

Programação Orientada a Objetos (POO) e Flask: Uma Introdução

Apresentamos uma visão geral da POO e uso do Flask.

Aprenda conceitos essenciais e como aplicá-los em frameworks.



Os Pilares da Programação Orientada a Objetos

Abstração

Simplificando a complexidade com modelos.

Encapsulamento

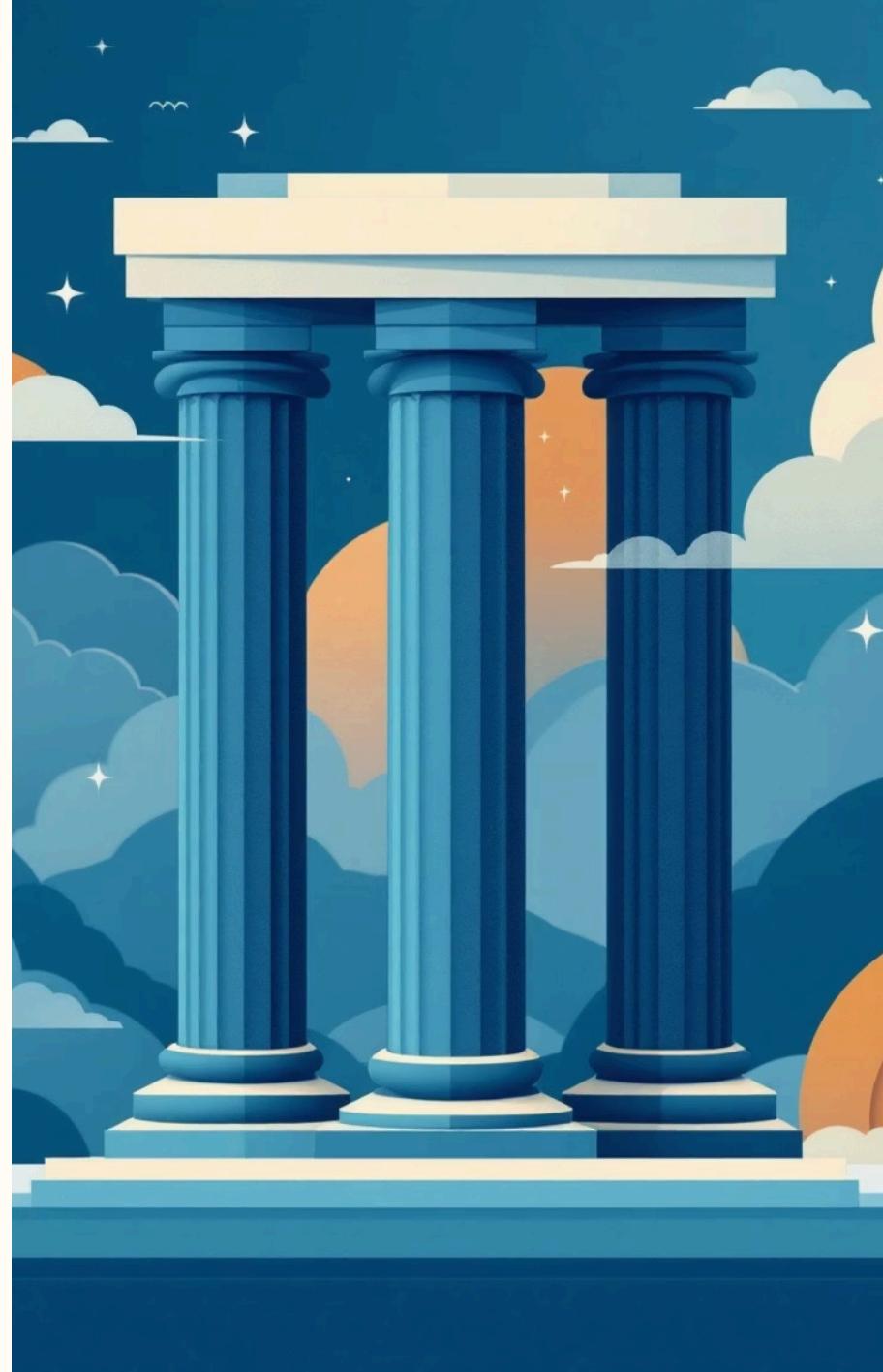
Protegendo dados com controle de acesso.

