**Javascript – Módulo 2 – B7Web**

**Material de estudo desenvolvido por:**

**Daniel Teixeira Quadros**

**Sumário**

[​ Aula 01 – O Que É DOM 2](#__RefHeading___Toc232_1188394332)

[​ Aulas 02 até 04 – Selecionando Elementos 2](#__RefHeading___Toc226_1188394332)

[​ Aula 05 – Manipulando Class do Elemento 1 2](#__RefHeading___Toc228_1188394332)

[​ Aula 06 – Manipulando Class do Elemento 2 3](#__RefHeading___Toc230_1188394332)

[​ Aula 07 – Tipos de Dados 3](#__RefHeading___Toc234_1188394332)

[​ Aula 08 – Comentários 3](#__RefHeading___Toc236_1188394332)

[​ Aula 09 – Criação e execução de Funções 4](#__RefHeading___Toc238_1188394332)

[​ Aula 10 – Parâmetros de Funções 4](#__RefHeading___Toc240_1188394332)

[​ Aula 11 – Manipulando Atributos 4](#__RefHeading___Toc242_1188394332)

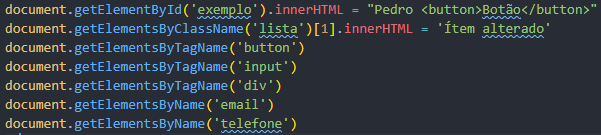
[​ Aula 12 – Dimensões 4](#__RefHeading___Toc244_1188394332)

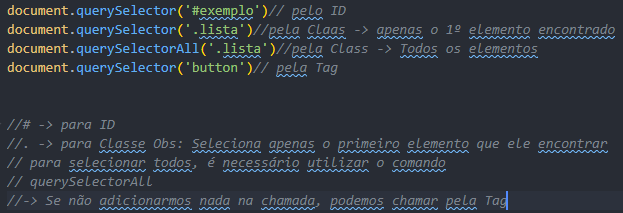
[​ Aula 13 – Distâncias e Scroll Suave 5](#__RefHeading___Toc246_1188394332)

[​ Aula 14 – Resposta do Exercício Scroll 7](#__RefHeading___Toc248_1188394332)

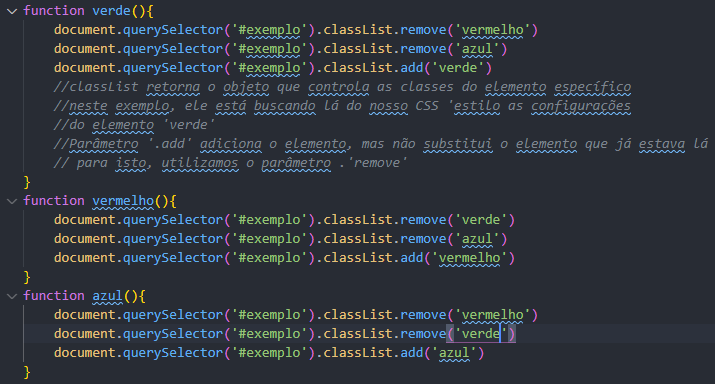
# Aula 01 – O Que É DOM

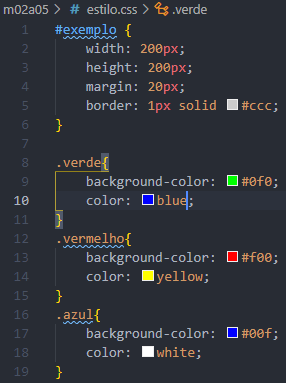
# **Aulas 02 até 04 – Selecionando Elementos**

****

****

# **Aula 05 – Manipulando Class do Elemento 1**

****

****

# **Aula 06 – Manipulando Class do Elemento 2**

****

# **Aula 07 – Tipos de Dados**

Tipos de variáveis

Algo interesante:

var tipoNome = typeof nome

var tipoIdade = typeof idade

Uma variável armazenando o tipo de uma outra variável.

