

Assignment-2

verify the login through postman

login.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="width=device-width,initial-
scale=1"/>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.
min.css">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.j
s"></script>
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.mi
n.js"></script>
    <title>Login Page</title>
  </head>
  <body style="background-color: aliceblue;">
    <nav style="color: blue;background-color: aqua;"
class="navbar navbar-default">
      <div class="container-fluid">
        <div class="navbar-header">
          <a class="navbar-brand" href="">Full Stack
Demo</a>
        </div>
        <ul class="nav navbar-nav">
          <li><a href="">Home</a></li>
          <li><a href="/">Register</a></li>
          <li class="active"><a
href="/login">Login</a></li>
        </ul>
      </div>
    </nav>
    <center>
      <div class="col-sm-4"></div>
      <div class="container col-md-4">
        <h1>Login Page</h1> <br/>
        <form action="/logindata" method="post">
          <div class="form-group">
            <label for="mobile">Mobile No:</label>
            <input type="text" class="form-control"
name="mobile" placeholder="Enter Mobile No"/>
          </div>
          <div class="form-group">
```

```

        <label for="password">Password:</label>
        <input type="password" class="form-control"
name="password" placeholder="Enter password"/>
    </div>
    <div class="form-group">
        <button class="btn btn-primary"
type="submit">Login</button>
    </div>
</form>
<span style="color:red">{{err1}}</span>
<span style="color:green">{{res1}}</span>
</div>

<div class="col-sm-4"></div>
</center>
<a href="https://www.Ammu.com">Ammu</a>

</body>
</html>

```

In the login.html ,we define that login page to active.

```
<li class="active"><a href="/login">Login</a></li>
```

And then,we create a handler “/loginpage” in the “app.py”.

app.py

```

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

from pymongo import MongoClient
client=MongoClient('127.0.0.1',27017)
db=client['Ammu']
c=db['flaskapp']
app=Flask(__name__)
app.secret_key='Ammu'
@app.route('/')
def homepage():
    return render_template('index.html')

@app.route('/login')
def loginpage():
    return render_template('login.html')
@app.route('/dashboard')
def dashboardpage():
    owner=session['username']
    return render_template('dashboard.html',owner=owner)
@app.route('/insertnotes')
def insertnotespage():
    return render_template('insertnotes.html')
@app.route('/viewnotes')
def viewnotes():

```

```

c1=db['notes']
data=[]
for i in c1.find():
    if i['owner']==session['username']:
        data.append(i['notes'])
return render_template('viewnotes.html',data=data,l=len(data))
@app.route('/formdata',methods=['post'])
def formdata():
    name=request.form['name']
    email=request.form['email']
    mobile=request.form['mobile']
    password=request.form['password']
    print(name,email,mobile,password)
    for i in c.find():
        if i['mobile']==mobile:
            return render_template('index.html',err='you have already
registered')
    k={}
    k['name']=name
    k['email']=email
    k['mobile']=mobile
    k['password']=password
    c.insert_one(k)
    return render_template('index.html',res='you have registered
successfully')
@app.route('/logindata',methods=['post'])
def logindata():
    mobile=request.form['mobile']
    password=request.form['password']
    print(mobile,password)
    for i in c.find():
        if i['mobile']==mobile and i['password']==password:
            session['username']=mobile
            return redirect('/dashboard')
            # return render_template('login.html',res1='valid
credentials')

    return render_template('login.html',err1='invalid credentials')

@app.route('/logout')
def logout():
    session['username']=None
    return redirect('/')

@app.route('/insertnotesdata',methods=['post'])
def insertnotesdata():
    notes=request.form['notes']
    owner=session['username']

```

```

print(notes,owner)
c1=db['notes']
k={}
k['owner']=owner
k['notes']=notes
c1.insert_one(k)

return render_template('insertnotes.html',res2='Notes saved')

if __name__=="__main__":
    app.run(host='0.0.0.0',port=5001,debug=True)

```

Output:

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS D:\s_internship\register page-save> & C:/Users/ammu4/AppData/Local/Programs/Python/Python311/python.exe "d:/s_i
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server inst
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5001
* Running on http://192.168.145.67:5001
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 124-491-721

```

Copy the link and paste it on the postman tool app, and select the post method and enter mobil,password...click on send button..

It shows that “valid credentials”

The screenshot shows a Postman interface for a POST request to `127.0.0.1:5001/logindata`. The request body is set to `form-data` with the following parameters:

Key	Value
mobile	9347244193
password	abc123

The response status is `200 OK` with a time of `43 ms` and a size of `2.46 KB`. The response body is displayed in HTML format, showing a login form structure. A span element indicates `valid credentials` in green text.

```

37 </div>
38 <div class="form-group">
39 <button class="btn btn-primary" type="submit">Login</button>
40 </div>
41 </form>
42 <span style="color:red"></span>
43 <span style="color:green">valid credentials</span>
44 </div>
45
46 <div class="col-sm-4"></div>
47 </center>
48 <a href="https://www.Ammu.com">Ammu</a>
49 </body>
50
51 </html>

```

If the login details are incorrect,then it displays that “invalid credentials”

127.0.0.1:5001/logindata

Add to collection

POST

127.0.0.1:5001/logindata

Send

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

	Key	Value	...	Bulk Edit
<input checked="" type="checkbox"/>	mobile	9347244193		
<input checked="" type="checkbox"/>	password	ab67y		
	Key	Value		

Body

Cookies (1)

Headers (5)

Test Results

Status: 200 OK

Time: 21 ms

Size: 2.34 KB

Save Response

Pretty

Raw

Preview

Visualize

HTML

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

</div>
<div class="form-group">
<button class="btn btn-primary" type="submit">Login</button>
</div>
</form>
invalid credentials

</div>
<div class="col-sm-4"></div>
</center>
Ammu
</body>
</html>