

ONE PAGE SUMMARY

IT & Software > Backend Development > Flask

Introduction to Flask

Flask: Your Gateway to Simple Web Development!

DEFINITION

Flask is a lightweight web framework for Python that helps you build web applications quickly and easily. It provides tools and libraries to create web servers and manage web requests.

KEY CONCEPTS

Routing

Routing in Flask determines how URLs are connected to functions in your application.

Templates

Templates allow you to separate HTML from Python code, making it easier to manage your web pages.

Request and Response

Flask handles incoming requests and sends back responses, enabling interaction between the server and client.

Flask Extensions

Extensions add functionality to Flask, such as database integration or authentication, enhancing your application.

Development Server

Flask comes with a built-in development server for testing your application locally before deployment.

EXAMPLES

- Creating a simple web page with Flask's routing.
- Using Jinja2 templates to render dynamic HTML content.
- Handling form submissions with request data in Flask.

MEMORY TIPS

- ★ Remember 'Flask' as a 'flask of tools' for building web apps.
- ★ Think of routing as 'mapping' URLs to functions.
- ★ Associate templates with 'blueprints' for your web pages.

COMMON MISTAKES

- ✗ Forgetting to import necessary Flask modules.
- ✗ Not defining routes correctly, leading to 404 errors.
- ✗ Neglecting to use the correct HTTP methods (GET, POST).

QUICK RECAP

Flask is a Python framework for creating web applications with ease. Key features include routing, templates, and request handling. Remember to utilize Flask extensions for added functionality and test your app with the built-in server.