

# ASSIGNMENT10

AIM: Implement a feedback form in flask with validations.

Theory:

Flask is a light weight and flexible web framework for Python, designed to make it easy to build web applications quickly and with minimal boilerplate code. Here's an overview of Flask:

## 1. Minimalist Framework:

- Flask is often described as a "micro" web framework because it aims to keep the core simple and extensible. It provides just what you need to get started without imposing any particular way of doing things.

## 2. Easy to Get Started:

- Flask is beginner-friendly and has a gentle learning curve. You can create a simple web application with just a few lines of code.

## 3. Routing:

- Flask uses routes to map URLs to Python functions called view functions. You can define routes using decorators like **@app.route('/')**.

## 4. Templates:

- Flask integrates with Jinja2, a powerful and feature-rich template engine for Python. Templates allow you to separate your application's logic from its presentation.

## 5. HTTP Request Handling:

- Flask provides convenient decorators for handling different types of HTTP requests, such as **@app.route('/post', methods=['GET', 'POST'])** for handling both GET and POST requests to the '/post' URL.

## 6. Request and Response Objects:

- Flask provides request and response objects to handle incoming requests and generate responses. These objects give you access to request data (e.g., form data, query parameters) and allow you to customize the response (e.g., setting headers, cookies).

## 7. Extensions:

- Flask has a rich ecosystem of extensions that add extra functionality to your applications, such as Flask-WTF for form handling, Flask-SQLAlchemy for database integration, Flask-RESTful for building RESTful APIs, and many more.

## 8. Development Server:

- Flask comes with a built-in development server, making it easy to test your applications locally during development. You can start the development server by running **flask run** from the command line.

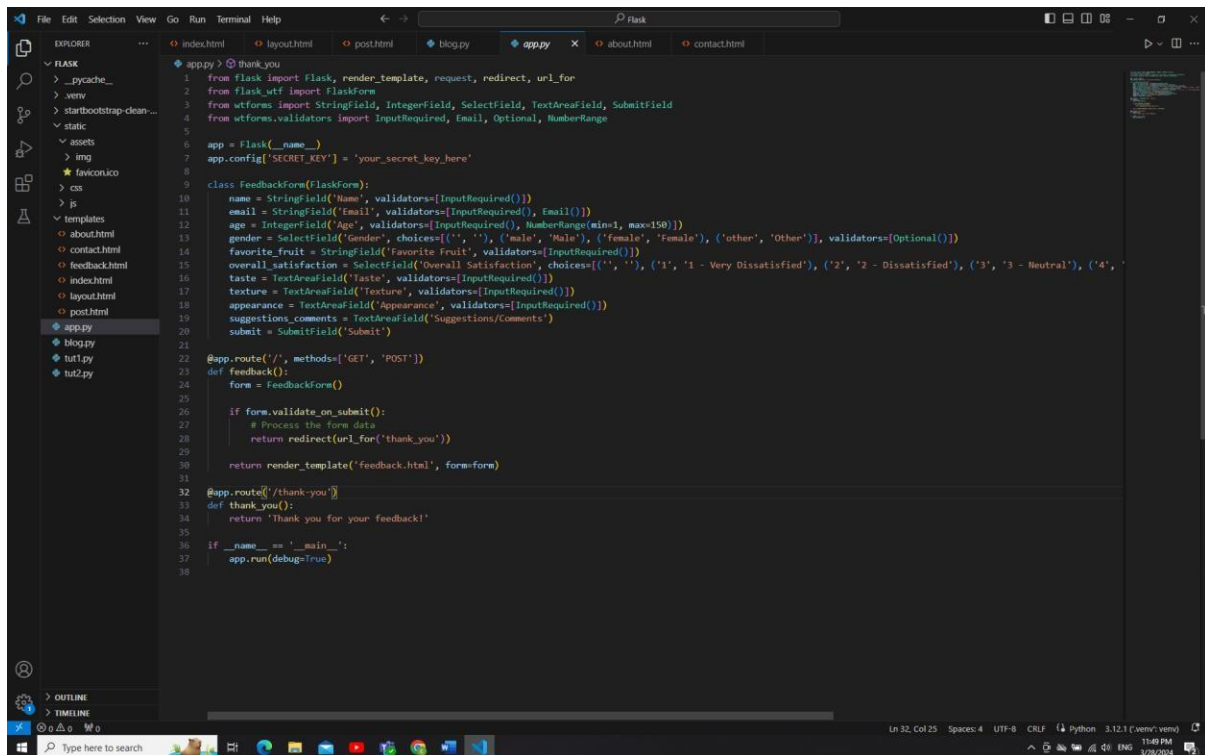
## 9. Deployment Options:

- Flask applications can be deployed in various ways, including traditional web servers like Apache or Nginx, cloud platforms like Heroku or AWS, or using containers with Docker.

## 10. Community and Documentation:

- Flask has a vibrant community of developers and enthusiasts who contribute to its Ecosystem by creating extensions, tutorials, and resources. The Flask documentation is comprehensive and well-maintained, making it easy to find answers to your questions.

