**Bem Vindo ao Curso de TypeScript pela Step lecionado pelo professor Fabio Luiz**

Neste curso aprenderemos fundamentos no TypeScript e algumas bibliotecas essenciais ao desenvolvimento Front end dinamico usando a linguagem TypeScript. Todos os códigos de exemplo usados nas aulas estara no repositorio: github.com/devfabioluiz/aulasStept na aplicacao meu-projeto-typescript, a qual sera a base de todas as aulas, sendo dividida por cada topico de cada aula em uma pasta separada com o nome Aula (numero da aula).

Bons estudos!

Recomendacoes de ambiente: instalar VSCode com plugins Codeium, VS Icons, Simple

React Snipets, ESLint e Prettier) ajudarao a melhorar sua perfomance.

**Antes da aula começar – Preparando ambiente:**

Introdução ao TypeScript

TypeScript é uma linguagem de programação desenvolvida pela Microsoft que é um superconjunto do JavaScript. Isso significa que todo código JavaScript válido também é válido em TypeScript, mas com a adição de tipos estáticos, que ajudam a evitar erros e melhorar a produtividade durante o desenvolvimento.

1. O que é TypeScript e por que usá-lo?

TypeScript adiciona tipagem estática ao JavaScript, o que significa que você pode definir os tipos de variáveis, parâmetros de funções e retornos. Isso facilita a detecção de erros antes mesmo de executar o código.

Benefícios do TypeScript:

- \*\*Detecção precoce de erros\*\*: Menos bugs em produção.

- \*\*Compatível com JavaScript\*\*: Permite a migração gradual de projetos.

2. Instalando o TypeScript

1. Node.js: Certifique-se de que o Node.js está instalado na sua máquina.

- Para verificar: `node -v`

2. npm: Instalado automaticamente com o Node.js.

- Para verificar: npm -v

3. Abra o terminal e execute o comando: npm install -g typescript

Para verificar se o TypeScript foi instalado corretamente: tsc -v

3. Configurando um Projeto com TypeScript

1. \*\*Crie uma pasta para o projeto:

mkdir meu-projeto-typescript

cd meu-projeto-typescript

2. Inicialize o projeto com npm:

npm init -y

3. Adicione o TypeScript como dependência do projeto:

npm install typescript --save-dev

4. Crie o arquivo de configuração do TypeScript:

tsc –init

Isso criará um arquivo `tsconfig.json`, onde você pode ajustar configurações como o diretório de saída (`outDir`) e o diretório de entrada (`rootDir`).

Exemplo básico de `tsconfig.json`:

{

"compilerOptions": {

"target": "ES6",

"module": "commonjs",

"strict": true,

"outDir": "./dist",

"rootDir": "./src"

}

}

5. \*\*Estrutura do Projeto\*\*:

Crie as pastas e arquivos:

meu-projeto-typescript/

├── src/

│ └── index.ts

├── dist/

├── package.json

└── tsconfig.json