

## Trabalhando com o framework

O Express é um framework para aplicativo da web do Node.js mínimo e flexível que fornece um conjunto robusto de recursos para aplicativos web e móvel.

Agora instale o Express no diretório app e salve-o na lista de dependências. Por exemplo:

```
$ npm install express --save
```

Para instalar o Express temporariamente não o inclua na lista de dependências, omita a opção `--save`:

```
$ npm install express
```

## Exemplo Hello World usando express

Crie um novo projeto chamado app.js com seguinte código.

```
var express = require('express');
var app = express();

app.get('/', function (req, res) {
  res.send('Hello World!');
});

app.listen(3000, function () {
  console.log('Example app listening on port 3000!');
});
```

Execute o aplicativo com o seguinte comando:

```
$ node app.js
```

Em seguida, carregue <http://localhost:3000/> em um navegador para visualizar a saída

O aplicativo inicia um servidor e escuta a porta 3000 por conexões. O aplicativo responde com “Hello World!” à solicitações para a URL raiz (/) ou **rota**. Para todos os outros caminhos, ele irá responder com um **404 Não Encontrado**.

<https://expressjs.com/pt-br/starter/hello-world.html>

## Roteamento Básico

O **Roteamento** refere-se à determinação de como um aplicativo responde a uma solicitação do cliente por um endpoint específico, que é uma URI (ou caminho) e um método de solicitação HTTP específico (GET, POST, e assim por diante).

Cada rota pode ter uma ou mais funções de manipulação, que são executadas quando a rota é correspondida. A definição de rotas aceita a seguinte estrutura

```
app.METHOD(PATH, HANDLER)
```

Onde:

- app é uma instância do express.
- METHOD é um método de solicitação HTTP.
- PATH é um caminho no servidor.
- HANDLER é a função executada quando a rota é correspondida.

## Roteamento Básico

Responder com Hello World! na página inicial:

```
app.get('/', function (req, res) {  
  res.send('Hello World!');  
});
```

Responder a uma solicitação POST na rota raiz (/) com a página inicial do aplicativo:

```
app.post('/', function (req, res) {  
  res.send('Got a POST request');  
});
```

## Roteamento Básico

Responder a uma solicitação PUT para a rota /user:

```
app.put('/user', function (req, res) {  
  res.send('Got a PUT request at /user');  
});
```

Responder a uma solicitação DELETE para a rota /user:

```
app.delete('/user', function (req, res) {  
  res.send('Got a DELETE request at /user');  
});
```

## Roteamento Básico renderizando arquivos HTML

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

app.get("/", function(req, res){
  res.sendFile(__dirname + "/html/index.html");
});

app.get("/sobre", function(req, res){
  res.sendFile(__dirname + "/html/sobre.html");
});

app.get('/blog', function(req, res){
  res.send("Bem vindo ao meu blog!");
});

app.listen(8081, function(){console.log("Servidor Rodando!");});
```

## Exercício – refaça o exercício anterior usando o express

Crie 3 arquivos HTML: `artigos.html`, `contato.html` e `erro.html`

Coloque qualquer conteúdo para cada pagina html;

Ao digitar no browser o path: `/artigos` deve renderizar **`artigos.html`**

A regra anterior também se aplica para o arquivo **`contato.html`**;

Ao digitar qualquer path diferente de `/artigos` e `/contato` deve renderizar **`erro.html`**;

A rota principal `"/` deve renderizar **`artigos.html`**;

**Tempo: 40 minutos**