# **Aula 08 – Comentários**

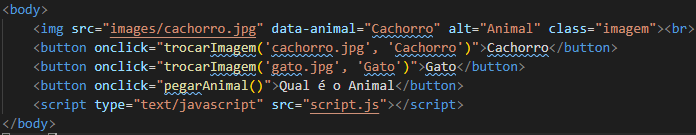
Comentários no código

# Aula 09 – Criação e execução de Funções

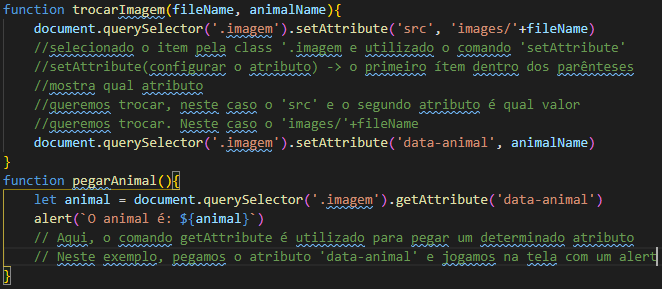
# **Aula 10 – Parâmetros de Funções**

# **Aula 11 – Manipulando Atributos**

**HTML:**

****

**JavaScript:**

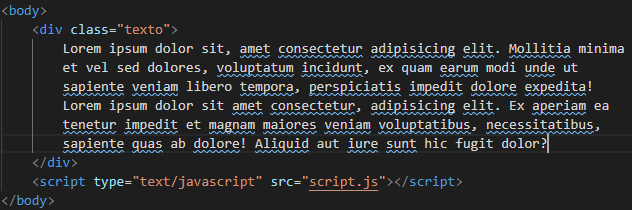


# **Aula 12 – Dimensões**

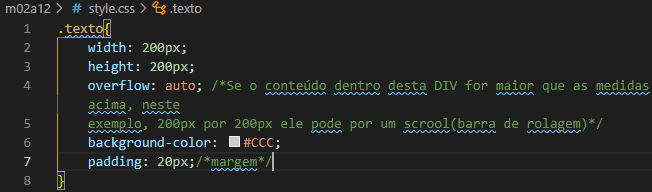
**Atributo – offset**

Veja a estrutura:

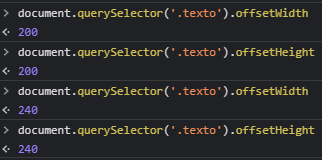
**HTML:**



**CSS:**



**Atributo offset no navegador:**



As duas primeiras linhas são o resultado apresentado antes de adicionar a linha ‘padding’ no CSS e as duas últimas linhas, o resultado depois de adicionado o ‘padding’.

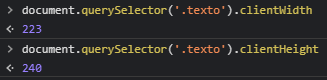
O Comando offset retorna as configurações do atributo relacionado. Neste caso class ‘.texto’

Obs.: O resultado engloba largura, scrollbar(dependendo do navegador), padding e borda

**Atributo client:**

**clientWidth** → pega a largura do conteúdo + o padding excluindo o scrollbar

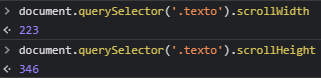
**clientHeigth** → pega a altura do conteúdo + o padding



**Atributo scroll: →** dá o tamanho real do conteúdo, ou seja, do texto

**scrollWidth →** largura total apenas do conteúdo, ou seja, neste exemplo, do texto

**scrollHeigth →** altura total apenas do conteúdo, ou seja, neste exemplo, do texto



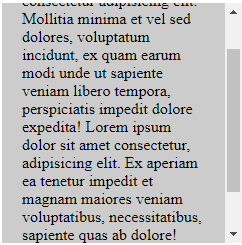
# **Aula 13 – Distâncias e Scroll Suave**

**Atributo scroll:**

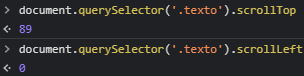
**scrollTop →** Mostra a posição que a scrollbar(vertical) está, a partir do topo.

**scrollLeft →** Mostra a posição que a scrollbar(horizontal) está, a partir da esquerda.

**Navegador:**

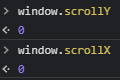
****

**Console:**

****

**Scroll para a tela inteira:**

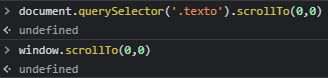
**window.scrollY** → Mostra a posição que a scrollbar(vertical) está, a partir do topo.

**window.scrollX →** Mostra a posição que a scrollbar(horizontal) está, a partir da esquerda.

**scrollTo()**

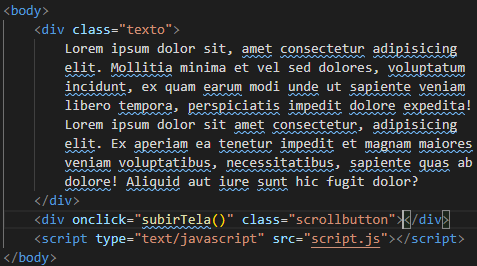
O comando scrollTo() leva o scroll para um ponto específico da tela.

ScrollTo(0, 0) → O primeiro parâmetro é a posição ‘X’(horizontal) e o segundo parâmetro é a posição ‘Y’(vertical).

