

A woman with long, dark hair and glasses is shown from the chest up, looking down at a laptop screen. The background is dark and slightly blurred.

IN

INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

AULA 15 - PROJETO

O QUE IREMOS APRENDER

01

PROJETO

IN

PROJETO

Projeto de gerenciamento de contatos

Desenvolva uma aplicação simples que permita aos usuários gerenciar uma lista de contatos. Eles poderão adicionar novos contatos, visualizar a lista existente, editar informações de contatos e excluir contatos desnecessários. Este é um projeto básico que pode ser personalizado e aprimorado com recursos adicionais para proporcionar uma experiência mais completa.

PROJETO

Requisitos Iniciais:

1. Adicionar Novos Contatos:

- Permita que os usuários adicionem novos contatos fornecendo informações como nome, número de telefone e endereço de e-mail.

2. Visualizar Lista de Contatos:

- Exiba uma lista de contatos, incluindo seus nomes e números de telefone, para que os usuários possam ter uma visão geral rápida.

PROJETO

3. Editar Informações de Contato:

- Implemente a capacidade de editar as informações de contato existentes, como nome, número de telefone ou endereço de e-mail.

4. Excluir Contatos:

- Permita que os usuários removam contatos que não são mais necessários.

SE LIGA NO CONTEÚDO DA PRÓXIMA AULA!

AULA 16 DE JAVASCRIPT.
ASYNC AWAIT



INFINITY SCHOOL
VISUAL ART CREATIVE CENTER

ASYNC AWAIT

Costumamos chamar de programação assíncrona o ato de executar uma tarefa em "segundo plano", sem nosso controle direto disso. Uma função assíncrona, em linguagens de programação como JavaScript, é uma função que é capaz de executar operações assíncronas de forma mais conveniente e legível usando a palavra-chave `async`. Isso permite que você escreva código que parece síncrono, mas que pode lidar com operações que levam tempo, sem bloquear a execução do programa.

JS Async/await

IN

ASYNC AWAIT

Exemplos da **Async Await** em JavaScript:

```
1  function addNumbers(numberA, numberB) {
2      return new Promise((resolve, reject) => {
3          let result = numberA + numberB;
4          if (result > 0) {
5              resolve(result);
6          } else {
7              reject(' O resultado é invalido')
8          }
9
10         });
11     }
12 console.log(addNumbers(2,5));
```



```
▼ Promise {<fulfilled>: 7} ⓘ
▶ [[Prototype]]: Promise
[[PromiseState]]: "fulfilled"
[[PromiseResult]]: 7
```



IN

INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

AULA 15 - PROJETO