

Roteiro Prático

Crie uma classe Estudante com os seguintes atributos: id, nome e credits. Os atributos id e nome devem ser definidos no construtor, o id não pode ser modificado após a criação. O atributo credito é privado e a classe deve ter um método de acesso que não permita valores igual ou menor que zero (0), caso isso ocorra deverá receber um valor padrão 1.

Ao final, adicione um método chamado "detalhar" na classe Estudante, que mostra todos os dados usando o console.log.

Teste sua classe criando dois estudantes, modificando seus créditos e exibindo os resultados usando o "detalhar()" de cada um.

Roteiro Prático

Crie três classes com os seguintes atributos

- Cidade: nome(string e somente leitura)
- Pessoa: nome(string), cidade(Cidade);
- Animal: nome(string), dono(Pessoa)

Em todas as classes os atributos devem ser declarados dentro dos construtores.

Depois de criar suas classes teste se o seguinte código funciona no mesmo programa:

```
const natal = new Cidade("Natal");  
const joao = new Pessoa("João", natal);  
const toto = new Animal("Totó", joao);
```

Roteiro Prático

Crie uma classe chamada Computador com os atributos `processador(string)` e `memoria(number)`.

Crie uma classe chamada Laptop, que herda de Computador e adiciona o atributo `bateria_watts(number)` com o valor padrão 0.

Crie uma classe chamada Desktop que herda de Computador e adiciona o atributo `cabinete(string)` com o valor padrão "".

OBS: A versão mais recente do TypeScript não permite a criação de atributos com tipos sem valor inicial ou que não esteja declarado em um construtor. Para criar, por exemplo, um atributo do tipo `string` com valor inicial (fora do construtor mesmo) faça o seguinte:

`public nome_do_atributo: tipo = "valor inicial"`. O mesmo serve para qualquer tipo, garantindo que o valor inicial é do tipo correto. Caso o atributo esteja declarado em um construtor ou seja setado dentro dele, não é necessário o declarar assim.

Roteiro de Laboratório

Neste laboratório iremos explorar todo potencial dos objetos em Javascript.

Para desenvolver as atividades deste roteiro, utilize o editor de código de sua preferência. Para executá-lo, utilize o console das Ferramentas do Desenvolvedor do seu navegador ou qualquer outro recurso para execução de código Javascript (node.js, por exemplo).

Atividade 01)

Escreva uma função chamada `excluirPropriedades` que dado um objeto e um array passados como argumentos, retorne esse objeto com todas as propriedades listadas no array removidas.

A assinatura do método e alguns casos de testes (assim como as expectativas de saída) estão abaixo:

```
const excluirPropriedades = (objeto, propriedades) => {  
  // Implemente aqui a sua lógica  
};  
  
var exemplo = { a: 1, b: 2, c: true};  
console.log(excluirPropriedades(exemplo, ['b'])); // { a: 1, c: true }  
  
exemplo = { a: 1, b: 2, c: true};  
console.log(excluirPropriedades(exemplo, ['b', 'a'])); // { c: true }
```

Atividade 02)

Escreva uma função que dado um objeto passado como argumento retorne um array formado por elementos do tipo array com dois elementos (chave, valor) do objeto original. Por exemplo, se for passado como argumento o objeto:

```
{ a: 1, b: 2 }
```

Deve ser retornado a array:

```
[['a', 1], ['b', 2]]
```

A assinatura do método e alguns casos de testes (assim como as expectativas de saída) estão abaixo:

```
const transformarObjEmArray = (objeto) => {  
  // Implemente aqui a sua lógica  
};  
  
var exemplo = { a: 1, b: 2, c: true};  
console.log(transformarObjEmArray(exemplo)); // [['a', 1], ['b', 2], ['c', 3]]
```

Roteiro de Laboratório

Neste laboratório iremos praticar como manipulamos o DOM para navegar, acessar e modificar elementos HTML em nossas páginas web.

Para desenvolver as atividades deste roteiro, utilize o editor de código de sua preferência. Para executá-lo, utilize o console das Ferramentas do Desenvolvedor do seu navegador ou qualquer outro recurso para execução de código Javascript (node.js, por exemplo).

Atividade 01)

Salve em um arquivo chamado *sobre.html* o conteúdo abaixo.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Sobre</title>
  </head>
  <body>
    <h1>Sobre</h1>
    <ul>
      <li>Nome: <span id="nome"></span></li>
      <li>Idade: <span id="idade"></span></li>
      <li>Cidade: <span id="cidade"></span></li>
    </ul>
  </body>
</html>
```

Em seguida, resolva os seguintes itens utilizando conceitos de manipulação de DOM com Javascript:

- Adicione uma tag **script** no final da tag **body**.
- Utilizando Javascript, mude o estilo CSS do **body** para que tenha o atributo **font-family** com valor *'Arial, sans-serif'*.
- Utilizando Javascript, substitua o valor no corpo de cada uma das tags **span** com sua própria informação.
- Utilizando Javascript, itere sobre todos os elementos **li** da página e inclua um estilo CSS que torne a fonte da cor vermelha.
- Utilizando Javascript, crie um novo elemento **img** e indique em seu atributo **src** uma imagem sua. Concatene/adicione esse novo elemento a página.

Roteiro de Laboratório

Neste laboratório iremos praticar a utilização de eventos em páginas web utilizando Javascript.

Para desenvolver as atividades deste roteiro, utilize o editor de código de sua preferência. Para executá-lo, utilize o console das Ferramentas do Desenvolvedor do seu navegador ou qualquer outro recurso para execução de código Javascript (node.js, por exemplo).

Atividade 01)

Salve em um arquivo chamado *calculadora.html* o conteúdo abaixo.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Calculadora</title>
  </head>
  <body>
    <label
      >Potência de 3 do número:
      <input type="number" id="potencia-input" size="2" />
    </label>
    <button id="potencia-button">Calculate</button>
    <br /><br />

    <label
      >Metade desse número:
      <input type="number" id="metade-input" size="2" />
    </label>
    <button id="metade-button">Calculate</button>
    <br /><br />

    <label
      >Calcular a área de um círculo de raio:
      <input type="number" id="area-input" size="2" />
    </label>
    <button id="area-button">Calculate</button>
    <br /><br />

    <div id="solucao"></div>
  </body>
</html>
```

Em seguida, resolva os seguintes itens utilizando conceitos de eventos com Javascript:

- a) Adicione uma tag **script** adicionando um arquivo chamado *script.js*, onde você irá incluir todo o seu código Javascript.
- b) Crie funções para cada uma das operações desejadas na nossa calculadora: uma para retornar a potência de 3 (ao cubo) de um número; outra para retornar a metade de um número; e uma para retornar a área de um círculo de raio N.
- c) Para cada operação, crie um escutador de evento para o botão, e quando ele for clicado, ache o valor passado na entrada (*input*) apropriada e mostre o resultado da **div** que tem como *id solucao*.
- d) Bônus: Responda também a eventos de teclado! Mapeie algumas teclas para isso!