

Os exercícios estão separados por tópicos na seção a seguir, as resoluções estarão na seção seguinte. **Atenção:** Tente fazer o exercício antes de conferir sua resolução, e use ela para conferir o que você estava errando.

## 1 Exercícios

### 1.1 Variáveis e operadores

1. Crie uma variável que armazene uma temperatura em Fahrenheit, e realize a conversão para Celsius. A equação para realizar a conversão é:

$$C = \frac{F - 32}{1.8}$$

2. Crie duas variáveis **nome** e **sobrenome**. Crie uma nova variável chamada **nomeCompleto** e realize a concatenação de nome e sobrenome com template strings.
3. Crie cinco variáveis que representam cinco notas de um estudante. Calcule a média ponderada, sabendo que o peso das notas são respectivamente: 3, 2, 1, 4 e 5.
4. Crie duas variáveis que representam as dimensões (em km) de um terreno: **comprimento** e **largura**. Calcule a área deste terreno, em seguida converta para hectares. **Nota:** 1 km<sup>2</sup> equivale a 100 hectares.
5. Crie uma variável que armazena uma temperatura em Kelvin. Converta para Celsius. A equação é:

$$C = K - 273$$

### 1.2 Estruturas condicionais

1. Crie três variáveis: a, b e c. Estas variáveis armazenam números. Exiba qual destes números é o maior.
2. Crie três variáveis que representam os três lados de um triângulo. Aplique a seguinte regra para definir se o triângulo é possível: Se a soma de dois lados é maior que o terceiro lado.
3. Crie uma variável idade. Caso seja uma idade inválida (negativa ou maior que 130) mostre uma mensagem de aviso.
4. Crie duas variáveis: salário e quantidadeDependentes. Siga a seguinte regra para aplicar ajustes salariais: Caso não possua dependentes não realize ajustes no salário, se houver entre 1 e 5 aumentar em 30% o salário. Caso seja acima de 5 aumentar em 40% o salário.

5. Crie uma variável para armazenar um dia da semana (1 - Domingo, 2 - Segunda, ... 7 - Sábado). Com base no valor indique se é final de semana ou dia de útil.
6. Crie uma variável ano e verifique se este ano é bissexto. Um ano é bissexto quando: **Caso 1)** É um número divisível por 4, mas não é divisível por 100. **Caso 2)** É um número divisível por 4, por 100 e por 400.

### 1.3 Estruturas de seleção

1. Defina uma variável para armazenar um caractere e determine se é uma vogal ou uma consoante. Use um switch-case para implementar a lógica.
2. Defina uma variável para armazenar um mês do ano (em número) e imprima o nome do mês correspondente. Use um switch-case para implementar a lógica.
3. Defina uma variável para armazenar um número de 1 a 12 e imprima o número de dias correspondente ao mês. Use um switch-case para implementar a lógica. Considere que fevereiro tem 28 dias em anos não bissextos.
4. Defina uma variável para armazenar um número de 1 a 4 e imprima o nome da estação do ano correspondente. Use um switch-case para implementar a lógica.
5. Defina uma variável para armazenar uma nota (de 0 a 10) e imprima a menção correspondente. Use um switch-case para implementar a lógica. Considere as seguintes faixas de notas: de 0 a 4 - Insuficiente, de 5 a 6 - Regular, de 7 a 8 - Bom, de 9 a 10 - Excelente.

### 1.4 Estruturas de repetição

1. Use um laço while para imprimir na tela todos os números pares de 2 a 10.
2. Use um laço for para imprimir na tela a tabuada do 7 (de 1 a 10).
3. Use um laço for para imprimir na tela a sequência de Fibonacci até o décimo termo.
4. Mostre a seguinte figura com utilização de laços:

```
*  
**  
***  
****  
*****
```

5. Escreva um programa que use um laço for para imprimir na tela os números de 10 a 1 (em ordem decrescente).
6. Mostre apenas os números ímpares de 1 até 1000.

