Assignment -2

Assignment Date	10 August 2022
Student Name	Suriya.N
Student Roll Number	211419205164
Maximum Marks	2 Marks

Question-1:

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 username and password. If the user is valid show the welcome page.

#app.py

```
from flask import Flask, render_template

app = Flask(_name_,template_folder='template')

@app.route("/login")

def hello_world():

   return render_template('login.html')

@app.route("/index")

def hello():

   return render_template ('bootstrap.html')

@app.route("/home")

def home():

   return "<p>hello,world!</p>"

@app.route("/about")

def about():

   return "<p>hello about homepage</p>"
```

#Login.html

```
{% extends 'bootstrap.html' %}

{% block head %}

<title>login page</title>

{% endblock %}
```

{% block body %}

```
<form action="action_page.php" style="border:1px solid #ccc">

<div class="container">

<h1>Sign Up</h1>

<p>Please fill in this form to create an account.</p>

<hr>


<label for="email"><b>Email</b></label>

<input type="text" placeholder="Enter Email" name="email" required>

<label for="psw"><b>Password</b></label>

<input type="password" placeholder="Enter Password" name="psw" required>

<label for="psw-repeat"><b>Repeat Password</b></label>

<input type="password" placeholder="Repeat Password" name="psw-repeat" required>

<label>

<input type="checkbox" checked="checked" name="remember" style="margin-bottom:15px">
Remember me

</label>

<p>By creating an account you agree to our <a href="#" style="color:dodgerblue">Terms &
Privacy</a>.</p>

<div class="clearfix">

<button type="button" class="cancelbtn">Cancel</button>

<button type="submit" class="signupbtn">Sign Up</button>

</div>

</div>

</form>
```

  {% endblock %}

#login.css

```
* {box-sizing: border-box}


/* Full-width input fields */

  input[type=text], input[type=password] {
```

```css
  width: 100%;

  padding: 15px;

  margin: 5px 0 22px 0;

  display: inline-block;

  border: none;

  background: #f1f1f1;

}


input[type=text]:focus, input[type=password]:focus {

  background-color: #ddd;

  outline: none;

}


hr {

  border: 1px solid #f1f1f1;

  margin-bottom: 25px;

}


/* Set a style for all buttons */
button {

  background-color: #04AA6D;

color: white;

  padding: 14px 20px;

  margin: 8px 0;

  border: none;

  cursor: pointer;

  width: 100%;

  opacity: 0.9;

}


button:hover {
```

```css
    opacity:1;

}


/* Extra styles for the cancel button */

.cancelbtn {

  padding: 14px 20px;

  background-color: #f44336;

}


/* Float cancel and signup buttons and add an equal width */

.cancelbtn, .signupbtn {

  float: left;

  width: 50%;

}


/* Add padding to container elements */

.container {

  padding: 16px;

}


/* Clear floats */

.clearfix::after {

  content: "";

  clear: both;

  display: table;

}


/* Change styles for cancel button and signup button on extra small screens */

@media screen and (max-width: 300px) {

.cancelbtn, .signupbtn {

    width: 100%;
```

```
  }
}
```

```html
<!DOCTYPE html>
<html lang="en">

<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content=
     "width=device-width, initial-scale=1">

<link rel="stylesheet" type="text/css"
    href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css">

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"></script>
   {% block head %} {% endblock %}
</head>

<body>
<div class="container">
     {% block body %} {% endblock %}
</div>
</body>

</html>
```

```
Command Prompt

Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Lenovo>flask --version
Python 3.10.7
Flask 2.2.2
Werkzeug 2.2.2

C:\Users\Lenovo>
```
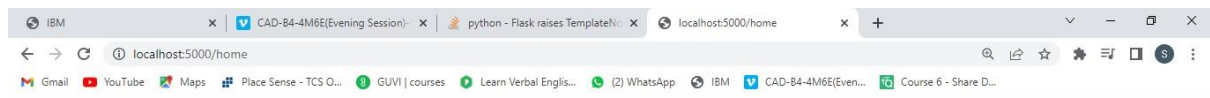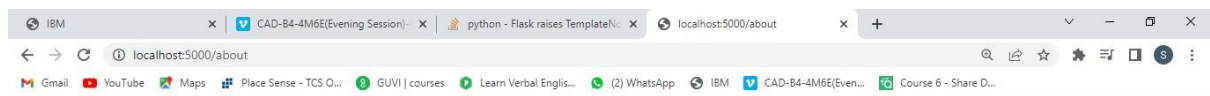


```
Command Prompt - flask --debug run

D:\flask>cd flask_1

D:\flask\flask_1>flask run
 * Debug mode: off
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
127.0.0.1 - - [24/Sep/2022 15:19:36] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [24/Sep/2022 15:19:39] "GET /favicon.ico HTTP/1.1" 404 -
[2022-09-24 15:20:03,224] ERROR in app: Exception on /index [GET]
Traceback (most recent call last):
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\flask\app.py", line 2525, in wsgi_app
    response = self.full_dispatch_request()
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\flask\app.py", line 1822, in full_dispatch_request
    rv = self.handle_user_exception(e)
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\flask\app.py", line 1820, in full_dispatch_request
    rv = self.dispatch_request()
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\flask\app.py", line 1796, in dispatch_request
    return self.ensure_sync(self.view_functions[rule.endpoint])(**view_args)
  File "D:\flask\flask_1\app.py", line 11, in hello
    return render_template ('index.html')
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\flask\templating.py", line 146, in render_template
    template = app.jinja_env.get_or_select_template(template_name_or_list)
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\jinja2\environment.py", line 1081, in get_or_select_template
    return self.get_template(template_name_or_list, parent, globals)
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\jinja2\environment.py", line 1010, in get_template
    return self._load_template(name, globals)
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\jinja2\environment.py", line 969, in _load_template
    template = self.loader.load(self, name, self.make_globals(globals))
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\jinja2\loaders.py", line 126, in load
    source, filename, uptodate = self.get_source(environment, name)
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\flask\templating.py", line 62, in get_source
    return self._get_source_fast(environment, template)
  File "C:\Users\Lenovo\AppData\Local\Programs\Python\Python310\lib\site-packages\flask\templating.py", line 98, in _get_source_fast
    raise TemplateNotFound(template)
jinja2.exceptions.TemplateNotFound: index.html
127.0.0.1 - - [24/Sep/2022 15:20:03] "GET /index HTTP/1.1" 500 -

D:\flask\flask_1>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
```
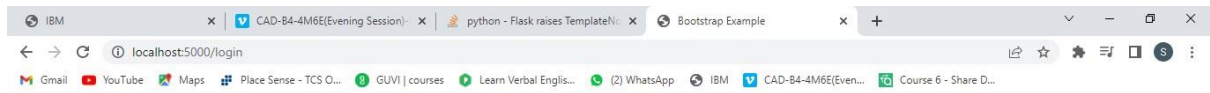
hello,world!

hello about homepage

IBM   ×   CAD-B4-4M6E(Evening Session)- ×   python - Flask raises TemplateN ×   Bootstrap Example   ×   +

localhost:5000/login

Gmail   YouTube   Maps   Place Sense - TCS O...   GUVI | courses   Learn Verbal Englis...   (2) WhatsApp   IBM   CAD-B4-4M6E(Even...   Course 6 - Share D...

Sign Up

Please fill in this form to create an account.

Email [Enter Email]   **Password** [Enter Password]   **Repeat Password** [Repeat Password]   ☑ Remember me

By creating an account you agree to our [Terms & Privacy](#).

[Cancel] [Sign Up]