Herança

Reutilizando código com hierarquias de classes.

Polimorfismo

Adaptando o comportamento de objetos.



Abstração e Encapsulamento em Detalhe

Abstração

Classe Animal como modelo para Cachorro e Gato.

Simplifica características comuns e oculta detalhes.

Encapsulamento

Atributos privados protegidos por getters e setters públicos.

Garante modularidade e segurança do código.

Herança e Polimorfismo em Ação

1

Herança

Classes base e subclasses compartilham propriedades.

Exemplo: Veículo -> Carro, Moto.

2

Polimorfismo

Métodos iguais com comportamentos diferentes.

Exemplo: método emitir_som() em diferentes animais.



Introdução ao Flask: Um Microframework Python

O que é Flask?

Microframework leve para desenvolvimento web.

Comparação

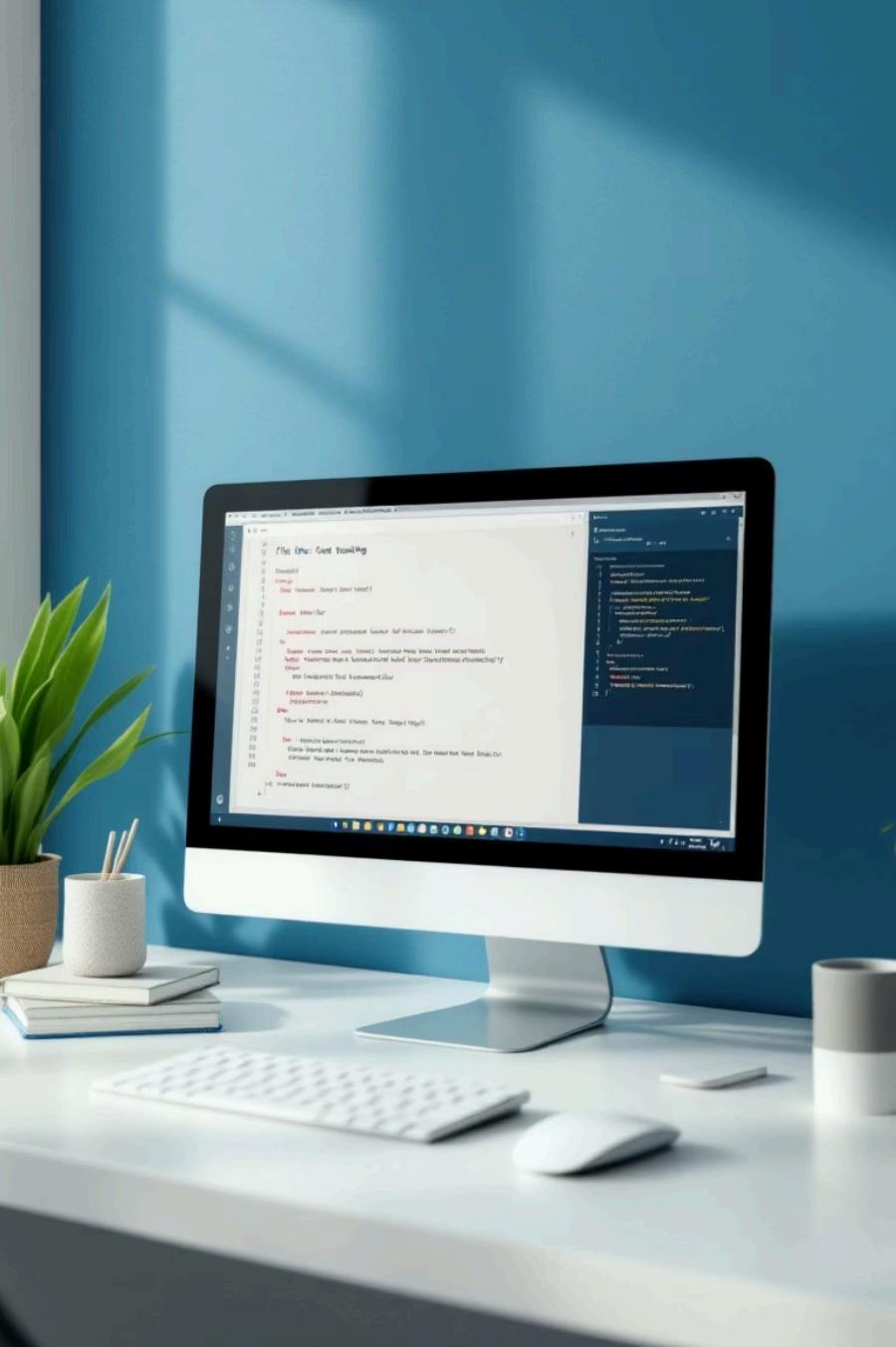
Mais simples e flexível que Django.

