



IN

Curso: Flask  
Aula 2

# Tema da aula:

## Aula 2 - Trabalhando com Dados

### Conteúdo Programático

- O que é um endpoint;
- Rotas estáticas;
- Rotas dinâmicas;
- Explorando as possibilidades do flask;
- Uma visão geral sobre o JSON.



# Arquitetura cliente servidor

É um modelo de aplicação distribuída, ou seja, um mesmo servidor deve ser capaz de responder a diversos clientes.

- Um servidor disponibiliza um recurso ou serviço;
- Os clientes são requerentes de serviços ou recursos dos servidores;
- Os clientes podem ser browsers ou outras aplicações.

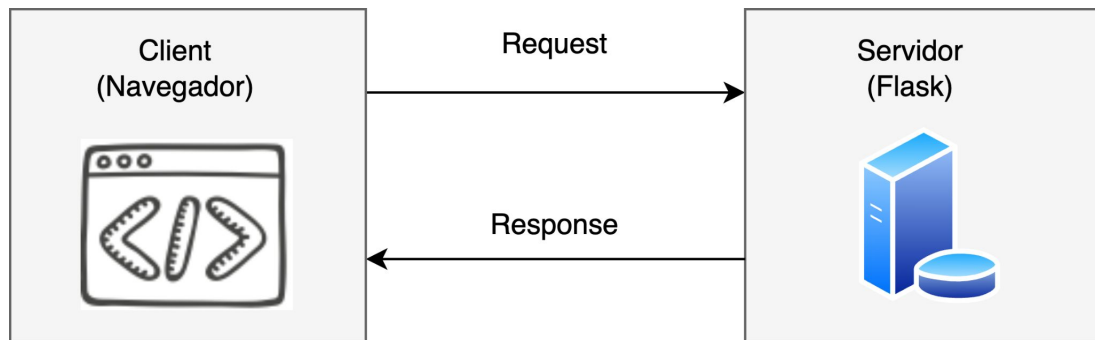


Figura 1: Comunicação do cliente (**Browser**) com o servidor (**Flask**).

# Rotas estáticas



## O que são rotas ?

- Uma interface de programação de aplicativos (API) é a forma pela qual um aplicativo solicita um serviço de outro aplicativo (CloudFlare, 2022);
- As APIs permitem que os desenvolvedores evitem recriar recursos de aplicativos que já existem (CloudFlare, 2022);
- Um *endpoint* da API é o local em que essas solicitações (conhecidas como API) são atendidas (CloudFlare, 2022).

# Rotas estáticas



Um *endpoint* é um caminho para um serviço final, a exemplo da Figura 2, em que a “Aplicação A” faz a requisição da idade de um usuário.

O *endpoint*, por sua vez, faz a consulta na base de dados, processa e retorna a idade.

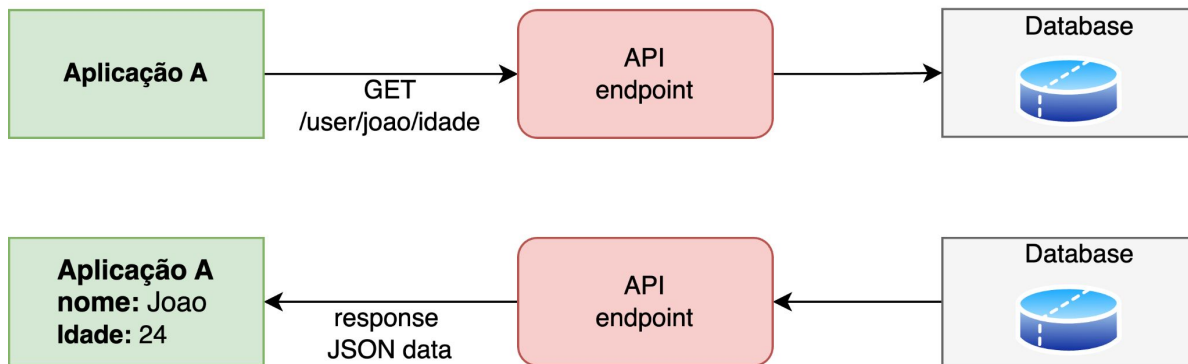


Figura 2: Comunicação de uma aplicação com um *endpoint*.

