



Objetivo:

Exercitar Node, NPM e a biblioteca externa Axios

Problema: Utilizando a linguagem JavaScript, extrair informações dos planetas e personagens do universo Star Wars utilizando sua API <https://swapi.dev/>. Os *endpoints* utilizados serão “/planets”, para os planetas, e “/people” para os personagens. Entenda API como uma *interface* que permite acessar alguns serviços. Para acessar esses serviços, precisamos do seu endereço. E *endpoints* é a parte final da URL que permite acessar algum serviço específico API. Por exemplo, ao combinarmos <http://swapi.dev/people> acessamos o serviço que nos retorna os personagens de Star Wars.

Para facilitar a resolução utilizaremos uma biblioteca externa chamada Axios (<https://axios-http.com/>) que irá nos auxiliar extrair informações de servidores remotos. É necessário ter o endereço desse servidor, isto é, a URL (*Uniforme Resource Locator*). Considerando o Star Wars, o endereço é <https://swapi.dev/>. Perceba que o endereço é sempre uma URL.



Tarefas:

1 -) Utilizando Node e axios, escreva um programa que imprima os primeiros 10 personagens do Star Wars

2-) Utilizando Node e axios, escreva um programa que imprima os primeiros 10 planetas do Star Wars

3 -) Utilizando Node e axios, escreva um programa que imprima todos os personagens do Star Wars fornecidos pela SWAPI

4 -) Utilizando Node e axios, escreva um programa que imprima todos os planetas do Star Wars fornecidos pela SWAPI

Dicas:

1. Fique atento a propriedade *next* que a API retorna para permitir navegar pelas demais páginas e fazer novas requisições. Quando o parâmetro *next* for *null* significa que estamos na última página.
2. Use o operador “ponto” (.) para navegar pelas propriedades dos dados de retorno após a requisição.
3. May the force be with you!

Tecnologias utilizadas:

1. Node
 - a. Ambiente que permite executar código JavaScript
2. NPM: Node Package Manager
 - a. Ferramenta que permite gerenciar projetos Node. Em outras palavras automatiza o processo de construir o software, além de gerenciar as dependências externas (bibliotecas)
3. Axios: Biblioteca JavaScript que permite acessar servidores remotos.
 - a. Utilizamos o NPM para fazer a instalação.

Preparando o Ambiente: Se espera um ambiente com terminal UNIX disponível. Podendo ser máquinas Windows + WSL, ou MacOS ou Linux.

É possível que os passos a seguir não funcionem em máquinas Windows que não possuem WSL instalado. Para instalar o WSL siga os passos no link: <https://learn.microsoft.com/pt-br/windows/wsl/install>

1. Instalação do NVM (*Node Version Manager*)

A instalação recomendada do Node é através do NVM (*Node Version Manager*), que é um gerenciador das versões do nodejs. Sua instalação segue a documentação disponível em: <https://github.com/nvm-sh/nvm>

No terminal execute o seguinte código:

```
wget -qO- https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.1/install.sh | bash
```

Em seguida feche e abra o terminal para que as variáveis de ambiente sejam atualizadas. Para verificar a instalação do NVM digite o seguinte comando:

```
command -v nvm
```

A resposta deverá ser “nvm”. Nesse momento é possível instalar o Node.

2. Instalação do Node

Para instalar a última versão do Node, utilize o seguinte comando:

```
nvm install node
```

Para instalar uma versão específica do Node, utilize o seguinte comando:

```
nvm install node <versão>, onde <versão> é um número como 21.0.0
```

Para listar as versões do Node, utilize o seguinte comando:

```
nvm ls, a resposta deverá ser algo como:
```

```
->      v22.2.0
default -> node (-> v22.2.0)
iojs    -> N/A (default)
unstable -> N/A (default)
node    -> stable (-> v22.2.0) (default)
stable  -> 22.2 (-> v22.2.0) (default)
lts/*   -> lts/iron (-> N/A)
lts/argon -> v4.9.1 (-> N/A)
lts/boron -> v6.17.1 (-> N/A)
lts/carbon -> v8.17.0 (-> N/A)
```

Para instalar a versão específica LTS (*long term support*), digite o comando:

```
nvm install --lts,
```

Consulte os demais comandos no README: <https://github.com/nvm-sh/nvm>

3. Instalação do NPM (Node Package Manager)

O NPM já é instalado juntamente com o Node.

Utilizando o Axios

O primeiro passo para utilizar uma biblioteca externa é fazer sua instalação. Para isso utilize o *npm*. Na raiz do seu projeto execute o comando no terminal:

```
npm install axios
```

Caso ainda não possua, os arquivos *package.json* e *package-lock.json* serão criados na raiz do seu projeto. Esse cenário ocorre quando não se inicia um projeto Node com NPM, através do comando *npm init*. Adicionalmente irá surgir a pasta “node_modules”, que contém o código das bibliotecas externas. A cada nova biblioteca instalado, essa pasta irá conter os códigos necessários.

Em seguida, com a biblioteca instalado, precisamos importá-la para o código. Para isso vamos utilizar o *require*.

```
const axios = require('axios');
```

Isso permite invocarmos diversas funcionalidades da biblioteca. A que iremos utilizar é a chamada *get*: Abaixo um exemplo

//A função precisa ter o modificador *async* pois iremos utilizar funções assíncronas

```
const buscaPersonagens = async () => {  
  const api = 'https://swapi.dev/api';  
  const endpoint = '/people';  
  const url = `${api}${endpoint}`;  
  const dados = await axios.get(url); // precisamos de utilizar await pois a chamada get é assíncrona  
  console.log(dados). //Estude esses dados de retorno  
}
```

Execute a função acima e observe a saída, sua tarefa será imprimir os dados solicitados no terminal de acordo com as instruções da tarefa.