



# HELLO WORD! REACT.JS

## DOCUMENTAÇÃO DO REACT

## HISTÓRIA DO REACT JS

## CONCEITOS BÁSICOS

### O que é o React?

- O React é uma biblioteca JavaScript que permite aos desenvolvedores criar interfaces de usuário com componentes reutilizáveis. Ele foi projetado para facilitar o desenvolvimento de aplicativos da web escaláveis, onde os componentes podem ser compostos para formar interfaces complexas.

### Por que usar React?

- Há várias razões para usar o React:
  1. **Componentização:** O React promove a construção de aplicativos da web usando componentes reutilizáveis. Isso ajuda a manter o código organizado e facilita a manutenção e escalabilidade do projeto.
  2. **Virtual DOM:** O React utiliza um conceito chamado Virtual DOM para otimizar o desempenho das atualizações na interface do usuário. Em vez de atualizar o DOM diretamente, o React atualiza uma

representação virtual do DOM e compara com o DOM real, aplicando apenas as alterações necessárias. Isso resulta em uma experiência do usuário mais rápida e responsiva.

3. **Declarativo:** O React permite que os desenvolvedores criem interfaces de usuário declarativamente, descrevendo como a interface deve ser exibida em diferentes estados da aplicação. Isso torna o código mais previsível e fácil de entender.
4. **Ecossistema robusto:** O React possui um ecossistema rico de bibliotecas e ferramentas, como o React Router para roteamento, Redux para gerenciamento de estado, e muitos outros.
5. **Suporte da comunidade:** O React é amplamente adotado pela comunidade de desenvolvimento web, o que significa que há uma abundância de recursos, tutoriais e suporte disponíveis.

## O que vou aprender com o React?

- Ao aprender React, você ganhará conhecimento sobre:
  1. **Componentização:** Aprenderá como dividir sua interface de usuário em componentes reutilizáveis e como compô-los para construir interfaces complexas.
  2. **Estado e Ciclo de Vida:** Aprenderá sobre o conceito de estado e como gerenciá-lo dentro de componentes React. Também entenderá o ciclo de vida dos componentes e como usar métodos de ciclo de vida para realizar tarefas específicas.
  3. **Renderização condicional:** Aprenderá como renderizar componentes condicionalmente com base no estado da aplicação.
  4. **Gerenciamento de eventos:** Aprenderá como lidar com eventos de usuário, como cliques e teclas pressionadas.
  5. **Integração com APIs:** Aprenderá como integrar aplicativos React com APIs externas para recuperar e enviar dados.
  6. **Roteamento:** Aprenderá como configurar o roteamento em aplicativos React para navegação entre diferentes páginas.

# PASSOS PARA RODAR O REACT NO SEU AMBIENTE DE TRABALHO:

1. **Instalar o Node.js e o npm:** O React utiliza o Node.js para executar o código JavaScript no servidor. Você pode baixar e instalar o Node.js no site oficial: <https://nodejs.org/>. O npm (Node Package Manager) é instalado automaticamente junto com o Node.js.
2. **Instalar o Create React App (opcional, mas recomendado):** O Create React App é uma ferramenta que facilita a criação de novos projetos React com uma estrutura de diretórios e configurações predefinidas. Você pode instalar o Create React App globalmente executando o seguinte comando no terminal:

```
npm install -g create-react-app
```

3. **Criar um novo projeto React:** Com o Create React App instalado, você pode criar um novo projeto React executando o seguinte comando no terminal:

```
npx create-react-app nome-do-seu-projeto
```

Substitua `nome-do-seu-projeto` pelo nome desejado para o seu projeto.

4. **Navegar para o diretório do projeto:** Depois de criar o projeto, navegue para o diretório recém-criado:

```
cd nome-do-seu-projeto
```

5. **Executar o projeto:** Agora você pode iniciar o servidor de desenvolvimento executando o seguinte comando no terminal:

```
npm start
```

Isso iniciará um servidor de desenvolvimento e abrirá automaticamente uma janela do navegador com o seu aplicativo React em execução. Qualquer alteração que você fizer nos arquivos do projeto será automaticamente recarregada no navegador.

6. **Explorar o projeto:** Agora que o projeto está em execução, você pode começar a explorar e editar os arquivos dentro do diretório do projeto. O ponto de entrada principal é o arquivo `src/index.js`.