CREATING A FLASK:

AIM :

To create a flask and add the home page about page and add the signup and singin button in the web page using flask

Requirements:

- Install python
- Install pip & Flask

Source code :

Python code:

app.py

```
from flask import Flask, redirect, url_for ,render_template


app = Flask(__name__)


@app.route("/")

def index():

   return render_template("index.html")


@app.route("/home")

def home():

   return render_template("home.html")


@app.route("/about")

def about():

   return render_template("about.html")


@app.route("/signin")

def signin():
```

```
    return render_template("signin.html")


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

HTML:

Index.html

```
<html>
   <head>
      <title>INDEX</title>
   </head>
   <body>
      <p> Welcome to the WEB PAGE</p>
      <a href="/home">HOME</a>
      <a href="/about">ABOUT</a>
     <button type="button" name="singup"> <a href="/singup"> SING UP</a></button>
     <button type="button" name="singin"> <a href="/signin">SIGN IN</a></button>
   </body>
</html>
```

Home.html

```
<html>
<head>
   <title>HOME PAGE</title>
   <body>
      <P> HI, THIS IS GOKUL RAJ S</P>
      <P>I have created the basic flask environment</P>
      <a href="about.html">ABOUT</a>
   </body>
</head>
</html>
```

About.html

```
<HTML>
  <HEAD>
    <TITLE>ABOUT</TITLE>
  </HEAD>
<BODY>
<B>Explain what is Flask and discuss its benefits?</B><br>
<b>What is Flask?</b><br>
```

Flask is a web framework. Flask allows you to build a web application by providing tools, libraries, and technologies. This web application will be a web page, a wiki, or a big web-based calendar application or commercial website. Flask is classified into a micro-framework that means it has little to no dependencies on external libraries. There are some pros and cons. Pros mean there are little dependencies to upgrade and to watch security bugs and cons means by adding the plugin you will increase the dependencies. Flask has two dependencies, they are-

Werkzeug a WSGI utility library

Jinja2 is a template engine

Flask for Python based on Werkzeug, jinja2, and good intention.

werkzeug

Werkzeug is a utility library used for Python programming language. And also it is a toolkit of WSGI (web server gateway interface). It has licensed under the BSD License. Werkzeug can registered software for the request, response and utility function. It can also build a software framework that can support Python 2.6, 2.7, and 3.3.

Jinja2

It is a template engine for the Python programming language and is licensed under BSD license. Using the template you can build the basic layout of your page and can mention which will be changed. In this way, you can change your header of the page by upgrading it in one place only.

The benefit of the flask

There are impressive features to use the flask in your web application framework. Like-

Integrated support for unit testing

Built-in development server and fast debugger

Restful request dispatching

Unicode base

Support for cookies

Templating jinja2

WSGI 1.0 compliant

Plus flask gives you some premier control to develop your project.

HTTP request handling function

Flask has a modular design and lightweight so that it can easy to transit into web framework with some extension

You can plug your favourite ORM

Basic fundamental API is nicely shaped and coherent

Highly flexible

It is easy to deploy the flask in production

Conclusion:

Flask is the most policed and feature-rich micro framework. Flask comes with all its benefit of the fast template, strong WSGI features, and extensive documentation. Flask gives lots of good features, vast no of extension facility for a new project.

</BODY>

</HTML>

Signup.html

```
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> signup </title>
<style>
Body {
  font-family: Calibri, Helvetica, sans-serif;
  background-color: pink;
}
button {
      background-color: #4CAF50;
      width: 100%;
       color: orange;
       padding: 15px;
       margin: 10px 0px;
       border: none;
       cursor: pointer;
        }
 form {
       border: 3px solid #f1f1f1;
   }
```

```
input[type=text], input[type=email],input[type=password] {
      width: 100%;
      margin: 8px 0;
      padding: 12px 20px;
      display: inline-block;
      border: 2px solid green;
      box-sizing: border-box;
  }
button:hover {
      opacity: 0.7;
  }
 .cancelbtn {
      width: auto;
      padding: 10px 18px;
      margin: 10px 5px;
  }



.container {
      padding: 25px;
      background-color: lightblue;
  }
</style>
</head>
<body>
   <center> <h1> Signup for FLASK  </h1> </center>
   <form>
      <div class="container">
         <label>First name : </label>
         <input type="text" placeholder="Enter firstname" name="fname" required>
         <label>Last name : </label>
```

