#)
[Signup](#)
[About us](#)
[Database Connectivity](#)
[Create](#)
[Chat](#)

Welcome to Student DB

[Add New Student](#)
[List Student](#)

Student Information

Name
Dhivya
Address
london
City
london
Pin
9787
submit

[Home](#) [Add New Student](#) [List Student](#)

| Name | Address | City | Pincode | |
|-----------|----------|---------|------------|------------------------|
| kalaivani | Madurai | Madurai | 2354 | Delete |
| kaviya | arsnrwer | Madurai | 5476457647 | Delete |
| kumar | vgscytvs | delhi | 678 | Delete |
| catherine | madurai | madurai | 890 | Delete |
| janu | mumbai | mumbai | 5678 | Delete |
| Dhivya | london | london | 9787 | Delete |

Records in IBM DB2:

IBM Db2 on Cloud

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

QZB03901.STUDENTS

Back

Export to CSV

| NAME | ADDRESS | CITY | PIN |
|-----------|----------|---------|------------|
| Dhivya | london | london | 9787 |
| catherine | madurai | madurai | 890 |
| janu | mumbai | mumbai | 5678 |
| kalaivani | Madurai | Madurai | 2354 |
| kaviya | arsnrwer | Madurai | 5476457647 |
| kumar | vgscytvs | delhi | 678 |