## Implementation:



```

1 from flask import Flask, render_template, request, redirect, url_for
2 from flask_wtf import FlaskForm
3 from wtforms import StringField, IntegerField, SelectField, TextAreaField, SubmitField
4 from wtforms.validators import InputRequired, Email, Optional, NumberRange
5
6 app = Flask(__name__)
7 app.config['SECRET_KEY'] = 'your_secret_key_here'
8
9 class FeedbackForm(FlaskForm):
10     name = StringField('Name', validators=[InputRequired()])
11     email = StringField('Email', validators=[InputRequired(), Email()])
12     age = IntegerField('Age', validators=[InputRequired(), NumberRange(min=1, max=150)])
13     gender = SelectField('Gender', choices=[('M', 'Male'), ('F', 'Female'), ('O', 'Other')], validators=[Optional()])
14     favorite_fruit = StringField('Favorite Fruit', validators=[InputRequired()])
15     overall_satisfaction = SelectField('Overall Satisfaction', choices=[('1', '1 - Very Dissatisfied'), ('2', '2 - Dissatisfied'), ('3', '3 - Neutral'), ('4', '4 - Satisfied')], validators=[Optional()])
16     taste = TextAreaField('Taste', validators=[InputRequired()])
17     texture = TextAreaField('Texture', validators=[InputRequired()])
18     appearance = TextAreaField('Appearance', validators=[InputRequired()])
19     suggestions_comments = TextAreaField('Suggestions/Comments')
20     submit = SubmitField('Submit')
21
22 @app.route('/', methods=['GET', 'POST'])
23 def feedback():
24     form = FeedbackForm()
25
26     if form.validate_on_submit():
27         # Process the form data
28         return redirect(url_for('thank_you'))
29
30     return render_template('feedback.html', form=form)
31
32 @app.route('/thank-you')
33 def thank_you():
34     return 'Thank you for your feedback!'
35
36 if __name__ == '__main__':
37     app.run(debug=True)
38

```

```
File Edit Selection View Go Run Terminal Help
index.html layout.html post.html blog.py feedback.html about.html contact.html

templates > feedback.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Fruit Feedback Form</title>
7   <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
8   <style>
9     .container {
10       max-width: 600px;
11       margin: auto;
12       margin-top: 50px;
13       background-color: #f8f9fa;
14       padding: 30px;
15       border-radius: 10px;
16       box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
17     }
18     .form-group {
19       margin-bottom: 20px;
20     }
21     .error {
22       color: red;
23     }
24     .btn-primary {
25       background-color: #007bff;
26       border-color: #007bff;
27     }
28     .btn-primary:hover {
29       background-color: #0056b3;
30       border-color: #0056b3;
31     }
32   </style>
33 </head>
34 <body>
35   <div class="container">
36     <h2 class="text-center mb-4">Fruit Feedback Form</h2>
37     <form method="POST">
38       {{ form.hidden_tag() }}
39       <div class="form-group">
40         <label for="name">Name:</label>
41         {{ form.name(size=30, class="form-control") }}
42         {% for error in form.name.errors %}
43           <span class="error">{{ error }}</span>
44         {% endfor %}
45       </div>
46       <div class="form-group">
47         <label for="email">Email:</label>
48         {{ form.email(size=30, class="form-control") }}
49         {% for error in form.email.errors %}
50           <span class="error">{{ error }}</span>
51         {% endfor %}
52       </div>
53       <div class="form-group">
54         <label for="age">Age:</label>
55         {{ form.age(size=30, class="form-control") }}
56         {% for error in form.age.errors %}
57           <span class="error">{{ error }}</span>
58         {% endfor %}
59       </div>
60       <div class="form-group">
61         <label for="gender">Gender:</label>
62         {{ form.gender(size=30, class="form-control") }}
63         {% for error in form.gender.errors %}
64           <span class="error">{{ error }}</span>
65         {% endfor %}
66       </div>
67       <div class="form-group">
68         <label for="fruit">Favorite Fruit:</label>
69         {{ form.fruit(size=30, class="form-control") }}
70         {% for error in form.fruit.errors %}
71           <span class="error">{{ error }}</span>
72         {% endfor %}
73       </div>
74       <div class="form-group">
75         <label for="satisfaction">Overall Satisfaction:</label>
76         {{ form.satisfaction(size=30, class="form-control") }}
77         {% for error in form.satisfaction.errors %}
78           <span class="error">{{ error }}</span>
79         {% endfor %}
80       </div>
81       <div class="form-group">
82         <label for="taste">Taste:</label>
83         {{ form.taste(size=30, class="form-control") }}
84         {% for error in form.taste.errors %}
85           <span class="error">{{ error }}</span>
86         {% endfor %}
87       </div>
88       <div class="form-group">
89         {{ form.submit(class="btn btn-primary") }}
90       </div>
91     </form>
92   </div>
93 </body>
94 </html>
```

ChatGPT WhatsApp Interface DHT11 (Humidity Sensor) angular compiler online - Google Fruit Feedback Form

127.0.0.1:5000

Gmail YouTube Maps Practice | GeeksforGeeks WhatsApp Shiver's SDE Sheet - T...

## Fruit Feedback Form

Name:

Email:

Age:

Gender:

Favorite Fruit:

Overall Satisfaction:

Taste:

11:40 PM 3/28/2024

The screenshot displays a web browser window with the address bar showing '127.0.0.1:5000'. The browser's tab bar includes 'ChatGPT', 'WhatsApp', 'Interface DHT11 (Humidity Sensor)', 'angular compiler online - Google', and 'Fruit Feedback Form'. The browser's address bar also shows a star icon, a lock icon, and a download icon. The browser's search bar contains '127.0.0.1:5000'. The browser's sidebar shows 'Gmail', 'YouTube', 'Maps', 'Practice | GeeksforGe...', 'WhatsApp', and 'Driver's SDE Sheet - T...'. The browser's main content area shows a form with the following fields:

- A dropdown menu.
- A text input field labeled 'Favorite Fruit:'.
- A text input field labeled 'Overall Satisfaction:'.
- A text input field labeled 'Taste:'.
- A text input field labeled 'Texture:'.
- A text input field labeled 'Appearance:'.
- A text area labeled 'Suggestions/Comments:'.
- A blue 'Submit' button.