```
        <input type="text" placeholder="Enter lastname" name="lname" required>
        <label>E-mail : </label>
        <input type="email" placeholder="Enter email" name="email" required>
        <label>Password : </label>
        <input type="password" placeholder="Enter Password" name="password" required>
        <label>Confirm Password : </label>
        <input type="password" placeholder="Enter confirm Password" name="confirm password"
required>
        <button type="submit">SIGN UP</button>
        <input type="checkbox" checked="checked"> Remember me
        <button type="button" class="cancelbtn"> Cancel</button>
      </div>
   </form>
</body>
</html>
```

Signin.html

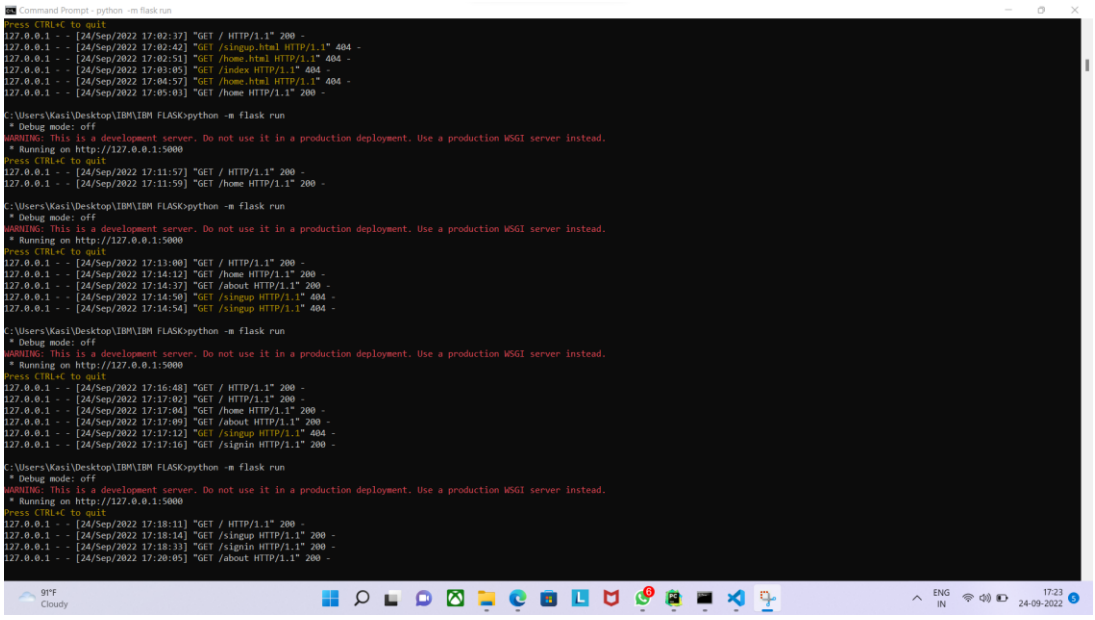
```
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> Login Page </title>
<style>
Body {
  font-family: Calibri, Helvetica, sans-serif;
  background-color: pink;
}
button {
      background-color: #4CAF50;
      width: 100%;
       color: orange;
       padding: 15px;
```

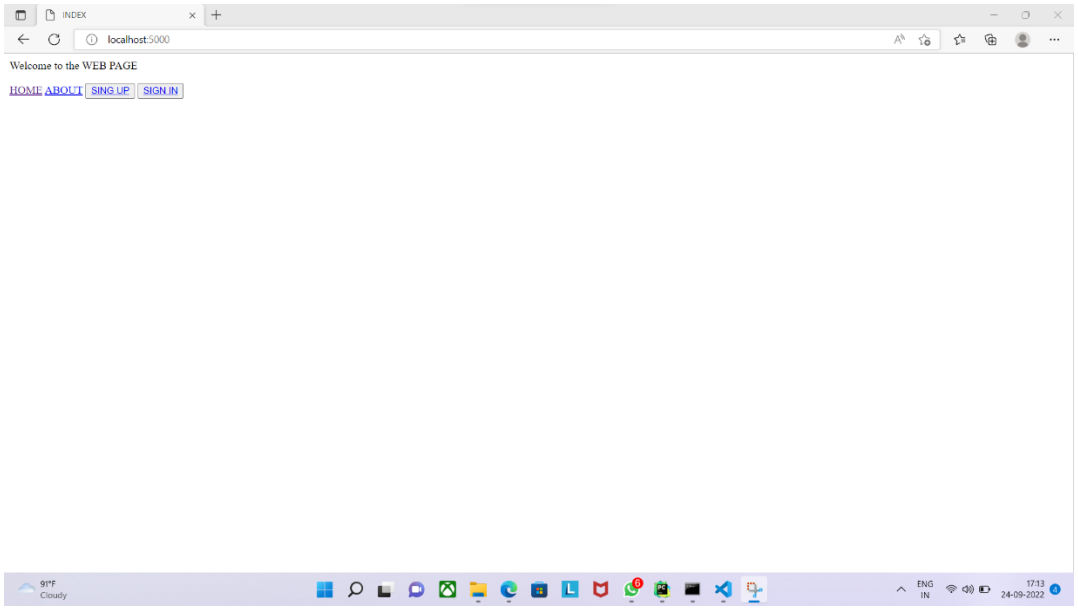
```
        margin: 10px 0px;
        border: none;
        cursor: pointer;
         }
 form {
        border: 3px solid #f1f1f1;
   }
 input[type=text], input[type=password] {
        width: 100%;
        margin: 8px 0;
        padding: 12px 20px;
        display: inline-block;
        border: 2px solid green;
        box-sizing: border-box;
   }
 button:hover {
        opacity: 0.7;
   }
  .cancelbtn {
        width: auto;
        padding: 10px 18px;
        margin: 10px 5px;
   }


