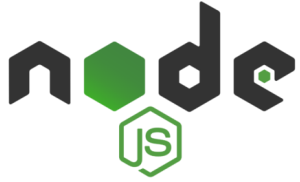


Tecnologia Web III





- Node.js é um ambiente de tempo de execução JavaScript de plataforma cruzada e código aberto que pode ser executado em Windows , Linux , Unix , macOS e muito mais.
- O Node.js é executado no mecanismo JavaScript V8 e executa código JavaScript fora de um navegador da web.
- Node.js permite que os desenvolvedores criem aplicativos front-end e back-end usando JavaScript. Foi lançado em 2009 por Ryan Dahl.

<https://nodejs.org/en>



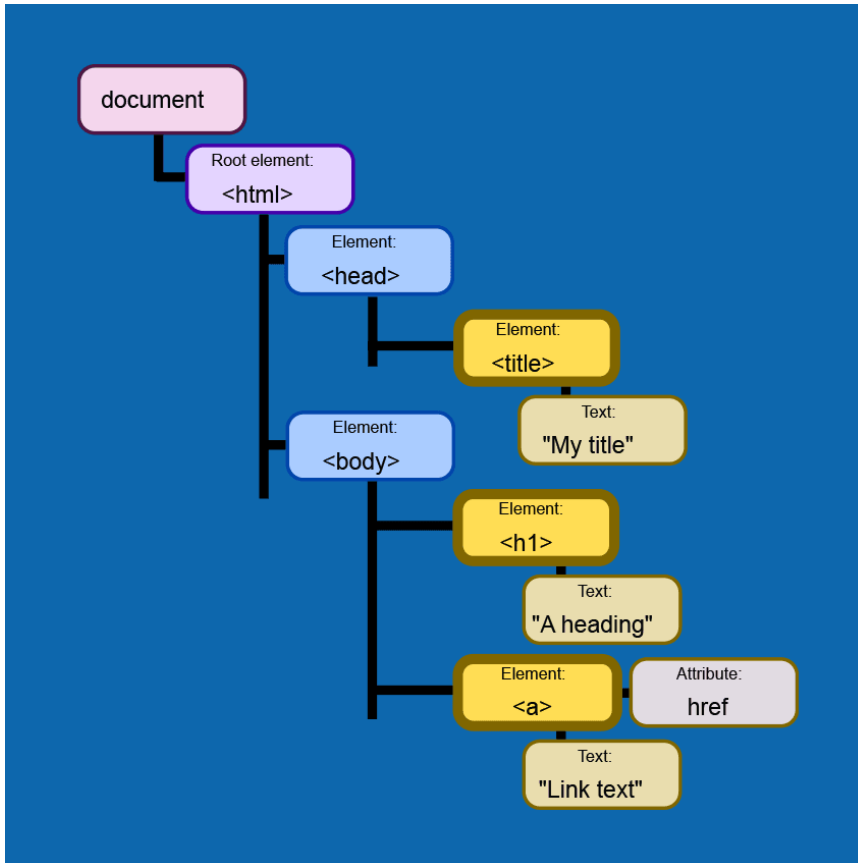
- Node.js é de código disponível publicamente. E é mantido por colaboradores de todo o mundo.
- Node.js é multiplataforma e não depende de nenhum software de sistema operacional. Pode funcionar em Linux, macOS, Windows ou containers.
- Node.js é um ambiente de tempo de execução JavaScript: quando você escreve código JavaScript em seu editor de texto, esse código não pode executar nenhuma tarefa a menos que você o execute. Para executar seu código, você precisa de um ambiente de execução.



- Navegadores como Chrome e Firefox possuem ambientes de execução. É por isso que eles podem executar código JavaScript. Antes da criação do Node.js, o JavaScript só podia ser executado em um navegador. E foi usado para construir apenas aplicativos front-end.
- Node.js fornece um ambiente de execução fora do navegador baseado no mecanismo JavaScript do Chrome V8 . Isso torna possível construir aplicativos de back-end usando a mesma linguagem de programação JavaScript.

Diferenças entre execução de JavaScript no navegador e do Node.js

Acesso às APIs DOM



- Em tempo de execução do navegador, você pode acessar o Document Object Model (DOM). E você pode realizar todas as operações do DOM.
- Diferentemente do JavaScript no navegador, onde o DOM representa a estrutura da página web e permite interações com elementos HTML, CSS e eventos da página, o Node.js não possui acesso direto ao DOM, uma vez que não está sendo executado em um ambiente de navegador.
- No entanto, existem bibliotecas como o jsdom que permitem simular o DOM em ambiente Node.js para testes e outras finalidades específicas, mas nativamente o Node.js não tem acesso direto ao DOM como um navegador web.

