

## Experiment 5

DATE:

Aim - TO Study Flask

Problem Statement -  
Design using flask

a. A contact us form

b. A portfolio website

Demonstrate the usage of Routing, Request object and Templating.

Theory -

1. List some of the core features of flask?

- a. Lightweight and easy to use.
- b. Jinja templating
- c. Unit testing
- d. Support for secure cookies.

2. Why do we use Flask(-name-) in flask?

- a. The ~~name~~ `--name--` argument is passed to the flask class to create its instance, which is then used to run the application.
- b. Flask uses it to know from where to get resources, templates, static files, etc. required to run the application.
- c. It is just a convenient way to get the import name of the place where the app is defined.



3. What is Template (Template Inheritance) in flask?

→ a. Template Inheritance is a mechanism that allows you to create a base 'skeleton' template that contains all common elements of your site and define blocks that child templates can override.

b. Example

```
# base.html
```

```
<!doctype html>
```

```
<html>
```

```
  <head>
```

```
    {% block title %} {% endblock %}
```

```
  </title>
```

```
</head>
```

```
<body>
```

```
  <header>
```

```
    {% block header %}
```

```
      <h1> Welcome! </h1>
```

```
    {% endblock %}
```

```
  </header>
```

```
# child.html
```

```
{% extends "base.html" %}
```

```
{% block title %} My page {% endblock %}
```



Q. What methods of HTTP are implemented in flask?

→ The HTTP methods used are:-

- a. GET - Used to retrieve data from the server
- b. POST - Enables users to send data to server.
- c. PUT - Used to update an existing resource on the server.
- d. DELETE - Used to delete a resource on the server.
- e. HEAD - Retrieves the resource's headers without the resource itself.

Q. What is difference between flask and Django framework?

	Flask	Django
→		
a.	It is built for rapid development	It is built for easy and simple projects.
b.	It is WSGI framework	It is a full stack framework
c.	Provides support for API	Does not support API.
d.	Support visual debug	No support for visual debug
e.	Does not offer dynamic HTML pages	Offers dynamic HTML pages.
f.	Suitable for single page application	We can divide a project into multiple pages.

**Q1.**

**Code:**

**app.py**

```
from flask import Flask, render_template, request

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('contact.html')

@app.route('/submit', methods=['POST'])
def submit():
    name = request.form.get('name')
    email = request.form.get('email')
    message = request.form.get('message')
    # do something with the form data, e.g. send an email
    return 'Thanks for your message, ' + name + '!'

if __name__ == '__main__':
    app.run(debug=True)
```

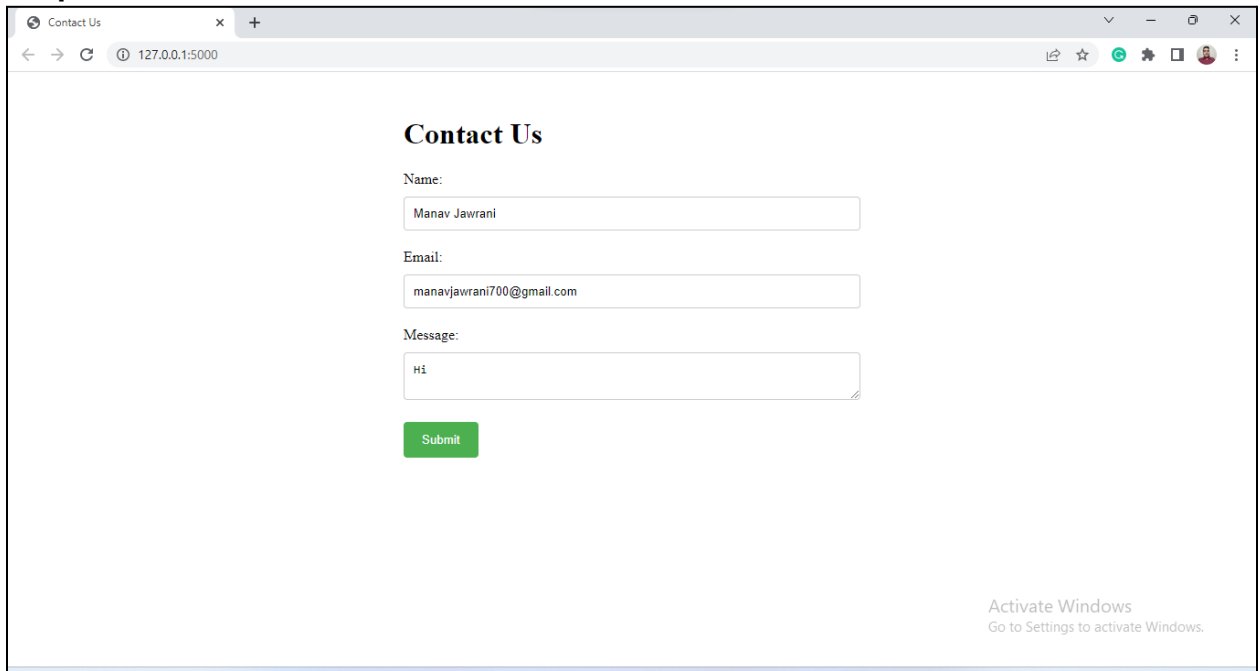
**contact.html**

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

<title>Contact Us</title>
<style>
    label {
        display: block;
        margin-bottom: 10px;
    }
    input[type="text"], input[type="email"], textarea {
        width: 100%;
        padding: 10px;
        margin-bottom: 20px;
        border: 1px solid #ccc;
        border-radius: 4px;
        box-sizing: border-box;
    }
    input[type="submit"] {
        background-color: #4CAF50;
        color: white;
```

