

Exercício 9 – Operadores

Temática

- Operadores
- Conversão de dados
- Manipulação do DOM

Para os seguintes problemas e tendo como base os ficheiros fornecidos deverá implementar as respetivas soluções em javascript.

Todo o código *javascript* deverá ser colocado no **ficheiro js externo**, ou seja, não deve proceder a qualquer alteração nos ficheiros HTML e de CSS.

Versão resolvida: <http://labmm.clients.ua.pt/LM3/LM3-p/ex09>

1. Operadores (parte1.js)

- 1.1. Crie a variável `num` e atribua-lhe o valor 40. Analise na consola o output das seguintes declarações e as suas diferenças.

```
num +=1 5;
num -= 5;
num *= 5;
num /= 5;
```

2. Concatenação (parte2.js)

- 2.1. Declare as variáveis `nome_primeiro` e `nome_apelido` atribua às variáveis os respetivos valores, “Zé” e “Manel”. Declare a variável `nome_completo` e atribua-lhe como valor a concatenação² das variáveis `nome_apelido` e `nome_primeiro`, separado por uma vírgula e mostre na consola o valor da variável `nome_completo`.
- 2.2. Elabore um algoritmo em que seja pedido ao utilizador o dia, mês e ano e mostre, por exemplo, a seguinte mensagem “15 de Janeiro de 2035”.
- 2.3. Declare as variáveis `num1`, `num2`, `num3` com os valores 10, 20 30, respetivamente. Declare a variável `resultado`. Apresente na consola o output da variável `resultado` para as seguintes operações:
 - Concatenação do texto 'Divisão: ' com o resultado da divisão de `num2` com `num1`;
 - Concatenação do texto 'Módulo: ' com o resto da divisão entre `num2` e `num1`;
 - Diferença entre a multiplicação de `num1` e `num2` e a divisão de `num3` e `num1`;
 - Concatenação de `num1` e `num2`;

3. Conversão de dados (parte3.js)

¹ Pode saber mais sobre atribuição em: https://www.w3schools.com/js/js_assignment.asp

² Pode ler mais sobre concatenação em: https://www.w3schools.com/js/js_operators.asp e em: <http://2ality.com/2011/10/string-concatenation.html>

- 3.1. Crie uma nova variável `texto1` com o valor “1.25px” e adicione ao evento **onclick** do botão com o id `converter` as seguintes ações:
 - 3.1.1. Mostrar na consola o resultado da conversão de `texto1` para um número inteiro;
 - 3.1.2. Mostrar na consola o resultado da conversão de `texto1` para um número real;
- 3.2. Crie duas variáveis `num1` e `num2` com os valores 1 e 2 respetivamente e adicione ao evento **onclick** do ponto anterior as seguintes ações:
 - 3.2.1. Mostrar na consola o resultado da soma de `num1` e `num2`;
 - 3.2.2. Converter `num1` para String e mostrar de seguida na consola o resultado da soma de ambas as variáveis;
 - 3.2.3. Converter `num1` e `num2` para String e mostrar de seguida na consola o resultado da soma de ambas as variáveis;
4. Manipulação do DOM (parte4.js)
 - 4.1. Declare uma variável `texto_inicial` e guarde nessa variável o conteúdo dentro da div “content”;
 - 4.2. Adicione um evento associado ao botão “adicionar” de modo que, ao clicar, seja adicionado à div “content” num novo parágrafo. Para tal, utilize a variável `texto_inicial`.
 - 4.3. Declare uma variável `tamanho` com o valor de 150. Se seguida, associe um evento onclick ao botão “reduzir”, dentro do qual, desenvolva os seguintes pontos:
 - 4.3.1. Modifique a largura³ da imagem “alvo” segundo o valor da variável `tamanho`;
 - 4.3.2. Modifique o posicionamento da imagem “alvo” para “absolute”⁴;
 - 4.3.3. Defina o posicionamento da imagem “alvo” relativamente ao topo em -150 pixéis;
 - 4.4. Utilizando javascript, adicione na div “botoes” um botão com id “posicionar” e em seguida:
 - 4.4.1. Associe um evento onclick ao botão que ao clicar posicione a imagem “alvo” à esquerda⁵ na posição -50 pixéis;
 - 4.4.2. Modifique o ponto anterior de modo a que a imagem “alvo” assuma o tamanho inserido no campo de input “tamanho”;
 - 4.4.3. Analise a razão pela qual:

³ https://www.w3schools.com/cssref/pr_dim_width.asp

⁴ https://www.w3schools.com/cssref/pr_class_position.asp

⁵ https://www.w3schools.com/cssref/pr_pos_left.asp

4.4.3.1.o código para adicionar o botão tem de ser colocado no início do script;

4.4.3.2.o estilo aplicado pelo botão “posicionar” só funciona após clicar no botão “reduzir”;

4.5. Utilizando javascript, adicione na div “botoes” um campo de input com id “posicao” e em seguida:

4.5.1.Modifique o evento onclick do botão posicionar de modo que seja aplicada na imagem “alvo” a posição inserida no campo de input “posicao”;

4.5.2.Comente o ponto anterior e adicione uma nova instrução de modo a que a posição atribuída seja sempre o dobro da posição inserida no campo de input;

4.5.3.Comente o ponto anterior e adicione uma nova instrução de modo a que a posição atribuída seja sempre a soma da posição inserida no campo de input e 150 pixéis;

5. Operadores (parte1.js)

Abra o ficheiro parte1.js implemente as seguintes alíneas.

5.1. Analise na consola o output das seguintes declarações:

```
var var1 = 10;
console.log(var1 > 5) -> True
var var2 = 5;
console.log(var1 > var2) -> True
var1 = "a";
var2 = "b";
console.log(var1 > var2) -> False
```

5.2. Analise na consola o output das seguintes declarações e as suas diferenças:

```
n1 = 10;
a = n1++6;
console.log(a) -> 10
console.log(n1) -> 11
n1 = 10;
a = ++n1;
console.log(a) -> 11
console.log(n1) -> 11
```

6. Carregamento do javascript (parte4.js)

6.1. Modifique no ficheiro HTML o local do carregamento do ficheiro parte4.js, de forma a que seja carregado na secção <head> da página, analise o resultado;

6.2. Adicione agora no ficheiro javascript um evento de modo a que todo o script apenas seja executado quando a página estiver totalmente carregada.

⁶ Pode saber mais sobre incremento em: https://www.w3schools.com/js/js_arithmetic.asp