

✓ FLASK

O que é:

Flask é um microframework para criar aplicações web em Python.

Pra que serve:

Permite criar rotas, páginas, lidar com formulários, sessões, etc., de forma simples e leve.

✓ FLASK-LOGIN

O que é:

Uma extensão do Flask que adiciona suporte a login e controle de usuários.

Pra que serve:

Facilita o gerenciamento de login, logout e verificação se o usuário está autenticado ou não.

✓ ROTAS

O que é:

As rotas são as URLs que definem o caminho para acessar uma página ou função no servidor.

Pra que serve:

Ex: `@app.route('/login')` indica que, ao acessar `/login`, uma função será chamada.

✓ MÉTODOS POST e GET

- **GET**: Envia os dados pela URL (ex: quando acessamos uma página).
- **POST**: Envia os dados de forma escondida (ex: ao enviar um formulário de login).

Pra que serve:

Permitem o envio e recebimento de dados entre o navegador e o servidor Flask.

✓ REQUEST

O que é:

Um objeto do Flask que pega os dados enviados pelo usuário.

Pra que serve:

Lê os campos de formulários com `request.form`, checa o método (`POST` ou `GET`), etc.

✓ MAKE_RESPONSE

O que é:

Função do Flask que cria uma resposta personalizada para o navegador.

Pra que serve:

Usado para definir **cookies**, cabeçalhos ou alterar a resposta padrão.

Exemplo:

python

CopiarEditar

```
resp = make_response(render_template('pagina.html'))
resp.set_cookie('nome', 'valor')
return resp
```

✓ COOKIES

O que é:

Pequenos dados salvos no navegador do usuário.

Pra que serve:

Armazenam informações simples, como nome do usuário ou preferências.

✓ SESSION (com SECRET_KEY)

O que é:

Um dicionário que guarda dados do usuário durante a navegação.

Pra que serve:

Armazena variáveis como `"usuario_logado"` para manter o estado da sessão.

SECRET_KEY: É obrigatória para proteger os dados da sessão com criptografia.

✓ URL_FOR

O que é:

Função do Flask para gerar URLs com base no nome da função da rota.

Pra que serve:

Evita erros de digitação e permite gerar links dinâmicos.

Exemplo:

python

CopiarEditar

```
url_for('login') # Gera "/login"
```

✓ TEMPLATES

O que é:

Arquivos HTML usados para mostrar conteúdo na tela.

Pra que serve:

Permite separar o visual do sistema (HTML) da lógica Python.

✓ EXTENDS

O que é:

Parte do sistema de herança de templates do Jinja2 (usado pelo Flask).

Pra que serve:

Permite que outras páginas **herdem** a base (`base.html`) e evitem repetir HTML.

Exemplo:

html

CopiarEditar

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

✓ INCLUDE

O que é:

Inclui outro pedaço de template dentro de um template.

Pra que serve:

Útil para menus, rodapés, cabeçalhos.

Exemplo:

html

CopiarEditar

```
{% include "menu.html" %}
```

FLASH

O que é:

Função do Flask para enviar **mensagens temporárias** para o usuário.

Pra que serve:

Mostrar avisos como "Login inválido" ou "Cadastro realizado com sucesso".

Exemplo:

python

CopiarEditar

```
flash('Login realizado com sucesso!')
```

LOGINMANAGER

O que é:

Objeto da extensão Flask-Login que gerencia a autenticação.

Pra que serve:

Configura o login, define a rota de login, e carrega usuários pela sessão.

LOGIN_REQUIRED

O que é:

Decorador do Flask-Login.

Pra que serve:

Protege rotas: só permite acesso **se o usuário estiver logado**.

Exemplo:

python

CopiarEditar

```
@app.route('/dashboard')
```

```
@login_required
def dashboard():
    ...
```

✓ LOGIN_USER

O que é:

Função que faz o login do usuário no sistema.

Pra que serve:

Após verificar a senha, você chama `login_user(usuario)` para autenticar.

✓ LOGOUT_USER

O que é:

Função do Flask-Login.

Pra que serve:

Desloga o usuário e limpa a sessão.

✓ @LOGIN_MANAGER.USER_LOADER

O que é:

Decorador que define como o sistema deve **buscar o usuário** pelo ID salvo na sessão.

Pra que serve:

Obrigatório para o Flask-Login funcionar.

Exemplo:

python

CopiarEditar

```
@login_manager.user_loader
def load_user(user_id):
    return usuarios.get(user_id)
```