 .container {
        padding: 25px;
        background-color: lightblue;
   }
</style>
</head>
```

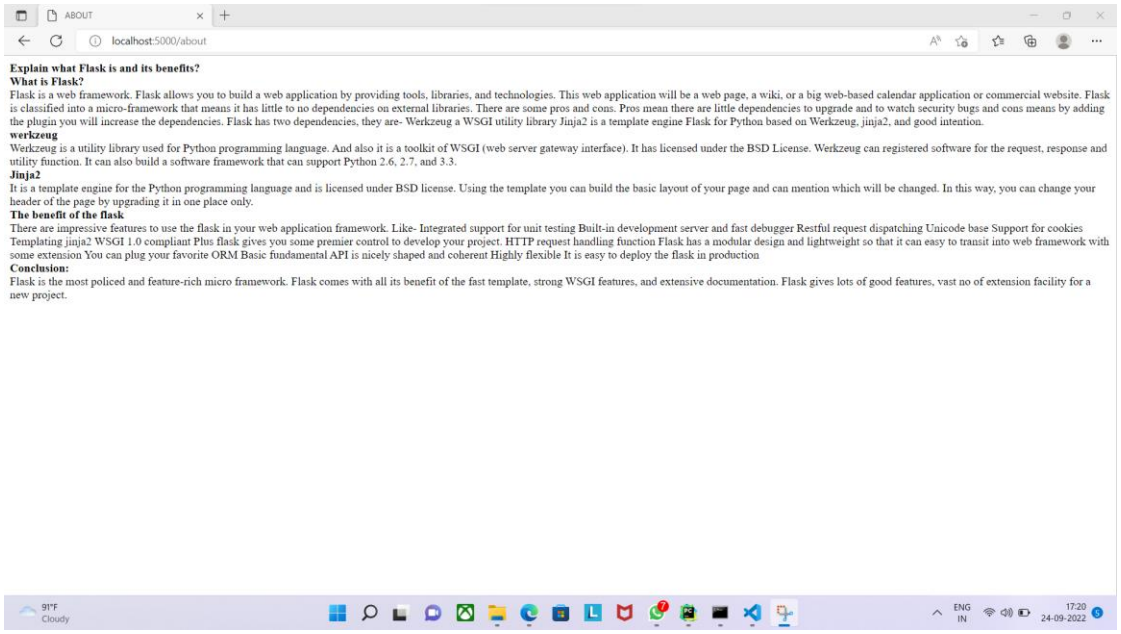
```
<body>
   <center> <h1> SIGN IN TO FLASK </h1> </center>
   <form>
      <div class="container">
         <label>Username : </label>
         <input type="text" placeholder="Enter Username" name="username" required>
         <label>Password : </label>
         <input type="password" placeholder="Enter Password" name="password" required>
         <button type="submit">Login</button>
         <input type="checkbox" checked="checked"> Remember me
         <button type="button" class="cancelbtn"> Cancel</button>
         Forgot <a href="#"> password? </a>
      </div>
   </form>
</body>
</html>
```

Output:

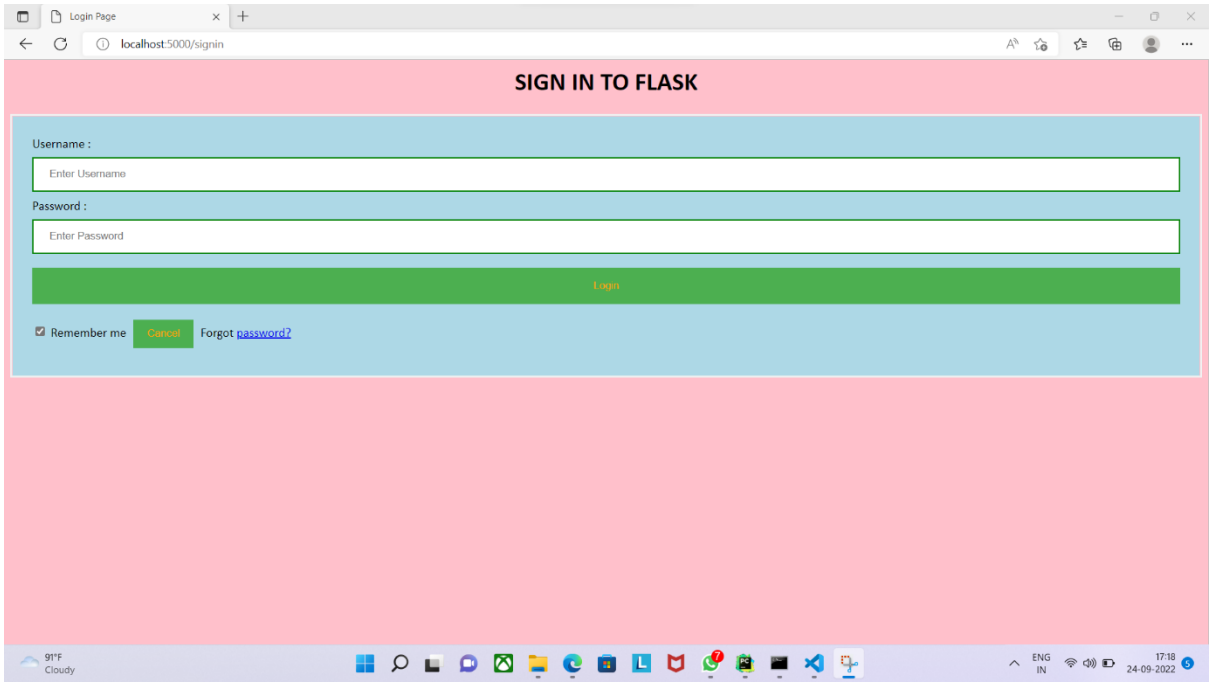



INDEX
localhost:5000
Welcome to the WEB PAGE
HOME ABOUT SING UP SIGN IN


HOME PAGE
File | file:///C:/Users/User/Desktop/Flask/Templates/home.html
HI, I AM GOKULRAJ S
I have created the basic flask environment
ABOUT


ABOUT
localhost:5000/about
Explain what Flask is and its benefits?
What is Flask?
Flask is a web framework. Flask allows you to build a web application by providing tools, libraries, and technologies. This web application will be a web page, a wiki, or a big web-based calendar application or commercial website. Flask is classified into a micro-framework that means it has little to no dependencies on external libraries. There are some pros and cons. Pros mean there are little dependencies to upgrade and to watch security bugs and cons means by adding the plugin you will increase the dependencies. Flask has two dependencies, they are- Werkzeug a WSGI utility library Jinja2 is a template engine Flask for Python based on Werkzeug, jinja2, and good intention.
werkzeug
