

Uso de classes em arquivos separados

Uso de classes em arquivos separados

- Uma maneira de organizar melhor o código é separar cada classe em um arquivo externo separado

export class

- Para que uma classe seja importada em um outro arquivo, primeiro é necessário exportar a classe declarada utilizando a palavra reserva **export**

Arquivo: Fruta.js

```
1  export class Fruta {  
2      #nome;  
3      constructor(nome) {  
4          this.#nome = nome;  
5      }  
6  }
```

type module

- Antes de importar uma classe no arquivo que será chamado dentro do documento html, é necessário informar ao navegador que serão utilizados arquivos de scripts separados conhecidos como **módulos**, através da inclusão de:
 - `type="module"` dentro da tag `<script>`

Arquivo: index.html

```
1 <body>
2   <h2>Programação Orientada a Objetos - JavaScript</h2>
3   <script type="module" src="index.js"></script>
4 </body>
```

import class

- Para que uma classe seja importada em um outro arquivo, é necessário utilizar a palavra reserva **import** seguida do nome classe que foi exportada e o caminho

Arquivo: index.js



```
1 import { Fruta } from './Fruta.js';  
2  
3 const fruta = new Fruta('ameixa');  
4 console.log(fruta);
```

import, export, type="module"

- É provável que navegadores não executem o código JavaScript utilizando módulos (`type="module"`) se a página html for carregada diretamente de uma pasta local de um computador para o navegador, por exemplo: `file:///C:/Users/index.html`
- Por questão de segurança, o tipo módulo (`type="module"`) deve ser executado através de um servidor web, como a extensão live server do VS Code, por exemplo, ou qualquer outro servidor web que preferir.