

Lista de Exercícios – JavaScript no Navegador

Resolva os exercícios codificando em uma página HTML usando a tag `<script>`

◆ Parte 1 – Declaração de Variáveis (let e const) (4 questões)

1. Declaração com let:

Crie uma variável `let idade` e atribua sua idade. Depois, atribua um novo valor representando a idade daqui a 5 anos. Mostre os dois valores no console: “Minha idade hoje é `<idade>` e daqui a 5 anos será `<nova_idade>`”

2. Uso de const:

- Declare uma constante **PI** com valor 3.14159. Tente atribuir outro valor a **PI** e veja o erro no console. Explique por que isso acontece.
- Declare uma constante **SemanasDoAno** sem atribuir valor. Na linha seguinte atribua o valor 52. O que ocorre?

3. Concatenação de Strings:

Crie três variáveis (`let nome`, `let sobrenome`, `let anoNascimento`) e mostre no console a frase:

“Olá, meu nome é `<nome completo>` e eu nasci em `<anoNascimento>`”. `<nome completo>` deve ser a concatenação de nome e sobrenome.

4. Diferença de Escopo:

Declare uma variável **let numero = 10** fora de um bloco `{}`. Dentro do bloco, declare novamente **let numero = 20**. Mostre o valor da variável dentro e fora do bloco com **console.log**.

◆ Parte 2 – Estruturas Condicionais (if, else if, else e operador ternário) (4 questões)

5. Verificação de Idade:

Peça ao usuário (com prompt) sua idade.

- Se for menor que 12, mostre “Criança”.
- Se estiver entre 12 e 17, mostre “Adolescente”.
- Se for maior ou igual a 18, mostre “Adulto”.

6. Número Par ou Ímpar:

Receba um número via prompt e use `if/else` para verificar se é par ou ímpar.

7. Nota de Aluno:

Receba uma nota de 0 a 10.

- `>= 7`: “Aprovado”.
- `>= 5 e < 7`: “Recuperação”.
- `< 5`: “Reprovado”.

8. Operador Ternário:

Receba um número via prompt e mostre no console:

"Positivo" se maior que 0, "Negativo" se menor que 0, ou "Zero" caso contrário, usando **apenas operador ternário**.

◆ **Parte 3 – Laços de Repetição (for, while, for...in, for...of) (4 questões)**

9. **Tabuada com for:**

Receba um número e mostre a tabuada de 1 a 10 usando for.

10. **Contagem Regressiva com while:**

Mostre no console a contagem regressiva de 10 até 1 com while.

11. **Percorrendo Objetos com for...in:**

Crie um objeto pessoa com nome, idade e cidade. Use for...in para listar suas propriedades e valores.

12. **Percorrendo Arrays com for...of:**

Crie um array de frutas e use for...of para mostrar cada fruta no console.

◆ **Parte 4 – Objetos (3 questões)**

13. **Criação de Objeto:**

Crie um objeto carro com propriedades marca, modelo e ano. Mostre todas as propriedades no console, usando uma instrução console.log para cada uma das propriedades.

14. **Método em Objeto:**

Adicione ao objeto carro um método descricao() que retorne uma string: "Marca: <marca>, Modelo: <modelo>, Ano: <ano>".
Chame esse método e mostre o resultado no console.

15. **Manipulação de Propriedades:**

Crie um objeto livro com título e autor.

- Adicione dinamicamente a propriedade ano.
 - Altere o título.
 - Exclua a propriedade autor.
 - Mostre o objeto atualizado no console.
-

◆ **Parte 5 – Arrays com map, filter, sort (4 questões)**

16. **Dobrar Valores (map):**

Dado o array [1, 2, 3, 4, 5], use **map** para criar um novo array com cada número dobrado.

17. **Filtrar Números Pares (filter):**

Dado o array [10, 15, 20, 25, 30], use **filter** para criar um array apenas com números pares.

18. Ordenação (sort):

Dado o array [5, 1, 9, 3, 7], ordene-o em ordem crescente usando sort.

19. Encadeando Métodos:

Dado o array [2, 5, 8, 11, 14], use filter para selecionar apenas os números maiores que 5 e depois map para multiplicá-los por 2. Mostre o resultado final.

20. Refaça os exercícios que precisam de entrada de dados substituindo o prompt por um formulário. Também faça com que a resposta apareça na própria página HTML em lugar do console.

Parte 6 – Controle de Fila de Atendimento

21. Escreva um programa para controlar uma fila de atendimento, deve guardar o nome da pessoa ao clicar no botão "entrar na fila" e inserir no final de uma fila. Quando a pessoa for atendida, deve clicar no botão "atender" que vai retirar o nome da pessoa que está no primeiro lugar na fila. Também guarde a data e hora que a pessoa entrou na fila (use **Date.now()** para obter essa informação . Veja como formatar a data usando **Intl.DateTimeFormat**).
- Quando ela for atendida (retirada da fila), deve calcular quanto tempo levou para ser atendida.
- Use vetores para armazenar as informações. Para inserir no final da fila use o método "push()" e para remover do início da fila use o método "shift()" . Crie um vetor para armazenar o nome e a hora (hora, minutos e segundos) que a pessoa entrou na fila. Em outro vetor armazene o nome, data e hora que pessoa entrou na fila, data e hora que foi atendida e quanto tempo demorou para ser atendida. Cada vez que uma pessoa entrar na fila ou for atendida deve exibir os dados armazenados nos dois vetores.