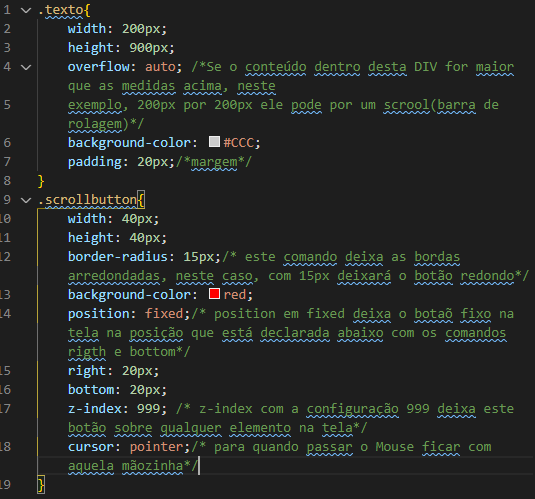


**CONFIGURANDO UM BOTÃO SCROLL NO FINAL DA PÁGINA PARA LEVAR A PÁGINA PARA O TOPO:**

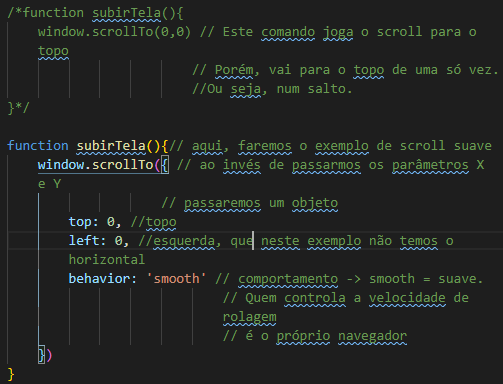
**HTML:**



**CSS:**

****

**JAVASCRIPT:**

****

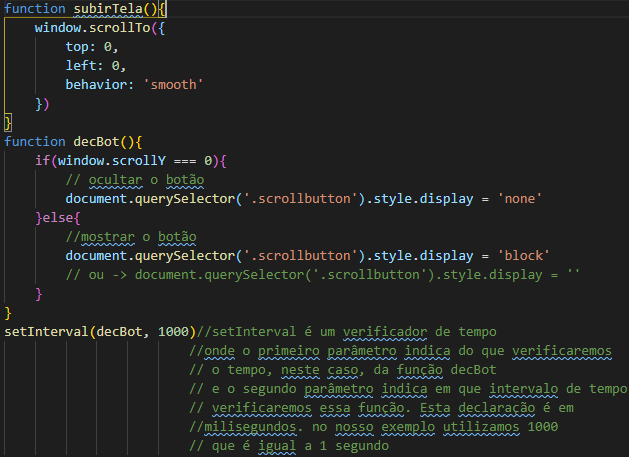
# **Aula 14 – Resposta do Exercício Scroll**

**SOLUÇÃO 01:**

**A SOLUÇÃO 1 NÃO É A MAIS EFETIVA, PORQUE ESTE MODELO FAZ COM QUE O SCRIPT FAÇA A VERIFICAÇÃO DA POSIÇÃO DO SCROLL A CADA SEGUNDO, MESMO QUE A TELA ESTEJA PARADA, GERANDO MUITO PROCESSAMENTO.**

Nesta solução, copiamos os arquivos da aula 13 e modificamos apenas o nosso JavaScript:

→ Método utilizado: **Temporizador**



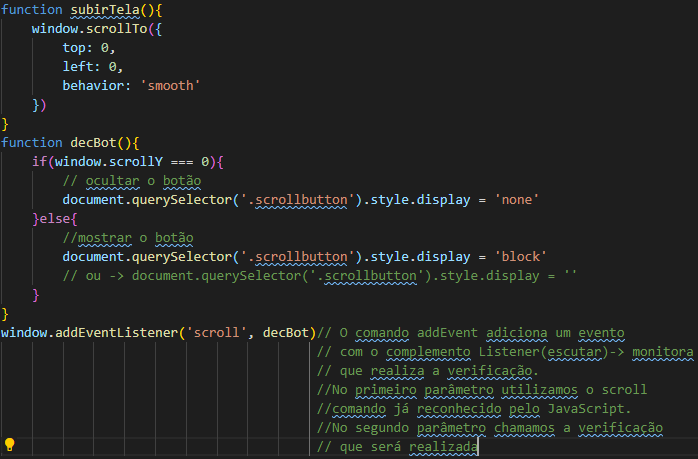
**SOLUÇÃO 02:**

**A SOLUÇÃO 2 É MAIS EFETIVA. NESTE MODELO, O SCRIPT SÓ IRA FAZER A VERIFICAÇÃO DA POSIÇÃO DA TELA SE HOUVER MOVIMENTO NA TELA. SE NÃO HOUVER MOVIMENTO O SCRIPT NÃO IRA CONSUMIR PROCESSAMENTO**

Nesta solução, copiamos os arquivos da aula 13 e modificamos o nosso JavaScript e o CSS:

→ Método utilizado: **Verificador de posição**.

**JAVASCRIPT:**



**CSS:** A única modificação realizada aqui, foi a adição do comando display no .scrollbutton

com a configuração ‘none’

