

FIC 2 – Módulo II

# Aula 9 – Criando rotas no Servidor Web com uso do módulo URL

Criação de páginas  
customizadas com Node.js

*Leonardo Dalvi Pio*



**INSTITUTO FEDERAL**  
Espírito Santo

# Criando rotas no Servidor Web com uso do módulo *URL*

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

# Criando rotas com uso do módulo URL

- No Replit, crie uma pasta nomeada "Aula 9".
- Dentro da pasta, crie um novo arquivo (New Repl).  
Selecione o Template 'Node.js' e atribua o título: "aula6". O arquivo 'index.js' será incluído em 'Files'.

# Criando rotas com uso do módulo URL

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

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

- Utilizaremos a porta 443.

```
const porta = 443
```

- Criaremos uma constante acessando o módulo 'url'.

```
const url = require('url')
```

- 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 com uso do módulo URL

- 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' })
```

- A constante 'url' (do módulo 'url') foi criada para receber a requisição (de parâmetros) via URL do Browser.
- Criaremos uma resposta (res) escrevendo o URL informado (req) na barra de endereços diretamente na página do servidor.

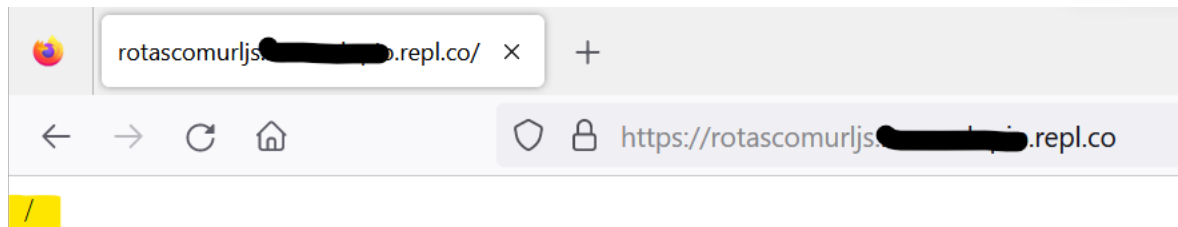
```
res.write(req.url)
```

- Indicamos o final da nossa resposta (res) com o comando `end()`.

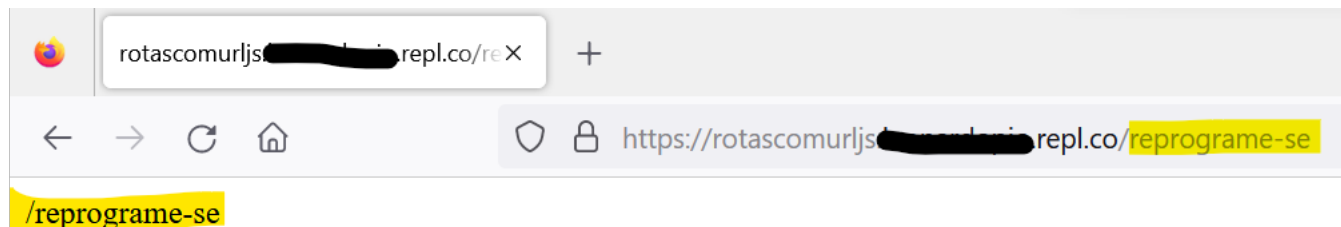
```
res.end( )
```

# Criando rotas com uso do módulo URL

- Vá na aba 'Webview', selecione para abrir no Browser (Open in a new tab).
- Abaixo é possível observar a "/" quando acessamos a rota (URL) padrão (raiz).



- Se digitarmos "reprograme-se" no URL, conforme abaixo, observaremos que o conteúdo informado (req) foi escrito (res) no HTML.



# Criando rotas com uso do módulo URL

- Com esse recurso do módulo URL, é possível trabalhar conteúdos não somente como rotas, mas sim como parâmetros (entrada – req) para sua aplicação Node.
- Para trabalhar com a requisição, criaremos uma constante 'parametro' que tratará o conteúdo não somente como uma simples rota, e sim como parâmetros. Da nossa constante 'url' criada (req.url), transformaremos o URL em *query string* (consulta de string), utilizando o comando 'parse()'.
- O código abaixo exhibe a constante 'parametro' recebendo os parâmetros de 'url'.