## 1.5 Funções

1. Escreva uma função que receba um número e retorne seu dobro.
2. Escreva uma função que receba um número e retorne seu quadrado.
3. Escreva uma função que receba nome, idade, endereço e retorne uma mensagem no formato: "Olá, eu sou X, tenho X ano(s) e moro em X". Faça uso da interpolação de strings.
4. Escreva uma função que receba dia, mês e ano e retorne a data por extenso no formato "DD/MM/AAAA".
5. Escreva uma função que receba um preço de um produto e um percentual. Retorne o valor aplicado ao desconto.
6. Escreva uma função que receba uma palavra e um número. Retorne a palavra repetida a quantidade de vezes indicada pelo segundo parâmetro. Exemplo: repetir("batata", 3) -> "batatabatatabatata". OBS: Utilize um loop para resolver.
7. Escreva uma função que recebe outra função e invoca ela.
8. Transforme os itens 1, 2 e 3 em arrow function expressions.

## 1.6 Strings

1. Escreva uma função que receba uma string e retorne o número de caracteres contidos nela.
2. Escreva uma função que receba uma string e retorne a mesma string em letras maiúsculas.
3. Escreva uma função que receba uma string e retorne a mesma string em letras minúsculas.
4. Escreva uma função que receba duas strings e retorne true se elas forem iguais ou false caso contrário.
5. Escreva uma função que receba uma string e um número e retorne os últimos caracteres da string de acordo com o número informado.
6. Escreva uma função que recebe uma frase e uma palavra. Retorne a frase com as ocorrências da palavra censuradas/trocadas por \*\*\*. Exemplo: censura("Eu jogo lol e valorant, mas lol é melhor", "lol") -> "Eu jogo \*\*\* e valorant, mas \*\*\* é melhor". Dica: use a função replaceAll.

7. Escreva uma função que receba uma string e um caractere e retorne o número de ocorrências desse caractere na string.

## 1.7 Arrays

1. Crie um array contendo os números de 1 a 10 e exiba-os no console com for-of e for comum.
2. Crie uma função que recebe um array de strings como parâmetro e retorne um novo array apenas com as strings acima de 5 caracteres.
3. Crie uma função que recebe um array de números e retorna um novo array com apenas os números que são maiores que a média do array do parâmetro.
4. Crie um array de booleans e realize um OU de todas os valores.
5. Crie uma função que recebe um array com os nomes dos alunos de uma turma, e outro array que recebe a lista de presença com os nomes. Indique quais alunos faltaram.
6. Crie uma função que recebe um array de números e retorna um array com todos os valores elevados ao quadrado.
7. Crie uma função que recebe um array de nomes de arquivos, e recebe também uma extensão, a função deve retornar apenas os nomes de arquivo que forem dessa extensão.
8. Crie uma função que recebe um número, em seguida retorne um array de números aleatórios com o tamanho especificado pelo número.
9. Refaça os itens 1, 2, 6, e 7. Utilize as funções map, filter e forEach.

## 2 Resoluções

### 2.1 Variáveis e operadores

1. 

```
let fTemp = 108.0;
let cTemp = (fTemp - 32) / 1.8;
console.log(`${fTemp}°F => ${cTemp.toFixed(2)}°C`);
```
2. 

```
let nome = "Pedro";
let sobrenome = "Silveira";
let nomeCompleto = nome + " " + sobrenome;

console.log(`Olá, meu nome é ${nomeCompleto}`);
```

```
3. let a = 7.5,
    b = 6.6,
    c = 4.2,
    d = 9.5,
    e = 8.8;

let mediaPonderada = (3 * a + 2 * b + 1 * c + 4 * d + 5 * e) / 15;

console.log(`A média ponderada é: ${mediaPonderada}`);

4. const comprimento = 20;
const largura = 40;
const areaEmKm2 = comprimento * largura;
const areaEmHa = areaEmKm2 * 100;

console.log(`${areaEmKm2} km² => ${areaEmHa} ha`);

5. let kTemp = 350;
let cTemp = kTemp - 273;

console.log(`${kTemp}K => ${cTemp}°C`)
```

