

# Lista 5

---

Gerado com IA

## Exercício: Criando um Sistema Completo de Sessões e Cookies com Formulário em Flask

**Objetivo:** Aprender a criar um formulário que armazena informações do usuário usando sessões e cookies, e a processar dados através de requisições **GET** e **POST**.

### Passo 1: Preparando o Ambiente

#### 1. Instalação do Flask (se necessário)

- Caso o ambiente virtual ainda não esteja configurado, crie e ative-o:

```
python -m venv venv
```

- Ative o ambiente virtual:

- No Windows:

```
venv\Scripts\activate
```

- No Mac/Linux:

```
source venv/bin/activate
```

- Instale o Flask:

```
pip install Flask
```

- 2. **Objetivo deste passo:** Preparar o ambiente de desenvolvimento para o Flask.

### Passo 2: Configurando o Arquivo de Aplicação para Sessões e Cookies

- 1. No diretório do projeto, crie o arquivo **app.py** com o seguinte conteúdo:

```
from flask import Flask, request, render_template, session, redirect, url_for, make_response

app = Flask(__name__)
app.secret_key = 'chave-secreta' # Necessário para usar sessões
```

2. **Objetivo deste passo:** Configurar a aplicação para suportar sessões definindo uma `secret_key`.

### Passo 3: Criando o Template com Formulário

1. Crie uma pasta chamada `templates` no mesmo diretório do arquivo `app.py`.
2. Dentro da pasta `templates`, crie um arquivo chamado `usuario.html` com o seguinte conteúdo:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
    <meta charset="UTF-8">
    <title>Formulário de Usuário</title>
</head>
<body>
    <h1>Informações do Usuário</h1>
    <form action="{{ url_for('salvar_usuario') }}" method="post">
        <label for="nome">Nome:</label>
        <input type="text" id="nome" name="nome"><br><br>
        <label for="idade">Idade:</label>
        <input type="number" id="idade" name="idade"><br><br>
        <button type="submit">Enviar</button>
    </form>
    {% if nome and idade %}
        <p>Bem-vindo, {{ nome }}! Idade: {{ idade }}</p>
    {% endif %}
    <a href="{{ url_for('sair') }}">Sair</a>
</body>
</html>
```

3. **Objetivo deste passo:** Criar um formulário HTML para o usuário inserir nome e idade, que serão armazenados em sessões e cookies.

### Passo 4: Definindo Rotas para Sessões e Cookies

1. No arquivo `app.py`, adicione as rotas para salvar e exibir informações do usuário, e para limpar sessões e cookies:

```
@app.route('/usuario', methods=['GET', 'POST'])
def salvar_usuario():
    if request.method == 'POST':
        nome = request.form['nome']
        idade = request.form['idade']
        # Armazenando dados na sessão
        session['nome'] = nome
        session['idade'] = idade
        # Armazenando dados em um cookie com validade de um dia
        resposta = make_response(redirect(url_for('salvar_usuario')))
        resposta.set_cookie('nome', nome, max_age=60*60*24)
```

```
        resposta.set_cookie('idade', idade, max_age=60*60*24)
    return resposta

# Recuperando dados da sessão ou cookies
nome = session.get('nome') or request.cookies.get('nome')
idade = session.get('idade') or request.cookies.get('idade')
return render_template('usuario.html', nome=nome, idade=idade)

@app.route('/sair')
def sair():
    session.pop('nome', None)
    session.pop('idade', None)
    resposta = make_response(redirect(url_for('salvar_usuario')))
    resposta.set_cookie('nome', '', max_age=0) # Exclui o cookie
    resposta.set_cookie('idade', '', max_age=0) # Exclui o cookie
    return resposta
```

## 2. **Objetivo deste passo:** Definir rotas para:

- Salvar dados do usuário em uma sessão e em cookies.
- Exibir as informações salvas ao recarregar a página, recuperando dados da sessão ou do cookie.
- Remover dados de sessão e cookies quando o usuário escolher sair.

## Passo 5: Testando a Aplicação

### 1. No terminal, execute o arquivo `app.py` para iniciar a aplicação:

```
python app.py
```

2. Acesse <http://127.0.0.1:5000/usuario> no navegador:
  - Preencha o formulário com o nome e idade e envie.
  - Observe que as informações inseridas aparecem na página após o envio.
3. Atualize a página para ver que as informações são recuperadas automaticamente, mesmo sem resubmeter o formulário.
4. Clique em "Sair" para limpar as informações da sessão e cookies e verifique que os dados foram removidos.

---

## Conclusão do Exercício

Este exercício mostrou como:

1. Utilizar sessões para armazenar dados temporários do usuário.
2. Utilizar cookies para armazenar dados persistentes no navegador do usuário.
3. Criar e renderizar formulários, além de processar dados `POST` e `GET`.

**Desafio Extra:** Adicione um campo "Email" ao formulário e salve-o na sessão e no cookie. Exiba as informações completas do usuário (nome, idade e email) ao recarregar a página.