

ASSIGNMENT-2

TEAM ID: PNT2022TMID14431

- (1) Write a flask program which should display Name, Email, and phone and it should display the same details once we hit submit.

```
ASSIGNMENT2(1)
Template
  display.html
  index.html
  app.py

Template > index.html > html > body > form > div > label
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>IBM</title>
8
9 </head>
10 <body>
11   <form action="/display" method="post">
12     <div>
13       <h1>ASSIGNMENT-2(1)</h1>
14     </div>
15     <div>
16       <label for="name">Name:</label>
17       <input type="text" name="name" id="name"><br><br>
18     </div>
19     <div>
20       <label for="email">Email:</label>
21       <input type="email" name="email" id="email"><br><br>
22     </div>
23     <div>
24       <label for="phone">Phone Number:</label>
25       <input type="text" name="phone" id="phone"><br><br>
26     </div>
27     <button type="submit">submit</button>
28   </form>
29 </body>
30 </html>

ASSIGNMENT2(1)
Template
  display.html
  index.html
  app.py

Template > display.html > html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Assignment</title>
8
9 </head>
10 <body>
11   <div>
12     <h1>SUBMISSION DETAILS</h1>
13     <table>
14       <tr>
15         <td>Name</td>
16         <td>{{name}}</td>
17       </tr>
18       <tr>
19         <td>Email</td>
20         <td>{{email}}</td>
21       </tr>
22       <tr>
23         <td>Phone Number</td>
24         <td>{{phone}}</td>
25       </tr>
26     </table>
27   </div>
28 </body>
29 </html>
```

ASSIGNMENT2(1)

Template

display.html

index.html

app.py

app.py > ...

```
1 from flask import Flask, render_template,request
2 app=Flask(__name__, template_folder='template')
3 @app.route('/')
4 @app.route('/index')
5 def homepage():
6     return render_template('index.html')
7 @app.route("/display",methods=['POST','GET'])
8 def register():
9     if request.method=='POST':
10         a = request.form.get('name')
11         b = request.form.get('email')
12         c = request.form.get('phone')
13         return render_template('display.html',name=a,email=b,phone=c)
14 if __name__=="__main__":
15     app.run(debug=True)
16
```

IBM x IBM x Assignments-CApD/Assignment. x +
← → ↻ 127.0.0.1:5000

ASSIGNMENT-2(1)

Name:

Email:

Phone Number:

SUBMISSION DETAILS

Name xxxxxxxxxxxx

Email user@gmail.com

Phone Number 9876543210

(2) Write a flask program which should cover cookies, sessions

The image shows a VS Code editor with a Python file named `app.py` and its web browser preview. The code implements a Flask application that uses cookies and sessions to manage user state.

