

FIC 2 – Módulo II

# Aula 7 – Carregando arquivo HTML utilizando Express

Criação de páginas  
customizadas com Node.js

*Leonardo Dalvi Pio*



INSTITUTO FEDERAL  
Espírito Santo

# Carregando arquivo *HTML* utilizando *Express*

- Carregando arquivos *HTML* utilizando *Express*.
- Carregando conteúdos *HTML* utilizando *Express* direto no Node.
- No Replit, se o pacote/recurso não estiver disponível, ele será baixado automaticamente.

# Carregando arquivo HTML

- No Replit, crie uma pasta nomeada "Aula 7".
- Dentro da pasta, crie um novo arquivo (New Repl). Selecione o Template 'Node.js' e atribua o título: "aula7". O arquivo 'index.js' em 'Files' será incluído.

# Carregando arquivo HTML

- Dentro da pasta, crie um novo arquivo (New Repl) no formato *HTML*, com nome 'arquivo.html'.
- Acesse o site abaixo para gerar o texto 'aleatório':  
<https://www.lipsum.com/>
- Copie um ou dois parágrafos e cole no arquivo.

# Carregando arquivo HTML

- Utilize uma estrutura básica do *HTML*, conforme código abaixo:

```
<html lang="pt-br">
<head>
  <title>Página Html</title>
</head>
<body>
  <h1>Página inicial do site</h1>
  <a href="/contato">Ir para Contato </a><br>
  <a href="/sobre">Ir para Sobre </a>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam pulvinar
ligula id magna suscipit dignissim. Praesent tincidunt, dui ac finibus lobortis, velit
dolor fringilla sapien, id tempor massa orci at est. Lorem ipsum dolor sit amet,
consectetur adipiscing elit. Praesent facilisis semper suscipit. Quisque posuere
sagittis lectus sed sodales.</p>
</body>
</html>
```

# Carregando arquivo HTML

- No arquivo 'index.js', criaremos uma constante nomeada 'express', conforme código:

```
const express = require('express')
```

- Utilizaremos o 'require' (requisição) para acessar o módulo 'express' (importação do módulo 'express').
- Em seguida criaremos uma aplicação *Express*. Esse objeto (tradicionalmente nomeado de 'app') tem métodos de roteamento de requisições *HTTP* e várias outras configurações e recursos.

```
const app = express()
```

# Carregando arquivo HTML

- Utilizaremos a porta 443.

```
const porta = 443
```

- O método `'app.get()'` especifica uma função de retorno de chamada que será invocada sempre que exista uma solicitação *HTTP GET* com um caminho `('/')` relativo à raiz do site (Dentro do servidor web). A função de retorno de chamada requer uma solicitação e um objeto de resposta como argumentos e, simplesmente, chama `'send()'` na resposta para retornar a string "Contato".

```
app.get("/contato", function(req, res) {  
  res.send('Contato')  
})
```



# Carregando arquivo HTML

- O mesmo processo ocorre quando se utiliza: 'Sobre'. A função de retorno de chamada requer uma solicitação e um objeto de resposta como argumentos e, simplesmente, chama 'send()' na resposta para retornar a *string* "Sobre".

```
app.get("/sobre", function(req, res) {  
    res.send('Sobre')  
})
```



# Carregando arquivo HTML

- Para se enviar um arquivo *HTML*, quando requisitado, utilizaremos o comando 'sendFile' como resposta (res), utilizando o código abaixo:

```
app.get("/", function(req, res) {  
  res.sendFile(__dirname + '/pagina.html')  
})
```

- No comando acima, devemos informar o caminho do arquivo completo (diretório + arquivo). Informando (além do nome do arquivo) o caminho/diretório do arquivo (no nosso caso, o arquivo 'pagina.html' se encontra na raiz do projeto). Utilizaremos o comando '\_\_dirname' para apontar o diretório.

# Carregando arquivo HTML

- Para finalizar, incluiremos o comando `'.listen()'` e a porta de comunicação (porta 443) com o Servidor, apresentando a mensagem: `'Servidor rodando'` (com o comando: `console.log`).

```
app.listen(porta, () => { console.log('Servidor rodando') })
```

- Pronto! Já conseguimos acessar e carregar arquivos HTML!

# Carregando arquivo HTML

Codificação final:

```
1  const express = require('express')
2  const app = express()
3  const porta = 443
4
5  app.get("/contato", function(req, res) {
6    res.send('Contato')
7  })
8
9  app.get("/sobre", function(req, res) {
10    res.send('Sobre')
11  })
12
13 app.get("/", function(req, res) {
14   res.sendFile(__dirname + '/pagina.html')
15 })
16
17 app.listen(porta, () => { console.log('Servidor rodando') })
```

# Testes!

- Para testar o código, inicie com o botão 'Run' do Replit.
- Na aba 'Console' é possível observar a mensagem: "Servidor rodando"!
- Na aba 'Webview' é possível observar a página inicial do site, carregando o arquivo *HTML* criado.
- Abra uma nova aba do código no *browser*.
- Teste a página 'Sobre', incluindo "/" ao final do URL: "/sobre".
- Faça o mesmo para 'Contato', incluindo "/" ao final do URL: "/contato".



**INSTITUTO  
FEDERAL**  
Espírito Santo