```
padding: 12px 20px;
border: none;
border-radius: 4px;
cursor: pointer;
}
input[type="submit"]:hover {
  background-color: #45a049;
}
.container {
  max-width: 500px;
  margin: auto;
  padding: 20px;
}
</style>
</head>
<body>
<div class="container">
  <h1>Contact Us</h1>
  <form action="/submit" method="post">
    <label for="name">Name:</label>
    <input type="text" id="name" name="name">
    <label for="email">Email:</label>
    <input type="email" id="email" name="email">
    <label for="message">Message:</label>
    <textarea id="message" name="message"></textarea>
    <input type="submit" value="Submit">
  </form>
</div>
</body>
</html>
```

## Output:



A screenshot of a web browser window titled 'Contact Us'. The address bar shows '127.0.0.1:5000'. The page content includes a heading 'Contact Us', followed by three input fields labeled 'Name:', 'Email:', and 'Message:'. The 'Name' field contains 'Manav Jawrani', the 'Email' field contains 'manavjawrani700@gmail.com', and the 'Message' field contains 'Hi'. Below these fields is a green 'Submit' button. In the bottom right corner, there is a watermark that says 'Activate Windows Go to Settings to activate Windows.'

Contact Us

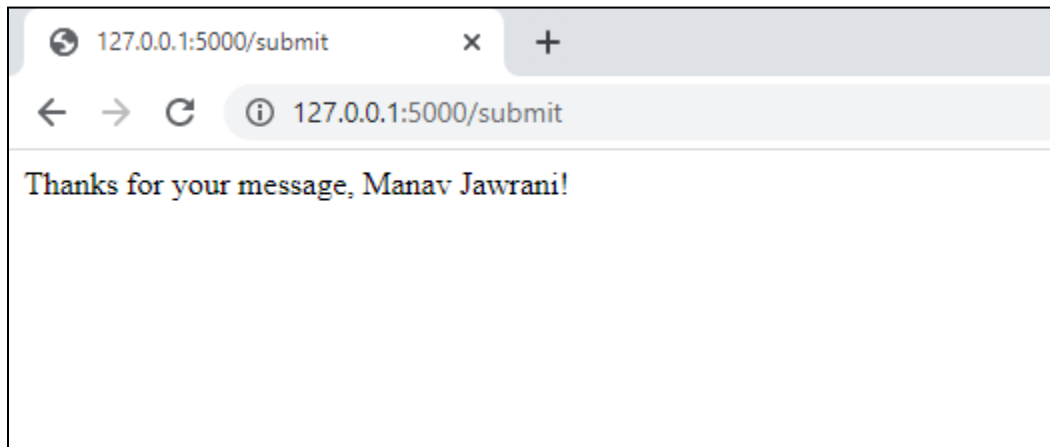
Name:  
Manav Jawrani

Email:  
manavjawrani700@gmail.com

Message:  
Hi

Submit

Activate Windows  
Go to Settings to activate Windows.



A screenshot of a web browser window showing the result of a submission. The address bar shows '127.0.0.1:5000/submit'. The page content displays the message 'Thanks for your message, Manav Jawrani!' in a serif font.

127.0.0.1:5000/submit

Thanks for your message, Manav Jawrani!

## Q2.

### Code

#### app.py

```
from flask import Flask, render_template, request

app = Flask(__name__)

# Home page
@app.route('/')
def home():
    return render_template('index.html')

# About page
@app.route('/about')
def about():
    return render_template('about.html')

# Contact page
@app.route('/contact', methods=['GET', 'POST'])
def contact():
    if request.method == 'POST':
        # process the form data
        name = request.form['name']
        email = request.form['email']
        message = request.form['message']
        # render the contact template with success message
        return render_template('contact.html', success=True)
    else:
        # render the contact template with failure message
        return render_template('contact.html', success=False)

if __name__ == '__main__':
    app.run(debug=True)
```

#### index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Portfolio | Home</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
</head>
<body>
    <header>
        <nav>
```

```
<ul>
  <li><a href="/">Home</a></li>
  <li><a href="/about">About</a></li>
  <li><a href="/contact">Contact</a></li>
</ul>
</nav>
</header>
<main>
  <section class="hero">
    <h1>Welcome to my portfolio website</h1>
    <p>I am a software engineer with experience in web development, machine
learning, and data analysis.</p>
  </section>
  <section class="about">
    <h2>About Me</h2>
    <div class="about-content">
      
      <p>I am a software engineer with experience in web development, machine
learning, and data analysis. I am pursuing a bachelor's degree in information technology
and I am passionate about solving challenging problems using technology.</p>
    </div>
  </section>
</main>
<footer>
  <p>&copy; 2023 - Portfolio Website</p>
</footer>
</body>
</html>
```



## Output

