

FIC 2 – Módulo II

# Aula 8 – Criando rotas no Servidor Web

Criação de páginas  
customizadas com Node.js

*Leonardo Dalvi Pio*



INSTITUTO FEDERAL  
Espírito Santo

# Criando rotas no Servidor Web

- Criação de um servidor web.
- Criação de rotas no servidor web.
- No Replit, se o pacote/recurso (***pagkage.json***, ***express***) não estiver disponível, ele será baixado automaticamente.

# Criando rotas no Servidor Web

- No Replit, crie uma pasta nomeada "Aula 8".
- Dentro da pasta, crie um novo arquivo (New Repl).  
Selecione o Template 'Node.js' e atribua o título: "aula8". O arquivo 'index.js' em 'Files' será incluído.

# Criando rotas no Servidor Web

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

```
const http = require('http')
```

- Utilizaremos a porta 443.

```
const porta = 443
```

- Em seguida, com o objeto 'http', utilizaremos a função 'createServer()'.
- Utilizaremos a *Arrow Functions*. Passaremos às devidas variáveis: **req** (requisição) e **res** (resposta). As *Arrow Functions* são utilizadas por: `=>`.

```
const servidor = http.createServer((req, res) => {
```

# Criando rotas no Servidor Web

- Na nossa *Arrow Functions* criada, começaremos com a resposta (res). Utilizaremos o `'writeHead(status:Code, reasonPhrase?)'`, com código 200 (conforme Página [http.cat – status ok](http://cat.status)) e cabeçalho `'Content-Type':'text/html'`.

```
res.writeHead(200, { 'Content-Type': 'text/html' })
```

- 
- Iniciaremos as rotas com o uso da requisição (**req**). Verificaremos o URL conforme código abaixo:

```
if(req.url == '/'){  
  res.write('<h1>Seja bem-vindo! </h1>')  
}
```

- Como inserimos 'text/html', poderemos utilizar *tags HTML*.
- Se o caminho informado for a página principal (raiz, '/'), será apresentada a resposta (res): "Seja bem-vindo!"

# Criando rotas no Servidor Web

- Continuando a criação de rotas, se o caminho informado for: `'/server'`, será apresentada a resposta (res): "Server".
- Se o caminho informado for: `'/server/node'`, será apresentada a resposta (res): "Server Web no Node".

```
else if(req.url == '/server'){
    res.write('<h1>Server </h1>')
}
else if(req.url == '/server/node'){
    res.write('<h1>Server Web no Node </h1>')
}
```

- E, por fim, se não for informado uma dos URLs acima (`'/'`; `'server'`; e `'/server/node'`), será apresentada a resposta (res): "PADRÃO". Finalizamos com o comando `'end()'`.

```
else {
    res.write('<h1>PADRAO </h1>')
}
res.end()
```



# Criando rotas no Servidor Web

- 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`).

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

- Pronto! Já temos um Servidor Web com Rotas (URLs)!

# Criando rotas no Servidor Web

Codificação final:

```
1  const http = require('http')
2  const porta = 443
3
4  const servidor = http.createServer((req, res) => {
5      res.writeHead(200, {'Content-Type': 'text/html'})
6      if(req.url == '/'){
7          res.write('<h1>Seja bem-vindo! </h1>')
8      }
9      else if(req.url == '/server'){
10         res.write('<h1>Server </h1>')
11     }
12     else if(req.url == '/server/node'){
13         res.write('<h1>Server Web no Node </h1>')
14     }
15     else {
16         res.write('<h1>PADRAO </h1>')
17     }
18     res.end()
19 })
20
21 servidor.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"!
- Abra uma nova aba do código no browser. (ex: <https://aula8.usuario.repl.co/>)
- Teste as rotas criadas:
  - '/' (Raiz) ➡ "Seja bem Vindo!"
  - '/server' ➡ "Server"
  - '/server/node' ➡ "Server Web no Node"
  - 'teste' ➡ "PADRÃO"



**INSTITUTO  
FEDERAL**  
Espírito Santo