Flask Web Development with Python - Full Guide

Pre-requisites to Learn Flask

Before starting with Flask, ensure you're comfortable with the following:

1. Python Programming:
   - Variables, data types, loops, functions, exception handling, file I/O

2. HTML, CSS, and JavaScript:
   - Basic structure of HTML documents, styling with CSS, and interactivity using JS

3. HTTP Protocol:
   - Understanding request methods like GET, POST, URL routing, status codes

4. Python Package Management:
   - Using pip to install libraries

5. (Optional) Jinja2 Template Engine:
   - Used by Flask to render dynamic web pages

Introduction to Flask

Flask is a lightweight and micro web framework for Python. It's widely used for building web applications, RESTful APIs, and dashboards.

Why Flask?
- Simple to use, beginner-friendly
- Flexible and minimalistic
- Extensible with plugins

Installation:
Use the following command to install Flask:

Flask Web Development with Python - Full Guide

pip install flask

Example 1: Hello World App

Code:

```
from flask import Flask


app = Flask(__name__)


@app.route('/')
def home():
    return "Hello, Flask World!"


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

Explanation:

- Creates a basic web server running on localhost:5000

- "/" is the home route

Example 2: Dynamic Routing

Code:

```
@app.route('/user/<username>')
def show_user(username):
    return f"Hello, {username}!"
```

Explanation:

- Dynamic segments in the URL handled using angle brackets < >

Example 3: Handling Forms

HTML:

```html
<form action="/login" method="post">
  <input type="text" name="username">
  <input type="submit">
</form>
```

Python:

```python
from flask import request


@app.route('/login', methods=['POST'])
def login():
    username = request.form['username']
    return f"Welcome, {username}"
```

Example 4: Jinja2 Templates

HTML Template (welcome.html):

```html
<h1>Welcome, {{ name }}</h1>
```

Python Code:

```python
from flask import render_template


@app.route('/welcome/<name>')
def welcome(name):
    return render_template('welcome.html', name=name)
```

Example 5: Creating a REST API

Code:

```python
from flask import jsonify
```

```
@app.route('/api/data')

def data():

    return jsonify({'name': 'Abhraneel', 'language': 'Python'})
```

Example 6: File Uploading

HTML:

```
<form action="/upload" method="post" enctype="multipart/form-data">

  <input type="file" name="file">

  <input type="submit">

</form>
```

Python:

```
@app.route('/upload', methods=['POST'])

def upload():

    f = request.files['file']

    f.save(f'uploads/{f.filename}')

    return "File uploaded"
```