Instalação

Use pip install Flask para começar.

Vantagens

Flexibilidade e alta extensibilidade.



Roteamento e Views no Flask

Rotas

Definidas com decorador
`@app.route()`.

Views

Funções que retornam HTML, JSON
e outros formatos.

Exemplo

`return render_template('index.html')` para exibir páginas.

Flask e POO: Uma Combinação Poderosa

Organização

Classes e objetos estruturam o código.

MVC

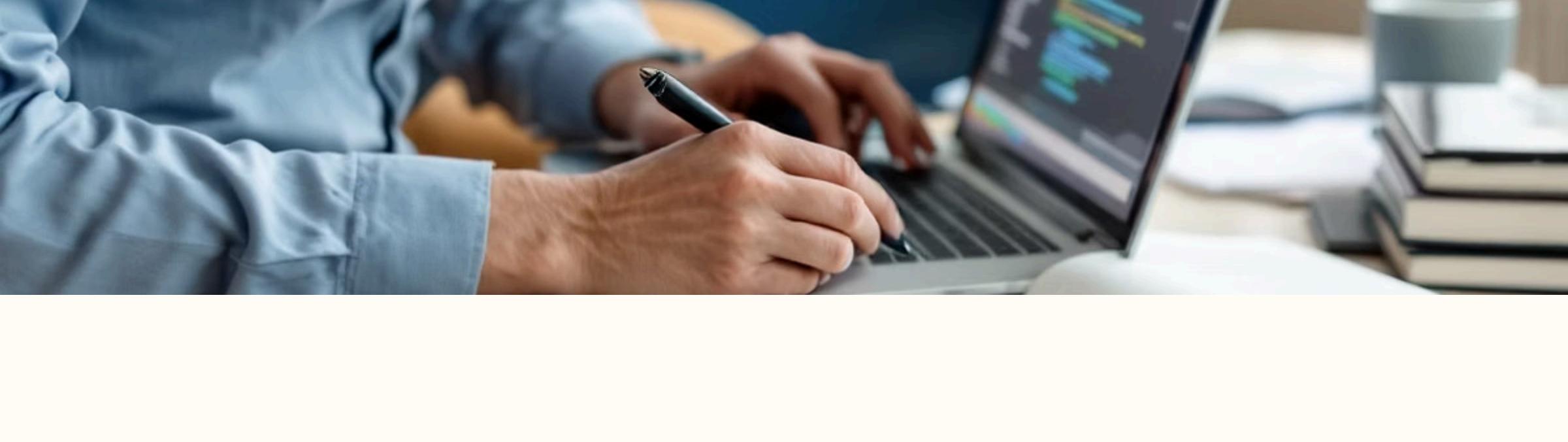
Separação clara entre lógica, dados e interface.

Escalabilidade

Criação de apps manutáveis e robustos.

Exemplo prático

Integração de POO em projetos Flask.



Conclusão e Próximos Passos

1 Recapitação

POO e Flask formam base para desenvolvimento eficiente.

2 Recursos

Explore tutoriais, documentação e cursos online.

3 Prática

Desenvolva projetos para fixar os conceitos.