

**ESMAD | TSIW | POO**  
**Ficha de Exercícios nº4 (Extra) – Objetos & Arrays**

Abra o Visual Studio Code e resolva os seguintes exercícios::

**1. Criação, acesso, modificação, contagem e iteração (ex1.html)**

- a. Crie um array **countries** com o nome dos países Portugal, Espanha e França
- b. Imprima numa caixa de alerta o primeiro elemento
- c. Imprima numa caixa de alerta o último elemento
- d. Adicione um novo elemento: Alemaha
- e. Altere o elemento Alemaha para Alemanha
- f. Imprima numa caixa de alerta a seguinte mensagem:  

**O array tem X países!**

(onde X indica o nº de elementos atuais no array)
- g. Crie outro array chamado **countries2**
- h. Iguale o array criado ao array **countries**
- i. Adicione um novo elemento ("Dinamarca") ao array **countries2**
- j. Imprima numa caixa de alerta o array **countries**. Analise o resultados.
- k. Itere (usando um ciclo **for**) sobre todos os elementos do array apresentando os nomes dos países na consola da seguinte forma:  

**1 - Portugal**

**2 - Espanha**

...
- l. Resolva o exercício anterior usando o ciclo **for..of**

**2. Métodos de arrays - adicionar/remover elementos (ex2.html)**

- a. Crie um array **names** vazio
- b. Inclua na página Web uma caixa de texto e um botão "ADICIONAR NOME". Ao pressionar o botão, deve adicionar no array o conteúdo da caixa de texto e imprimir na consola o conteúdo do array
- c. Adicione 5 nomes ("Rui", "Pedro", "Rita", "Joana", "Manuel")
- d. Extraia o último nome do array e exiba-o numa caixa de alerta
- e. Extraia o primeiro nome do array e exiba-o numa caixa de alerta
- f. Crie um segundo array chamado **names2** com os elementos: "Ricardo", "Maria"
- g. Junte ambos os arrays num novo array chamado **allNames** e mostre-o na consola
- h. Resolva a questão anterior usando sintaxe de propagação (spread)

- i. Melhore a alínea b) de forma a não inserir nomes repetidos

### 3. Métodos de arrays - pesquisar elementos (ex3.html)

- a. Crie array numérico chamado **grades** com as notas de alunos a POO (4,12,16,11,8)
- b. Imprima numa caixa de alerta a primeira nota positiva do array
- c. Crie uma função **getPositiveGrades** que deve retornar um array com todas as notas positivas. Em seguida, imprima uma caixa de alerta com o conteúdo desse array
- d. Crie uma função **checkDisapprovals** que deve retornar um valor lógico indicando se existem ou não notas negativas. Imprima o resultado na consola.
- e. Crie uma função **validateGrades** que deve retornar um valor lógico indicando se as notas do array são válidas. Entendem-se como notas válidas todas as notas que estejam entre 0 e 20 inclusive. Imprima o resultado na consola

### 4. Métodos de arrays - transformar array (ex4.html)

- a. Crie array numérico chamado **ages** com a idade de todos os membros de uma família (1, 9, 41, 75, 44, 73)
- b. Crie um novo array chamado **fiveYearsFromNowAges** com as idades da família daqui a 5 anos. Imprima o array numa caixa de alerta
- c. Imprima na consola a soma de todas as idades
- d. Imprima na consola a média de todas as idades
- e. Crie um novo array chamado **adults** com as idades inferiores a 18 transformadas em "-" e as restantes inalteradas
- f. Ordene o array das idades e apresente o resultado numa caixa de alerta
- g. Inverta o resultado da alínea anterior e guarde num novo array chamado **inverseOrderedAges**

### 5. Arrays & Objetos (ex5.html)

- a. Crie o próximo array:

```
const games = [
  { name: 'Rick Dangerous', year: 1989 },
  { name: 'Goblins', year: 1992 },
  { name: 'Golden Axe', year: 1989 }
]
```

- b. Exiba na consola informação sobre os jogos:

**Nome: N / Ano: A**

...

(onde N é o nome do jogo e A é o ano em que o jogo foi lançado)

- c. Exiba na consola o nº de jogos existentes
- d. Crie função chamada **addGame** que crie um novo objeto (com dados do utilizador provenientes de 2 caixas de texto e um botão “ADICIONAR JOGO”) e insira os mesmos no array
- e. Melhore a alínea anterior de forma a não deixar inserir um jogo já existente. Caso já exista deve ser exibida uma caixa de alerta com o texto: “JOGO JÁ EXISTENTE!”
- f. Crie uma função chamada **averageLaunchDate** que calcule e devolva a média do ano de lançamento dos jogos. Apresente o resultado numa caixa de alerta
- g. Crie um novo botão chamado “REMOVER JOGO” que ao ser premido deve remover o jogo cujo nome foi inserido na caixa de texto. Se o jogo não existir deve ser apresentada a numa caixa de alerta a mensagem: “JOGO INEXISTENTE”
- h. Crie uma função chamada **gamesFrom90** que retorne os nomes dos jogos lançados na década de 90. Apresente o resultado numa caixa de alerta
- i. Crie uma função chamada **concatenateGameNames** que devolva uma string com todos os nomes dos jogos em minúsculas e sem espaços. Apresente o resultado numa caixa de alerta