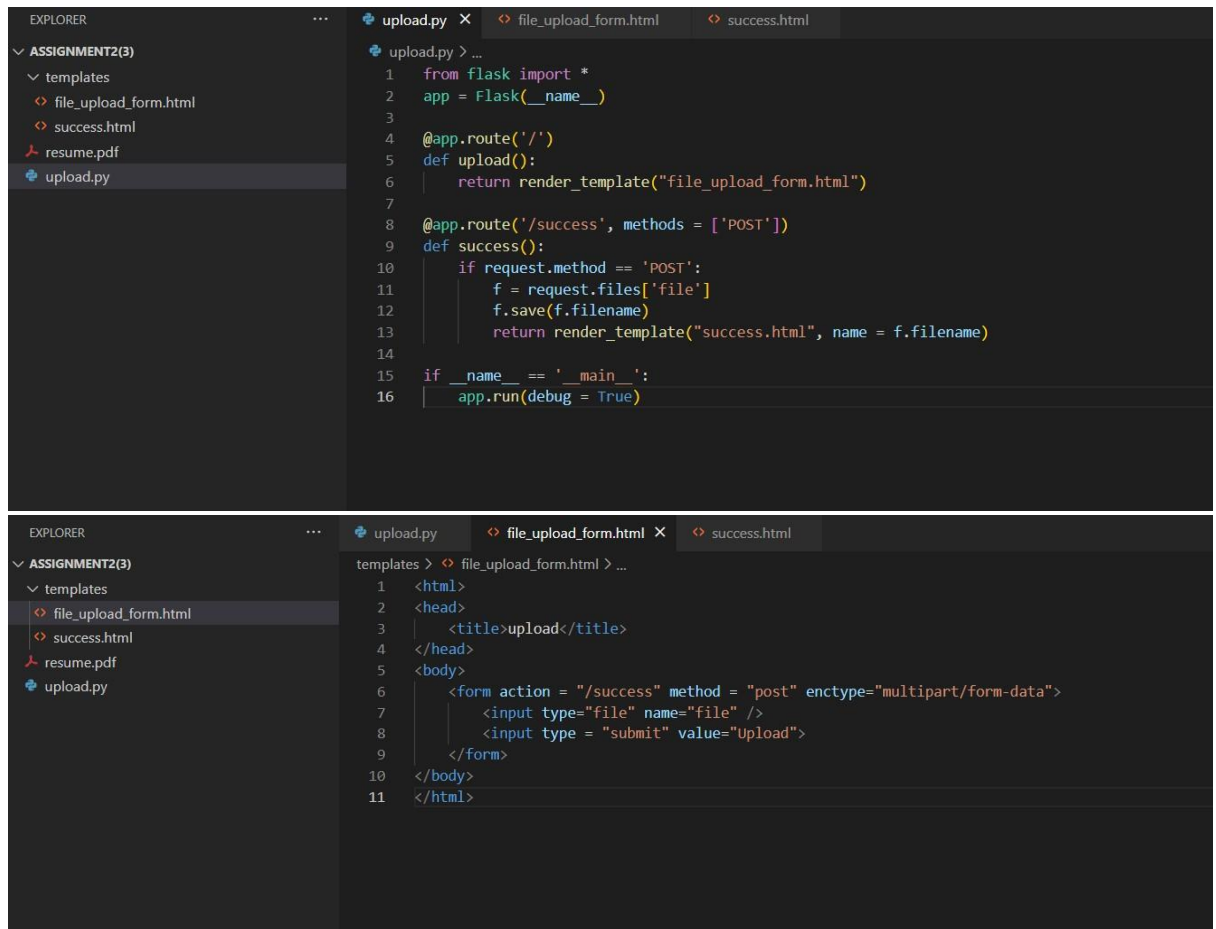
```
1 from flask import render_template, make_response, request, Flask, redirect, url_for, session
2 import os
3 app = Flask(__name__, template_folder='templates')
4 app.secret_key = os.urandom(24)
5 @app.route('/')
6 @app.route('/index')
7 @app.route('/output', methods=['POST'])
8 def index():
9     if not (session.get('user')):
10         if not request.cookies.get('choice'):
11             if request.method == 'POST':
12                 colour = str(request.form['colour'])
13                 response = make_response("Setting your Cookie and Session! Please refresh the page to see the results - ")
14                 response.set_cookie('choice', colour)
15                 session['user'] = str(request.form['name'])
16                 return response
17             else:
18                 response = make_response(render_template('output.html', cookie = str(request.cookies.get('choice')), user = (str(session.get('user')))))
19                 return redirect(url_for('output'))
20         else:
21             return render_template('output.html', cookie = str(request.cookies.get('choice')), user = (str(session.get('user'))))
22     return render_template('index.html')
23
24 if __name__ == "__main__":
25     app.run(debug = True)
```

The web browser preview shows the application running at `127.0.0.1:5000`. It displays a form with the following elements:

- A text input field labeled "Enter userID" with the value "xxxxx".
- A dropdown menu labeled "Select your favourite colour:" with the value "Blue".
- A "Login" button.

The browser's developer console shows a warning: "WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead." and a message: "Running on http://127.0.0.1:5000".

(3) Write a flask program which should display resume details and also have upload resume option by using file uploading.



The image displays two screenshots of a VS Code editor interface, showing the implementation of a Flask application for file uploads.