## 2.2 Estruturas condicionais

```
1. let a = 7,
    b = 7,
    c = 5;

if (a >= b && a >= c) {
  console.log(`${a} é o maior número`);
} else if (b >= a && b >= c) {
  console.log(`${b} é o maior número`);
} else if (c >= a && c >= b) {
  console.log(`${c} é o maior número`);
}

2. let lado1 = 20,
    lado2 = 15,
    lado3 = 5;

if (lado1 + lado2 > lado3 && lado1 + lado3 > lado2 && lado2 + lado3 > lado1) {
  console.log("O triângulo é possível");
} else {
```

```
        console.log("O triângulo não é possível");
    }

3. let idade = 200;

    if(idade < 0 || idade > 130) {
        console.log("Idade inválida: " + idade);
    } else {
        console.log("Idade válida");
    }

4. let salario = 3000;
    let quantidadeDependentes = 3;

    if (quantidadeDependentes === 0) {
        console.log("Você não possui dependentes.");
    } else if (quantidadeDependentes >= 1 && quantidadeDependentes <= 5) {
        salario += salario * 0.3;
    } else {
        salario += salario * 0.4;
    }
    console.log(`O salário é: ${salario.toFixed(2)}`);

5. let diaSemana = 7;

    if (diaSemana >= 2 && diaSemana <= 6) {
        console.log("Dia útil");
    } else {
        console.log("Final de semana");
    }

6. let ano = 2024;

    if (
        (ano % 4 == 0 && ano % 100 != 0) ||
        (ano % 4 == 0 && ano % 100 == 0 && ano % 400 == 0)
    ) {
        console.log(ano + " é um ano bissexto");
    } else {
        console.log(ano + " não é um ano bissexto");
    }
}
```

## 2.3 Estruturas de seleção

```
1. const letra = 'f';

switch (letra.toLowerCase()) {
  case 'a':
  case 'e':
  case 'i':
  case 'o':
  case 'u':
    console.log('A letra é uma vogal.');
```

break;

```
  default:
    console.log('A letra é uma consoante.');
```

}

2.

```
const mes = 7;

switch (mes) {
  case 1:
    console.log('Janeiro');
```

break;

```
  case 2:
    console.log('Fevereiro');
```

break;

```
  case 3:
    console.log('Março');
```

break;

```
  case 4:
    console.log('Abril');
```

break;

```
  case 5:
    console.log('Maio');
```

break;

```
  case 6:
    console.log('Junho');
```

break;

```
  case 7:
    console.log('Julho');
```

break;

```
  case 8:
    console.log('Agosto');
```

break;

```
  case 9:
```

```
        console.log('Setembro');
        break;
    case 10:
        console.log('Outubro');
        break;
    case 11:
        console.log('Novembro');
        break;
    case 12:
        console.log('Dezembro');
        break;
    default:
        console.log('Mês inválido');
}
```

3. const mes = 2;

```
switch (mes) {
    case 1:
    case 3:
    case 5:
    case 7:
    case 8:
    case 10:
    case 12:
        console.log('31 dias');
        break;
    case 4:
    case 6:
    case 9:
    case 11:
        console.log('30 dias');
        break;
    case 2:
        console.log('28 dias em anos não bissextos');
        break;
    default:
        console.log('Mês inválido');
}
```

4. const numero = 3;

```
switch (numero) {
    case 1:
        console.log('Verão');
```



```
        break;
    case 2:
        console.log('Outono');
        break;
    case 3:
        console.log('Inverno');
        break;
    case 4:
        console.log('Primavera');
        break;
    default:
        console.log('Número inválido');
}
```

5. `const nota = 7.5;`

```
switch (Math.floor(nota)) { // arredonda para 7
    case 0:
    case 1:
    case 2:
    case 3:
    case 4:
        console.log('Insuficiente');
        break;
    case 5:
    case 6:
        console.log('Regular');
        break;
    case 7:
    case 8:
        console.log('Bom');
        break;
    case 9:
    case 10:
        console.log('Excelente');
        break;
    default:
        console.log('Nota inválida');
}
```

## 2.4 Estruturas de repetição

1. `let numero = 2;`

```
while (numero <= 10) {
  console.log(numero);
  numero += 2;
}
```

2. for (let i = 1; i <= 10; i++) {  
 console.log(`7 x \${i} = \${7 \* i}`);  
}

3. let primeiroTermo = 0;  
 let segundoTermo = 1;

```
for (let i = 1; i <= 10; i++) {
  const proximoTermo = primeiroTermo + segundoTermo;
  primeiroTermo = segundoTermo;
  segundoTermo = proximoTermo;
  console.log(primeiroTermo);
}
```

