**PROJETO LISTA DE TAREFAS**

**HTML**

No linck do javascriot, foi colocado o comando defer(adiar), os arquivos carrega depois do HTML para evitar erro nos comando na inicialização por trabalhar com o DOM(Document Object Model).

Usaremos a biblioteca: font-awesome

O que são Bibliotecas JavaScript?

A grosso modo, uma biblioteca JavaScript é um “pedaço de código” reutilizável que pode servir de base para a implementação de outros códigos.

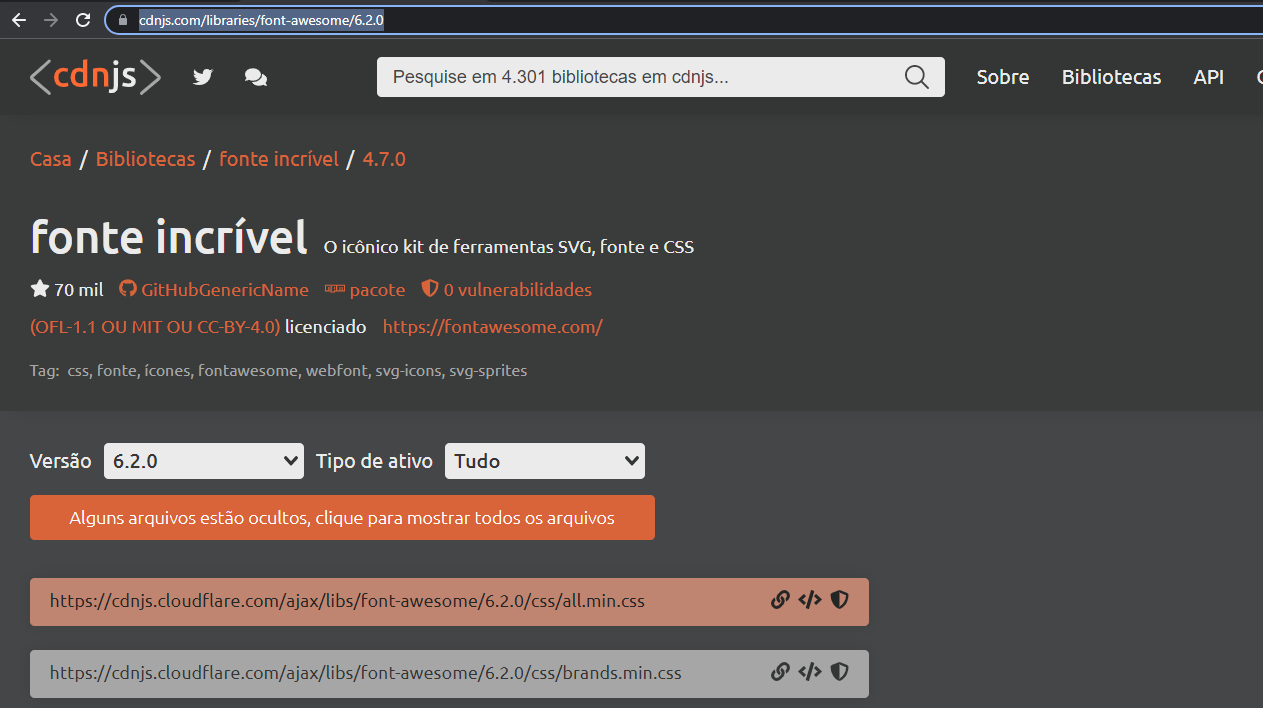
A ideia de utilização de uma Biblioteca é a de reutilizar códigos existentes para não perder tempo escrevendo códigos que outro programador já escreveu.

Em outras palavras, é um arquivo de JavaScript que contém um monte de funções, e essas funções realizam alguma tarefa útil para sua aplicação web.

**INICIANDO OPROJETO**

Para criar uma TAG com a pseudoclass: eu uso o abbreviation form.nome da class ou div#nome do id, ao clicar enter, o vscode vai entender que uma pseudoclass e criar altomaticamente com a tag que foi mencionada antes.

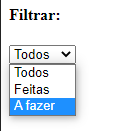
Vamos utilizar os ícones neste site: <https://cdnjs.com/libraries/font-awesome/6.2.0>: fazendo referência no arquivo html vscode, referência copiada do site:



Copia a referência e cola no arquivo HTML:

Ex: <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.2.0/css/all.min.css" integrity="sha512-xh6O/CkQoPOWDdYTDqeRdPCVd1SpvCA9XXcUnZS2FmJNp1coAFzvtCN9BmamE+4aHK8yyUHUSCcJHgXloTyT2A==" crossorigin="anonymous" referrerpolicy="no-referrer" />

**CRIANDO LISTA DE OPÇÃO**



Usamos as tag’s: <select> que vai criar a caixa de seleção, e <option> que vai ficar escondida dentro da caixa <select>, ao clicar vai abrir para escolher umas das <option>

<select id="filter-select">

<option value="all">Todos</option>

<option value="done">Feitas</option>

<option value="todo">A fazer</option>

</select>

**CSS**

**Medidas:**

A grande vantagem de se usar % é que não importa o tamanho que colocamos na <div>, a <img> sempre vai acompanhar o tamanho da sua tag mãe (a <div>).

**EM** e **REM** tem o mesmo conceito de % mas ao invés de serem baseadas no tamanho de um elemento, essas medidas são baseadas em tamanho de fonte. **EM** usa o tamanho da fonte do elemento pai e **REM** usa o tamanho da fonte do <html>.

**Comandos:**

**font-size: 1rem;** a letra(font), vai se adaptar ao tamanho da font HTML

**cursor: pointer;** mostra a mãozinha quando passar por cima dos botões.

**pointer-events: none;** comando que tira do ícone do botão o comando de click, porque o ícone está por cima do botão.

**Flex-direction: column;** define como os elementos flexível vão ficar, **column:** define que vão ficar em coluna: **um abaixo do outro.**

**Flex: 1;** deixa todo espaçamento em uma só largura.

**JavaScript**

Const todoform = docment.queryselector(“todo-form”);

Usamos o seletor de elementos (queryselector), para selecionar o elemento HTML pelo ID ou CLASS, que queremos usar no javascript, e vamos identificar esse elemento entre os (“todo-form”)

Criamos as variáveis abaixo:

const totoform = document.querySelector("#todo-form");

const todoInput = document.querySelector("#todo-input");

const totoList = document.querySelector("#todo-list");

const editForm = document.querySelector("#edit-form");

const editInput = document.getElementsByClassName("#edit-input");

const cancelEditeBtn = document.getElementById("#cancel-edit-btn");