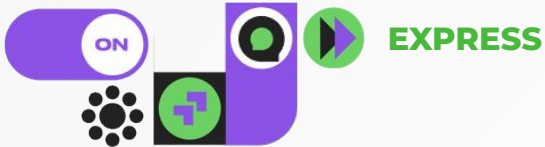


A decorative graphic on the left side of the slide. It features a vertical stack of Scratch blocks and icons. From top to bottom: a green flag icon, a green sun-like icon, a green eye icon, a green mustache icon, a green speech bubble icon, a green chess knight icon, a green hand icon, a green circular arrow icon, a green circle icon, a green heart icon, a green square icon, a green circle with the text 'ON', and a green wavy line.

Express - Introdução

Express I



Express

Framework web rápido, flexível e minimalista para Node.js

Consulte a documentação
<https://expressjs.com/pt-br/>

Express

Framework web rápido, flexível e minimalista para Node.js

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

Aplicativos da Web

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.

APIs

Com uma miríade de métodos utilitários HTTP e middleware a seu dispor, criar uma API robusta é rápido e fácil.

Desempenho

O Express fornece uma camada fina de recursos fundamentais para aplicativos da web, sem obscurecer os recursos do Node.js que você conhece e ama.

LoopBack

[Desenvolva aplicativos orientado a modelos com um framework baseado em Express.](#)
[Descubra mais em loopback.io.](#)

KEYNOTE: Express, State of the Union by ...
node interactive
KEYNO xpress, State of the Union
Doug Wilson, Express

Iniciando projeto com Express

Crie uma pasta para o projeto, e através do terminal, execute os seguintes comandos:

01. Iniciando o projeto node

```
> npm init -y
```

Esse comando vai gerar o arquivo `package.json` dentro da sua pasta

02. Instalando Express

```
> npm install express
```

A pasta **node_modules** deve aparecer na sua pasta

03. Crie um arquivo chamado `app.js`

```
> touch app.js
```

O arquivo pode ser criado manualmente ou através do terminal

Criando o servidor

No arquivo app.js

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

app.listen(3000, () => {
  console.log("Servidor rodando na porta 3000")
});
```

Importação do módulo **express**

Inicializando express

app.listen inicia o servidor na porta indicada no primeiro parâmetro

Executando o servidor

No terminal, execute o seguinte comando:

```
> node app.js
```

Utilize o mesmo nome do arquivo
que você criou o servidor

Deve aparecer a mensagem do console.log:

```
> node app.js  
Servidor rodando na porta 3000
```

A primeira rota

No arquivo app.js

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

app.get('/', (req, res) => {
  res.send('Olá, mundo!')
})

app.listen(3000, () => {
  console.log("Servidor rodando na porta 3000")
});
```

Adicione esse trecho para criar uma rota simples que retorna o texto "Olá, mundo!"

Não se preocupe em entender todo o código agora, vamos explorar cada detalhe nas próximas aulas :)

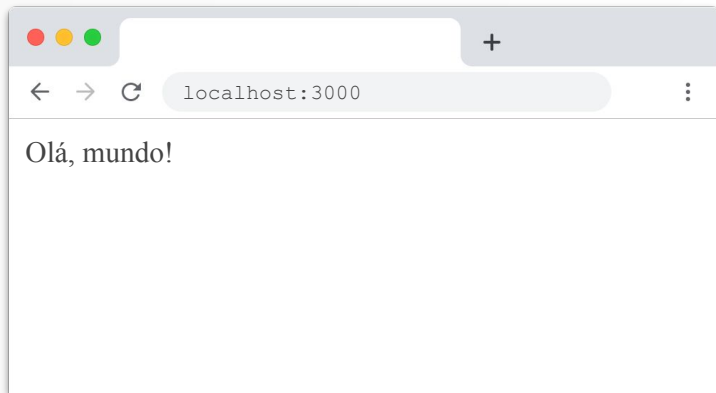
Testando no navegador

Execute a url no navegador

```
localhost:3000
```

localhost significa que está localizado na nossa máquina,
3000 é a porta que o servidor está rodando

Resultado



Nível desbloqueado

Sua primeira página rodando em aplicação Node.js com Express, que orgulho!