```
const parametro =url.parse(req.url,true).query
```



# Criando rotas com uso do módulo URL

- Informaremos agora os parâmetros que gostaríamos de obter na nossa aplicação. Vamos enviar nome, sobrenome e cidade de uma pessoa como parâmetros via URL no nosso servidor.
- Utilizaremos o código abaixo para definirmos os parâmetros desejados:

```
res.write('<br/>'+parametro.nome)  
res.write('<br/>'+parametro.sobrenome)  
res.write('<br/>'+parametro.cidade)
```

- Criaremos um *HTML* ('<br/>') concatenado com o parâmetro desejado para melhor visualização no nosso servidor.
- 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') })
```





# Criando rotas com uso do módulo URL

- Pronto! Já temos um Servidor Web com uso do módulo 'url'!

# Criando rotas com uso do módulo URL

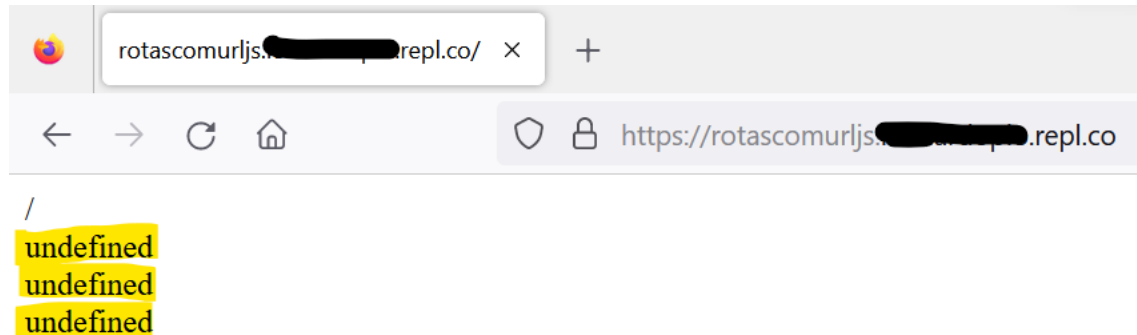
Codificação final:

```
1  const http = require('http')
2  const porta = 443
3  const url = require('url')
4
5  const servidor = http.createServer((req, res) => {
6    res.writeHead(200, {'Content-Type': 'text/html'})
7    res.write(req.url)
8    const parametro = url.parse(req.url, true).query
9    res.write('<br/>' + parametro.nome)
10   res.write('<br/>' + parametro.sobrenome)
11   res.write('<br/>' + parametro.cidade)
12   res.end()
13 })
14
15 servidor.listen(porta, () => { console.log('Servidor Rodando') })
```



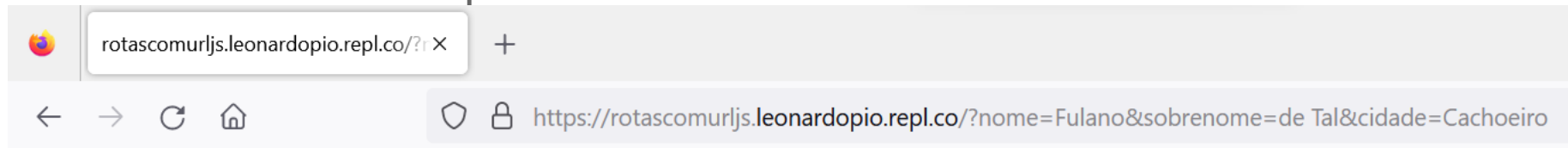
# Testes!

- 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://rotascomurljs.usuario.repl.co/>)
- Você observará a seguinte mensagem:



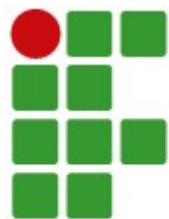
# Testes!

- Para enviarmos os parâmetros que criamos (nome, sobrenome e cidade), devemos informar o código diretamente no URL.
- Primeiramente, informaremos o '/' seguido de '?' com o parâmetro criado e seu conteúdo, conforme abaixo:
- **/?nome=Fulano**
- Para incluir o parâmetro 'sobrenome', acrescentaremos o '&' a cada novo parâmetro. O código completo é apresentado abaixo:
- **?nome=Fulano&sobrenome=de Tal&cidade=Cachoeiro**
- Na página do nosso servidor, é possível observar o URL e os conteúdos dos parâmetros informados.



/?nome=Fulano&sobrenome=de%20Tal&cidade=Cachoeiro

Fulano  
de Tal  
Cachoeiro



**INSTITUTO  
FEDERAL**  
Espírito Santo