**120A3051**
**Shreya Idate**
**Batch: E3**

## Experiment 10

**Aim**: Design Feedback Form using Flask

**Theory:**
Flask is a web application framework written in Python. It is developed by Armin Ronacher, who leads an international group of Python enthusiasts named Pocco. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

WSGI

Web Server Gateway Interface (WSGI) has been adopted as a standard for Python web application development. WSGI is a specification for a universal interface between the web server and the web applications.

Werkzeug

It is a WSGI toolkit, which implements requests, response objects, and other utility functions. This enables building a web framework on top of it. The Flask framework uses Werkzeug as one of its bases.

Jinja2

Jinja2 is a popular templating engine for Python. A web templating system combines a template with a certain data source to render dynamic web pages.

The route() function of the Flask class is a decorator, which tells the application which URL should call the associated function.

app.route(rule, options)

The add_url_rule() function of an application object is also available to bind a URL with a function as in the above example, route() is used.

The url_for() function is very useful for dynamically building a URL for a specific function. The function accepts the name of a function as first argument, and one or more keyword arguments, each corresponding to the variable part of URL.

Jinja2 template engine, on which Flask is based. Instead of returning hardcode HTML from the function, a HTML file can be rendered by the render_template() function.

Flask will try to find the HTML file in the templates folder, in the same folder in which this script is present.

- Application folder
  - Hello.py
  - templates
    - hello.html

```python
from flask import Flask, render_template,url_for,request

app = Flask(__name__)


@app.route('/',methods = ['GET','POST'])
def index():
    if request.method == "POST":
        Firstname = str(request.form.get("fname"))
        Lastname = str(request.form.get("lname") )
        print(Firstname)
        print(Lastname)
        return Firstname+" "+Lastname+" : "+" Your feedback is submitted"
    return render_template('feedback.html')
```

Add feedbackform.html

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

<head>
    <meta charset="UTF-8">
    <title>Feedback form</title>
</head>
```

```html
<body>

    <div class="wrapper">
        <h2>Feedback Form</h2>
        <div id="error_message"></div>
        <form method="post" id="myform" onsubmit="return validate();">
            <div class="input_field">
                <input type="text" placeholder="First Name" id="fname" name="fname">
            </div>
            <div class="input_field">
                <input type="text" placeholder="Last Name" id="lname" name="lname">
            </div>
            <div class="input_field">
                <input type="text" placeholder="Phone" id="phone">
            </div>
            <div class="input_field">
                <input type="text" placeholder="Email" id="email">
            </div>
            <div class="input_field">
                <textarea placeholder="Your Feedback" id="yourfeedback"></textarea>
            </div>
            <div class="btn">
                <input type="submit">
            </div>
        </form>
    </div>

</body>
</html>
```
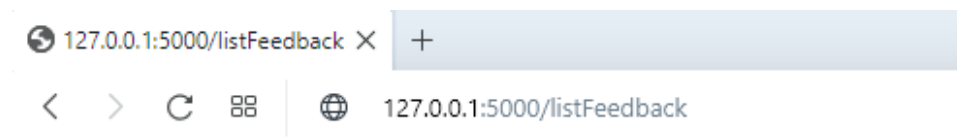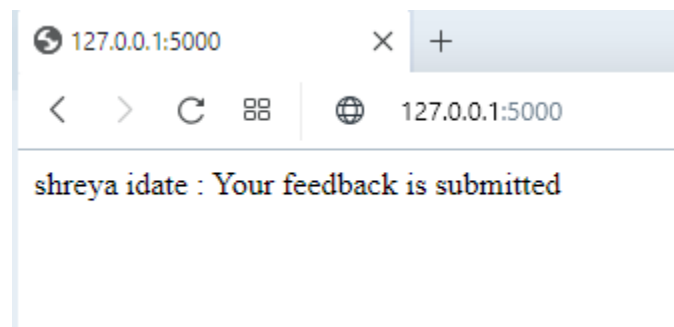
54

**Output:**

### Feedback Form

| | |
|---|---|
| shreya | |
| idate | |
| 123456789 | |
| shreya@gmail.com | |
| good | |

Submit

shreya idate : Your feedback is submitted

| FirstName | LastName | Phone | Email | Feedback |
|-----------|----------|-----------|------------------|-----------|
| Shreya | Idate | 8779626135 | shreya@gmail.com | Excellent |
| Jane | Doe | 123456789 | jane@gmail.com | good |

Go back to home page

**Conclusion:** Feedback Form is designed using Flask and run successfully.