Werkzeug is a utility library used for Python programming language. And also it is a toolkit of WSGI (web server gateway interface). It has licensed under the BSD License. Werkzeug can registered software for the request, response and utility function. It can also build a software framework that can support Python 2.6, 2.7, and 3.3.
Jinja2
It is a template engine for the Python programming language and is licensed under BSD license. Using the template you can build the basic layout of your page and can mention which will be changed. In this way, you can change your header of the page by upgrading it in one place only.
The benefit of the flask
There are impressive features to use the flask in your web application framework. Like- Integrated support for unit testing Built-in development server and fast debugger Restful request dispatching Unicode base Support for cookies Templating jinja2 WSGI 1.0 compliant Plus flask gives you some premier control to develop your project. HTTP request handling function Flask has a modular design and lightweight so that it can easy to transit into web framework with some extension You can plug your favorite ORM Basic fundamental API is nicely shaped and coherent Highly flexible It is easy to deploy the flask in production
Conclusion:
Flask is the most policed and feature-rich micro framework. Flask comes with all its benefit of the fast template, strong WSGI features, and extensive documentation. Flask gives lots of good features, vast no of extension facility for a new project.
91°F
Cloudy
ENG
IN
17:20
24-09-2022


signup
localhost:5000/singup
Signup for FLASK
First name :
Enter firstname
Last name :
Enter lastname
E-mail :
Enter email
Password :
Enter Password
Confirm Password :
Enter confirmPassword
SIGN UP
Remember me
Cancel
91°F
Cloudy
ENG
IN
17:18
24-09-2022


Login Page
localhost:5000/signin
SIGN IN TO FLASK
Username :
Enter Username
Password :
Enter Password
Login
Remember me
Cancel
Forgot password?
91°F
Cloudy
ENG
IN
17:18
24-09-2022