

Protocolo HTTP: Formulários - Request

Protocolo HTTP: Formulários - Request

👉 Objetivo: compreender o funcionamento de request/response de forma prática utilizando formulários HTML.

👉 Conteúdo:

👉 Servidor http `http.server`

👉 Request

👉 `<form>`

👉 Métodos `GET` e `POST`

👉 Mensagens de `status`

Formulários

Formulários

- As requisições podem ser do tipo `POST` e `GET`. Há outros verbos HTTP que serão discutidos posteriormente (Veja [aqui](#) os demais).
- Uma forma muito comum de enviar requisições `POST` é via **Forms HTML** (primeira que vamos utilizar).
- A tag (elemento HTML) `<form> </form>` permite o envio de requisições tanto `POST` quanto `GET`.
- Quando precisamos realizar algum tipo de alteração no servidor, utilizamos `POST`: Login/logout de usuário é um exemplo

Formulários

↪ Podemos utilizar formulários para envio de requisições do tipo `GET`, entretanto se torna mais interessante quando definimos para `POST`.

Exemplo

```
<form method="POST">  
  <label>Nome</label>  
  <input type="text" placeholder="nome">  
</form>
```

↪ O atributo utilizado é `method` que pode ser tanto `GET` quanto `POST`.

↪ Nos dois casos temos utilidades como veremos.

Formulários

- Quando vamos enviar dados para o servidor, podemos definir como os dados serão codificados.
 - O atributo `enctype` é utilizado para especificar tal propriedade.
 - Este atributo é utilizado apenas com método `POST`
- `enctype` suporta:
 - `application/x-www-form-urlencoded`
 - `multipart/form-data`
 - `text/plain`

Formulários

- ↗ O `enctype` padrão é `application/x-www-form-urlencoded`: transforma espaço no símbolo `+` e para caracteres especiais utiliza o código Hexadecimal ASCII.

Exemplo

```
<form action="/" method="POST" enctype="application/x-www-form-urlencoded">  
  <input type="text" placeholder="Texto" name="valor">  
  <button>Enviar</button>  
</form>
```

- ↗ Veja o exemplo completo no link e como utilizá-lo: [Exemplo1](#)

Formulários

- Utilizamos **multipart/form-data** para envio de arquivos (upload):
- O código abaixo ilustra como podemos indicar no formulário que faremos um upload:

Exemplo

```
<form action="/" method="POST" enctype="multipart/form-data">  
  <input type="file" placeholder="filename" name="userfile">  
  <button>Enviar</button>  
</form>
```

- Veja o exemplo completo e como utilizá-lo no link: [Exemplo2](#)

Exemplo

Formulários e Request

Exemplo - Formulários e Request

➤ Vamos ao um exemplo prático utilizando python + Flask.

➤ O código está disponível [aqui](#).

➤ Neste exemplo, vamos ter a definição de um rota simples:

Exemplo

```
@app.route('/')  
def index():  
    method = request.method  
    return f"<h1>Minha primeira aplicação: método HTTP - {method}</h1>"
```

➤ Observe que vamos acessar um objeto `request`

Exemplo - Formulários e Request

- O objeto `request` permite acessar dados da requisição. No exemplo, acessamos o método da requisição e mostramos ele para o usuário que tentou acessar a página.
- Neste exemplo, em particular, utilizamos o `request` para checar apenas o método da requisição.
- Podemos ampliar seu uso e verificar algum dado que tenha sido enviado pelo usuário.
- Para tanto vamos, adicionar mais duas rotas nesta aplicação e ver essa forma de usar o objeto `request`.

Exemplo - Formulários e Request

↪ As novas rotas da aplicação são:

Exemplo

```
@app.route('/cadastro' ,methods=['GET'])
def create():
    return render_template('cadastro.html')

@app.route('/cadastro', methods=['POST'])
def store():
    nome = request.form['nome']
    return f"<h1>Você cadastrou essa pessoa: {nome} </h1>"
```

↪ Observe que indicamos na rota qual método é suportado. Isso é feito através do atributo `methods=[]`.

↪ Incluímos o uso de `render_template` que prepara o HTML para ser enviado.

Exemplo - Formulários e Request

➤ Para melhor entender as rotas do tipo POST, vamos observar o arquivo `cadastro.html` (uma parte).

Exemplo

```
<h1>Cadastro</h1>
<form action="{{url_for('store')}}" method="post">
  <input type="text" name="nome" id="nome">
  <button>Enviar</button>
</form>
```

➤ um atributo muito importante dos formulários é `action`. Nele indicamos, para onde vamos enviar os dados do formulário (`form-data`).

➤ Neste caso, utilizamos uma função do Flask chamada `url_for` com o parâmetro `store`. Assim indicamos qual rota vai receber o formulário.

Resumo

Resumo

- ~ Vimos atributos importantes dos formulários HTML: `enctype`, `method` e `action`.
- ~ Criamos nossa primeira aplicação Flask
- ~ Criamos rotas que suportam POST e GET
- ~ Utilizamos o objeto `request` do Flask
- ~ Utilizamos a função `render_template`
- ~ Analisamos o resultado de requisições no navegador com exemplos explorando `content-type` do protocolo HTTP.

Referência

[Content-type](#)

[Mime-types - Multipurpose Internet Mail Extensions](#)

[HTTP Methods](#)

[Flask](#)