



# ReactTs

📅 Created	@June 2, 2022 7:00 PM
👤 Created by	
🏷️ Tags	Front End
☰ Property	
☰ Property 1	

## AULA 1- EXERCICIO 1

Configurar a formatação auto do VS e prettier

- fazer exercícios para a aula

Exercícios:

Crie uma pasta contendo 3 arquivos .txt em seu computador. Após isso, abra a pasta com o VSCode e descubra o que fazem os comandos abaixo, após isso, responda os exercícios.

- Ctrl + p → Abre a pesquisa de arquivo por nome
- Ctrl + f → Abre a pesquisa por texto no arquivo
- Ctrl + d → Seleciona toda a palavra onde o cursor está, depois seleciona as próximas ocorrências desta palavra
- Alt + up → Move a linha atual para cima
- Alt + down → Move a linha atual para baixo
- Shift + alt + up → Copia a linha atual para cima
- Shift + alt + down → Copia a linha atual para baixo

## AULA 2- INTRODUÇÃO AO HTML E CSS

Classe: o que vai se repetir

ID: o que não se repete

`input[type="text"] {}` → style selecionando o type do input do html

## AULA 3 - INTRODUÇÃO AO HTML E CSS / REACT

React: Uma biblioteca JavaScript para criar interfaces de usuário - aplicações SPA (Single Page Applications)

Desvantagem: Não deve ser utilizado em aplicações que necessitam um bom SEO. → busca do google não lê JS e não destaca o site. → ideal para empresas/ aplicações que não precisam de indexação. (Usar React e ter um bom SEO? nesse caso, utilizar o Next.js → mantêm a indexação.)

Vantagem: Desempenho e reaproveitamento de código.

→ Iniciar o projeto:

- utilizar o vite → ferramenta de construção (várias tecnologias).
- yarn → `npm install —global yarn` → mesmo que npm, só que BEM mais rápido.
- `yarn create vite my-app —template react-ts` → dentro da pasta de projeto → my-app é o nome do projeto (utilizar letras minúsculas e sem acento, separar por -)
- Now run:  
`cd my-app`  
`yarn` → baixa os node modules  
`yarn dev` → (abre o localhost) → alterado para “start” no package.json
- Adicionar biblioteca → `yarn add`

## AULA 3 → FORM para REACTTS

**Componente** → letra maiúscula (ex: `function Teste(){}` ) → chamar com tag html (ex: `<Teste/>`)

**Função** → letra minúscula (ex: `function renderizarTitulo(){}` ) → chamar com chaves dentro de uma tag (ex: `<h1>{renderizarTeste()}</h1>` )

**Variável** → chamar dentro de tag com chaves (ex: <h1>{nomeVariavel}</h1>)

**Classe CSS** → className (Padrão camelCase)

class = className

onclick = onClick

function funcao(){ } → const funcao () ⇒ { }

function MeuComponente() {return <div/>}

const MeuComponente = () ⇒ {return <div/>}

AULA 4 → Estudar ciclo de vida da aplicação. SOLID e clean code.

- Principios do React :
  - **Componentes:** São como funções de JS, recebem parâmetros (props) e retornam um valor (JSX).
  - **Propriedades:** São como parâmetros passados para as tags HTML, porém são passados para os componentes como props.
  - **Estado (state, memória dos componentes):** onde guardamos os dados do nosso componente.
- Hooks: forma de utilizar o state e outros recursos do React sem precisar escrever classe.

### Hook useState()

- O que o useState faz? ele declara uma variável state (ciclo de vida)
- O que passamos para o useState como argumento? o valor inicial do state
- O que o useState retorna? um par de valores: [state, funcaoQueAtualizaOState]

### Hook useEffect()

- O que o useEffect faz? usando esse Hook, você diz ao React que o componente precisa fazer algo apenas depois da renderização.
- useEffect executa depois de toda renderização? Sim! Por padrão, ele roda depois da primeira renderização e depois de toda atualização
- (hook de efeito) → observar o state e, quando acontecer a minha condição, irá executar o que eu pedi.
- observa as renderizações. → pode ser especificado um valor também.

Codando:

Caso ocorra logo no início um erro ao executar o projeto, se você acabou de criá-lo ou cloná-lo, para instalar as dependências execute o comando:

`yarn` ou `npm install` na primeira vez e iniciar com `yarn start` ou `npm start`

- Styled Components: `yarn add styled-components @types/styled-components`

Lib Styled-components **yarn add styled-components** (estilização) - **yarn add @types/styled-components**