# Criando uma rota estática



Para criar uma rota estática é necessário indicar no *decorator* o nome da rota, **`@app.route('/titulo_da_pagina')`**. O *decorator* sempre está associado a uma função que vai tratar o retorno esperado da rota.

```
1 from flask import Flask
2 app = Flask(__name__)
3 @app.route('/titulo_da_pagina')
4 def titulo_da_pagina():
5     return 'Meu título'
6 app.run()
```

# Rotas estáticas



1. Crie uma rota estática que retorne o seu nome e idade em formato de string;
2. Crie uma rota estática que retorne o seu nome e idade em uma estrutura de dados JSON (Dicionário no Python).



# Rotas dinâmicas



Para criar uma rota dinâmica é necessário indicar no *decorator* o nome e parâmetros da rota, `@app.route('/v1/users/idade/<nome>')`.

```
1 from flask import Flask
2 app = Flask(__name__)
3 @app.route('/v1/users/idade/<nome>')
4 def retorna_idade(nome: str):
5     if nome == 'Meu título':
6         return { 'idade': 24 }
7     else:
8         return { 'idade': '' }
9 app.run()
```



# Rotas dinâmicas



É possível indicar o método ('POST', 'GET', 'DELETE'...) aceito, usando o parâmetro ***methods***=['GET'].

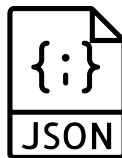
```
1 from flask import Flask
2 app = Flask(__name__)
3 @app.route('/v1/users/idade/<nome>', methods=['GET'])
4 def retorna_idade(nome: str):
5     if nome == 'Meu título':
6         return { 'idade':24 }
7     else:
8         return { 'idade':' ' }
9 app.run()
```

# Vamos a prática



1. Crie uma lista fictícia de usuários. Em cada posição da lista deve conter um dicionário com dados (nome, e-mail, idade) de usuários fictícios.
2. Crie um *endpoint* (**v1/user/idade/<nome>**) que viabilize a consulta de uma idade, vinda de um nome que está na sua lista de usuários. Caso não esteja, retorne uma mensagem informando que o usuário não existe.
3. Crie um *endpoint* (**v1/user/email/<nome>**) que viabilize a consulta de um e-mail, vindo de um nome que está na sua lista de usuários. Caso contrário, retorne uma mensagem informando que o usuário não existe.
4. Crie um *endpoint* que possibilite uma consulta por nome ou e-mail na lista de usuário e retorne os dados do usuário, contido na lista.

# Uma visão geral sobre o JSON



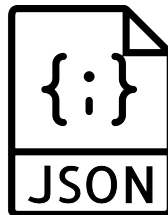
JSON é uma estrutura de dados comumente usada em aplicações WEB e API. A principal vantagem é a simplicidade da linguagem que possibilita a aplicação em diversas linguagens.

- **chaves** "{" }" São para delimitar os objetos, iniciar e encerrar a estrutura;
- **colchetes** "[" ]" para indicar um array (lista do python);
- **dois pontos** ":" para separar a chave e seu valor correspondente;
- **vírgula** "," para separar elementos.

1

```
{ nome": "joao" }
```

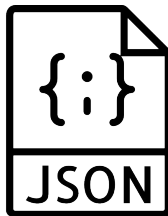
# Arrays em JSON



Um array simples

```
{  
  "nomes": ["Maria", "João", "Francisco"]  
}
```

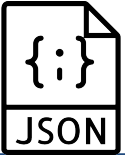
# Objetos em JSON



Um objeto.

```
{  
  "nome": "Joao",  
  "idade": 24,  
  "email": "joao@email.com"  
}
```

# Lista de objetos em JSON



```
{
  "users" : [
    {
      "nome": "Joao",
      "idade": 24
    },
    {
      "nome": "Francisco",
      "idade": 27
    }
  ]
}
```



IN

Obrigado!

The logo consists of the letters 'IN' in a white, serif font, centered within a solid red square. The background of the entire image is a vibrant red with abstract, geometric shapes and lines in varying shades of red, creating a sense of depth and movement.

# IN

71 3901 1052 | 71 9 9204 0134  
@infinity.school

[www.infinityschool.com.br](http://www.infinityschool.com.br)

Salvador Shopping Business | Torre Europa Sala  
310 Caminho das Árvores, Salvador - BA CEP:  
40301-155