4. let text = "";

```
for (let i = 0; i < 5; i++) {
  for (let j = 0; j <= i; j++) {
    text += "*";
  }
  text += "\n"
}
```

```
console.log(text);
```

5. for (let i = 10; i >= 1; i--) {  
 console.log(i);  
}

6. for (let i = 1; i <= 1000; i++) {  
 if (i % 2 === 1) {  
 console.log(i);  
 }  
}

## 2.5 Funções

```
1. function dobrarNumero(numero) {  
    return numero * 2;  
}  
console.log(dobrarNumero(2)); // 4  
  
2. function calcularQuadrado(numero) {  
    return numero * numero;  
}  
console.log(calcularQuadrado(3)); // 9  
  
3. function apresentacao(nome, idade, endereco) {  
    return `Olá, eu sou ${nome}, tenho ${idade} ano(s) e moro em ${endereco}.`;   
}  
console.log(apresentacao("José", 17, "Fortaleza, Ceará"));  
  
4. function dataPorExtenso(dia, mes, ano) {  
    // Esta expressão é um operador ternário:  
    // podemos usar ele para atribuir um valor baseado numa condição!  
    const diaFormatado = dia < 10 ? "0" + dia : dia;  
    const mesFormatado = mes < 10 ? "0" + mes : mes;  
  
    return `${diaFormatado}/${mesFormatado}/${ano}`;  
}  
console.log(dataPorExtenso(23, 2, 2023)); // "23/02/2023"  
  
5. function valorDesconto(preco, percentual) {  
    const desconto = preco * (percentual / 100);  
    return preco - desconto;  
}  
// Estamos calculando um desconto 20% nos 100 reais = 80  
console.log(valorDesconto(100, 20));  
  
6. function repetir(palavra, vezes) {  
    let resultado = "";  
    for (let i = 0; i < vezes; i++) {  
        resultado += palavra;  
    }  
    return resultado;  
}  
console.log(repetir("batata", 3)); // "batatabatatabatata"
```

```
7. function invocarFuncao(funcao) {  
    funcao();  
}  
function minhaFuncao() {  
    console.log("Essa é a minha função!");  
}  
  
invocarFuncao(minhaFuncao);  
  
8. const dobrar = (numero) => numero * 2;  
const quadrado = (numero) => numero ** 2;  
const apresentacao = (nome, idade, endereco) =>  
    `Olá, eu sou ${nome}, tenho ${idade} anos e moro em ${endereco}.`;
```

## 2.6 Strings

```
1. function contarCaracteres(str) {  
    return str.length;  
}  
const minhaString = "Olá, mundo!";  
const numCaracteres = contarCaracteres(minhaString);  
console.log(numCaracteres); // 11  
  
2. function maiusculas(str) {  
    return str.toUpperCase();  
}  
const minhaString = "Olá, mundo!";  
const stringMaiuscula = maiusculas(minhaString);  
console.log(stringMaiuscula); // "OLÁ, MUNDO!"  
  
3. function minusculas(str) {  
    return str.toLowerCase();  
}  
const minhaString = "OLÁ, MUNDO!";  
const stringMinuscula = minusculas(minhaString);  
console.log(stringMinuscula); // "olá, mundo!"  
  
4. function saoIguais(str1, str2) {  
    if (str1 === str2) {  
        return true;  
    } else {  
        return false;  
    }  
}
```

```
    }  
  }  
  console.log(saoIguais("batata", "batatinha")); // false  
  
5. function ultimosCaracteres(str, num) {  
  if (num >= str.length) {  
    return str;  
  } else {  
    // Podemos usar negativo no slice para pegar os últimos caracteres  
    // Exemplo -1 representa o último caractere  
    return str.slice(-num);  
  }  
}  
const resultado = ultimosCaracteres("JavaScript", 4);  
console.log(resultado); // "ript"  
  
6. function censurar(palavra, frase) {  
  return frase.replaceAll(palavra, "***");  
}  
// "Eu gosto muito de ***"  
console.log(censurar("batata", "Eu gosto muito de batata"));  
  
7. function contarCaractere(str, caractere) {  
  let contador = 0;  
  for (let i = 0; i < str.length; i++) {  
    if (str[i] === caractere) {  
      contador++;  
    }  
  }  
  return contador;  
}  
console.log(contarCaractere("batata", "a")); // 3
```