Top Screenshot: Python Code (upload.py)

```
upload.py > ...
1  from flask import *
2  app = Flask(__name__)
3
4  @app.route('/')
5  def upload():
6      return render_template("file_upload_form.html")
7
8  @app.route('/success', methods = ['POST'])
9  def success():
10     if request.method == 'POST':
11         f = request.files['file']
12         f.save(f.filename)
13         return render_template("success.html", name = f.filename)
14
15 if __name__ == '__main__':
16     app.run(debug = True)
```

Bottom Screenshot: HTML Template (file_upload_form.html)

```
templates > file_upload_form.html > ...
1  <html>
2  <head>
3      <title>upload</title>
4  </head>
5  <body>
6      <form action = "/success" method = "post" enctype="multipart/form-data">
7          <input type="file" name="file" />
8          <input type = "submit" value="Upload">
9      </form>
10 </body>
11 </html>
```

EXPLORER

ASSIGNMENT2(3)

templates

file_upload_form.html

success.html

resume.pdf

upload.py

OUTLINE

TIMELINE

upload.py

file_upload_form.html

success.html

templates

success.html

html

body

form.resume

div

ul

li

```
1 <html>
2 <head>
3 <title>success</title>
4 <style>
5     .resume{
6         width:600px;
7         padding:10px;
8         border: 3px solid black;
9     }
10    .header{
11        text-align: center;
12        font-size: 20px;
13    }
14 </style>
15 </head>
16 <body>
17 <h1>File uploaded successfully</h1>
18 <p>File Name: {{name}}</p>
19 <form class="resume">
20 <div class="header">
21     <p><b>Name:</b>XXXXXXXX</p>
22     <p><b>Mobile Number:</b>9876543210</p>
23     <p><b>Email:</b>xxx@gmail.com</p>
24     <hr>
25 </div>
26 <div>
27     <h2>Objective</h2>
28     <p>I am seeking a role in a company where I
29     can contribute my ideas and be
30     mentored towards a successful career.</p><hr>
31 </div>
32 <div>
33     <h2>Education</h2>
34     <p>2019-2023</p>
35     <p>Bachelor of Engineering(CSE)</p>
36     <p>Saranathan College of Engineering</p>
37     <p><b>CGPA:</b>9.2</p><hr>
```

EXPLORER

ASSIGNMENT2(3)

templates

file_upload_form.html

success.html

resume.pdf

upload.py

OUTLINE

TIMELINE

upload.py

file_upload_form.html

success.html

templates

success.html

html

body

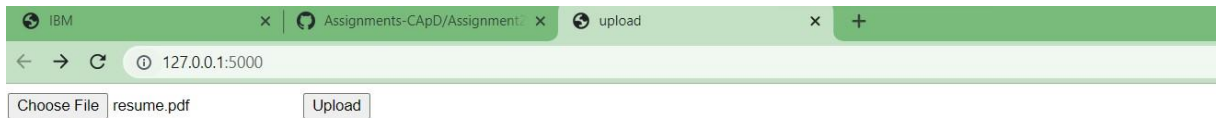
form.resume

div

ul

li

```
38 </div>
39 <div>
40     <h2>Technical Skills</h2>
41     <ul>
42         <li>Python</li>
43         <li>C and C++</li>
44         <li>HTML and CSS</li>
45     </ul><hr>
46 </div>
47 <div>
48     <h2>Personal Traits</h2>
49     <ul>
50         <li>Team work</li>
51         <li>Adaptability</li>
52         <li>Time Management</li>
53     </ul><hr>
54 </div>
55 <div>
56     <h2>Certifications</h2>
57     <ul>
58         <li>Certified in Programming in C and C++ from Internshala
59         </li>
60         <li>Certified in Python from Great Learning</li></ul>
61 </div>
62 </form>
63 </body>
64 </html>
```



File uploaded successfully

File Name: resume.pdf

Name: XXXXXXXX
Mobile Number: 9876543210
Email: xxx@gmail.com
<hr/>
Objective
I am seeking a role in a company where I can contribute my ideas and be mentored towards a successful career.
<hr/>
Education
2019-2023
Bachelor of Engineering(CSE)
Saranathan College of Engineering
CGPA: 9.2
<hr/>
Technical Skills
<ul style="list-style-type: none">• Python• C and C++• HTML and CSS
<hr/>
Personal Traits
<ul style="list-style-type: none">• Team work• Adaptability• Time Management
<hr/>
Certifications
<ul style="list-style-type: none">• Certified in Programming in C and C++ from Internshala• Certified in Python from Great Learning