Diferenças entre execução de JavaScript no navegador e do Node.js **window vs objeto global**

- JavaScript possui um objeto global integrado.
- O objeto global JavaScript para o navegador é chamado de objeto ***window***.
- No Node.js, o objeto global atende pelo nome ***global***.
- O objeto ***window*** contém métodos e propriedades disponíveis apenas no ambiente do navegador.



Objeto *global* do Node.js

O objeto ***global*** contém propriedades e métodos que estão disponíveis em todos os módulos do Node.js sem a necessidade de importá-los explicitamente:

- *global*: Referência ao objeto global.
- *console*: Para exibir mensagens no console.
- *process*: Fornece informações e controle sobre o processo em execução.
- *require()*: Função para importar módulos em um arquivo.
- *module*: Representa o módulo atual.
- *__dirname*: Caminho do diretório do arquivo atual.
- *__filename*: Caminho do arquivo atual.
- *setTimeout()*: Função para agendar a execução de uma função após um determinado tempo.
- *setInterval()*: Função para agendar a execução de uma função em intervalos regulares.

A Biblioteca npm (Node Package Manager)

- A biblioteca contém um registro de mais de um milhão de pacotes. Um pacote são trechos de código reutilizável.
- Pode criar um pacote para uma tarefa ou problema recorrente e compartilhar o código com outras pessoas por meio do registro.
- Pode baixar pacotes que outras pessoas compartilharam. Para muitas tarefas que os desenvolvedores realizam regularmente, existem pacotes disponíveis para isso.

A Biblioteca **npm**

NPM é o maior registro de software do mundo. Desenvolvedores de código aberto de todos os continentes usam o **npm** para compartilhar e emprestar pacotes, e muitas organizações também usam o **npm** para gerenciar o desenvolvimento privado. **npm** consiste em três componentes distintos:

- Use o site <https://www.npmjs.com/> para descobrir pacotes, configurar perfis e gerenciar outros aspectos do **npm**. Por exemplo, você pode configurar organizações para gerenciar o acesso a pacotes públicos ou privados.
- A CLI é executada em um terminal e é como a maioria dos desenvolvedores interage com o **npm**.
- A URL de registro usada é determinada pelo escopo do pacote. Se nenhum escopo for especificado, o registro padrão será usado, que é fornecido pelo parâmetro `registry`. Também é utilizado para resolver pacotes por nome e versão, o **npm** se comunica com um site de registro

Exemplos de bibliotecas do npm

- **Express:** Um framework web rápido, flexível e minimalista para Node.js, utilizado para construir aplicações web e APIs.
 - `npm install express`
- **React:** Uma biblioteca JavaScript para construir interfaces de usuário interativas.
 - `npm install react`
- **Lodash:** Uma biblioteca de utilitários JavaScript que simplifica a manipulação de arrays, objetos, strings, etc.
 - `npm install lodash`
- **Mongoose:** Uma ferramenta de modelagem de objetos MongoDB para Node.js que fornece uma solução baseada em esquemas para modelar os dados da aplicação.
 - `npm install mongoose`
- **Axios:** Um cliente HTTP baseado em Promises para fazer requisições HTTP no navegador e Node.js.
 - `npm install axios`
- **Socket.io:** Uma biblioteca que permite comunicação em tempo real entre o cliente e o servidor.
 - `npm install socket.io`

Diferença principal entre npm e npx

npm (Node Package Manager)

- O npm é o gerenciador de pacotes do Node.js, usado para instalar, compartilhar e gerenciar dependências de um projeto.
- Com o npm, você pode instalar pacotes localmente (dentro do diretório do projeto) ou globalmente (em todo o sistema).
- Além de instalar pacotes, o npm é usado para executar scripts definidos no arquivo package.json usando o comando `npm run <script-name>`.

```
"scripts": {
```

```
  "start": "node server.js"
```

```
}
```

```
npm start
```

npx

- O npx é um executável que vem junto com o npm a partir da versão 5.2.0 em diante.
- Ele é usado para executar pacotes npm temporariamente, sem a necessidade de instalá-los globalmente ou localmente.
- Com o npx, você pode executar comandos de pacotes npm diretamente da linha de comando, sendo útil para executar ferramentas de linha de comando específicas sem instalá-las no sistema.

```
npx create-react-app my-app
```

Glossario

- DOM: DOM significa Document Object Model e é a representação de dados de um documento da web, como uma página HTML, um gráfico SVG ou um arquivo XML. DOM é um conjunto de objetos (Document Object Model) que constituem uma página da web. Cada objeto é a implementação real de uma interface – o modelo que descreve a aparência de um objeto. Assim, uma interface pode ser implementada por vários objetos.