

Javascript Revisões – Parte 1

Instruções

- Resolve os exercícios escrevendo código JavaScript
- Usa `prompt()`, `window.alert()`, `console.log()` ou `document.write()` quando necessário

1. Estrutura switch

1.1. Cria um programa que peça ao utilizador um número de 1 a 3 e mostre:

- "Opção A" se o utilizador introduziu 1
- "Opção B" se o utilizador introduziu 2
- "Opção C" se o utilizador introduziu 3
- "Opção inválida" para outro valor

1.2. Cria um programa que peça um mês (número de 1 a 12) e mostre o nome do mês (janeiro, fevereiro, etc.) .

1.3. Cria um programa que peça ao utilizador uma letra e mostre:

- "a" → "Vogal"
- "e" → "Vogal"
- "i" → "Vogal"
- "o" → "Vogal"
- "u" → "Vogal"
- outro valor → "Consoante ou inválido"

2. Ciclo for

2.1. Mostra os números de 15 a 1 (ordem decrescente), utilizando um ciclo for.

2.2. Mostra os números pares entre 24 e 68, utilizando um ciclo for com incrementos de 2.

2.3. Pede ao utilizador um número entre 1 e 10 e mostra a tabuada desse número, utilizando um ciclo for.

2.4. Calcula e mostra a soma dos números de 10 a 20, utilizando um ciclo for.

3. Ciclo while

3.1. Mostra os números de 1 a 5 usando while.

3.2. Cria um programa que peça um número ao utilizador e repita o pedido enquanto o número for menor que 20, usando while.

3.3. Cria um programa que peça uma palavra-passe e continue a pedir até que seja "EPA", usando while.

3.4. Mostra os números pares entre 10 e 20 usando while.

3.5. Cria um contador que comece em 1 e vá até ao número indicado pelo utilizador.

Projeto integrador para revisões

Objetivos

Rever:

- declaração de variáveis com let e const;
- tipos de dados;
- operadores aritméticos e relacionais;
- template literals;
- estruturas condicionais;
- ciclos;
- apresentação de resultados na consola.

1. Criar o projeto

Cria uma pasta com o nome:
gestor-inventario

Dentro da pasta cria os ficheiros:

- index.html
- script.js

No ficheiro index.html, cria a estrutura básica de uma página HTML e associa o ficheiro script.js.

Nesta fase, todos os resultados deverão ser apresentados através da **consola do navegador (console.log)**.

2. Registar um produto

Considera que existe no inventário um monitor com os seguintes dados:

- Produto: Monitor
- Preço unitário: 149,90 €
- Quantidade: 8

Cria no ficheiro script.js três variáveis para armazenar estas informações.

Apresenta na consola:

Produto: Monitor

Preço: 149.90 €

Quantidade: 8

3. Calcular o valor existente em stock

Calcula o valor total correspondente às unidades existentes:

valor em stock = preço × quantidade

Guarda o resultado numa variável e apresenta uma mensagem semelhante a:

Valor existente em stock: 1199.20 €

Utiliza uma **template literal** para construir a mensagem.

4. Verificar o nível de stock

A escola considera que um produto tem **stock baixo quando existem menos de 5 unidades**.

Utilizando uma estrutura if...else, apresenta uma das seguintes mensagens:

Estado: Stock baixo

ou

Estado: Stock suficiente

Altera temporariamente a quantidade do produto e verifica se o programa apresenta corretamente as duas situações.

5. Determinar a necessidade de encomenda

Pretende-se manter pelo menos **10 unidades** de cada produto disponíveis.

Se existirem menos de 10 unidades, calcula quantas unidades devem ser encomendadas.

Exemplo:

Se existirem 8 unidades:

É necessário encomendar 2 unidades.

Se existirem 10 ou mais:

Não é necessário efetuar encomenda.

6. Simular a entrada de equipamento

Foi efetuada uma encomenda de **5 novas unidades**.

Atualiza a quantidade existente e apresenta:

Quantidade anterior: 8

Unidades recebidas: 5

Nova quantidade: 13

Calcula novamente o valor total do produto existente em stock.

7. Utilizar um ciclo

Pretende-se colocar uma etiqueta numérica em cada unidade existente.

Utiliza um ciclo for para apresentar:

Unidade 1

Unidade 2

Unidade 3

...

O número de linhas apresentado deverá depender do valor da variável que contém a quantidade.

8. Códigos

A escola pretende atribuir um código a cada unidade do monitor.

Os códigos deverão seguir o formato:

MON-001

MON-002

MON-003

...

Cria um ciclo que apresente automaticamente todos os códigos correspondentes às unidades existentes.