## 2.7 Arrays

```
1. let numeros = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];  
  
for (let numero of numeros) {  
  console.log(numero);  
}  
  
for (let i = 0; i < numeros.length; i++) {  
  console.log(numeros[i]);  
}
```

```
}

2. function filtrarStrings(array) {
  let novoArray = [];

  for (let i = 0; i < array.length; i++) {
    let string = array[i];

    if (string.length > 5) {
      novoArray.push(string);
    }
  }

  return novoArray;
}

filtrarStrings(["Javascript", "VueJS", "Node", "Angular", "React", "CSS"]);

3. function filtrarNumerosMaioresQueMedia(array) {
  let soma = 0;
  for (let i = 0; i < array.length; i++) {
    soma += array[i];
  }

  const media = soma / array.length;
  const novoArray = [];

  for (let i = 0; i < array.length; i++) {
    const numero = array[i];

    if (numero > media) {
      novoArray.push(numero);
    }
  }

  return novoArray;
}

let resultadoArray = filtrarNumerosMaioresQueMedia([1, 2, 3, 4, 5, 6, 7, 8, 9, 10]);

4. let booleans = [true, false, true, true, false];

let resultado = false;

for (let i = 0; i < booleans.length; i++) {
```

```
    resultado = resultado || booleans[i];
  }

  console.log(resultado); // true

5. function listarFaltantes(alunos, listaPresenca) {
  let faltantes = [];

  for (let i = 0; i < alunos.length; i++) {
    let aluno = alunos[i];

    if (!listaPresenca.includes(aluno)) {
      faltantes.push(aluno);
    }
  }

  return faltantes;
}

const alunos = ['João', 'Maria', 'Pedro', 'Lucas', 'Ana', 'Mariana'];
const listaPresenca = ['João', 'Maria', 'Lucas', 'Mariana'];

const faltantes = listarFaltantes(alunos, listaPresenca); // ['Pedro', 'Ana']

6. function elevarAoQuadrado(array) {
  const resultado = [];

  for (let i = 0; i < array.length; i++) {
    resultado.push(array[i] ** 2);
  }

  return resultado;
}

const numeros = [1, 2, 3, 4, 5];

const aoQuadrado = elevarAoQuadrado(numeros);

console.log(aoQuadrado); // [1, 4, 9, 16, 25]

7. function filtrarPorExtensao(arquivos, extensao) {
  const resultado = [];

  for (let i = 0; i < arquivos.length; i++) {
```

```
    const arquivo = arquivos[i];

    if (arquivo.endsWith(`.${extensao}`)) {
        resultado.push(arquivo);
    }
}

return resultado;
}

const arquivos = ['relatorio.pdf', 'imagem.png', 'texto.doc', 'planilha.xlsx'];
const extensao = 'pdf';

const filtrados = filtrarPorExtensao(arquivos, extensao);

console.log(filtrados); // ['relatorio.pdf']

8. function gerarArrayAleatorio(tamanho) {
    const arrayAleatorio = [];

    for (let i = 0; i < tamanho; i++) {
        arrayAleatorio.push(Math.floor(Math.random() * 100));
    }

    return arrayAleatorio;
}

const tamanho = 5;
const arrayGerado = gerarArrayAleatorio(tamanho);

console.log(arrayGerado); // [ 57, 88, 16, 24, 92 ]

9. // Item 1
    numeros.forEach(numero => console.log(numero));

    // Item 2
    function filtrarStrings(array) {
        return array.filter(str => str.length > 5);
    }

    // Item 6
    function elevarAoQuadrado(numeros) {
        return numeros.map((num) => num ** 2);
    }
```



```
// Item 7
function filtrarArquivosPorExtensao(array, extensao) {
  return array.filter(arquivo => arquivo.endsWith(extensao));
}
```

