

Lista de Exercícios - JavaScript Avançado

Gisely Luca Bernardino

Lembre-se: mais importante do que resolver os exercícios é entender o motivo daquele resultado.

1 - Crie uma função que escreva uma mensagem informando se um número é par ou ímpar.

Exemplo 1:

Entrada:

2

Saída:

O número 2 é par

Exemplo 2:

Entrada:

77

Saída:

O número 77 é ímpar

2 - Crie uma nova função semelhante ao exercício anterior, porém agora ela deve retornar o texto que será usado por um console.log.

Exemplo 1:

Entrada:

2

Saída:

identificaParOuImpar(2) // retorna "par"

O número 2 é par // retorno do console.log que chama a função

3 - Crie uma função que inverte a ordem dos itens de um array:

Exemplo 1:

Entrada:

["Abacaxi", "Uva", "Banana", "Maçã", "Goiaba"]

Saída:

["Goiaba", "Maçã", "Banana", "Uva", "Abacaxi"]

Exemplo 2:

Entrada:

[10, 2, 45, 87, 99, 20, 1, 3]

Saída:

[3, 1, 20, 99, 87, 45, 2, 10]

4 - Crie uma função que retorne a soma entre dois números.

Exemplo 1:

Entrada:
soma(3, 4)
Saída:
3 + 4 = 7

5 - Faça uma função que irá receber um array de ingredientes e retornar um log com a mensagem de saída:

Exemplo 1:

Entrada:
["Pão", "Carne", "ovo", "bacon", "Mussarela", "Presunto", "Alface", "Tomate",
"Batata Palha"]
Saída:
Pão, Carne, ovo, bacon, Mussarela, Presunto, Alface, Tomate e Batata Palha

6 - Complemente o exercício anterior com outra função que quando chamada irá adicionar o ingrediente informado como parâmetro ao array de ingredientes e escreve uma nova mensagem de saída:

Exemplo 1:

Entrada:
array: ["Pão", "Carne", "ovo", "bacon", "Mussarela", "Presunto", "Alface",
"Tomate", "Batata Palha"]
adicionarIngrediente("Milho")

Saída:
Pão, Carne, ovo, bacon, Mussarela, Presunto, Alface, Tomate, Batata Palha
e Milho


7 - Crie uma função que receba os dados: Nome, idade e endereço e retorne um objeto com essas propriedades.

Exemplo 1:

Entrada:
preencherNome("Camila", 23, "Rua dos Andradas, 21")

Saída:
{
 nome: "Camila",
 idade: 23,
 endereco: "Rua dos Andradas, 21"
}


8 - Crie uma função que retorne a soma de todas as quantidades exibidas no carrinho abaixo:



jaqueta trucker jeans manga longa
com forro de pelo azul médio g

R\$ 249,99
 - 1 +
 R\$ 249,99


vendido e entregue por **CeA**
[excluir da sacola](#)



mochilete média ever after high
17z

de ~~R\$ 399,90~~
por **R\$ 62,90**
R\$ 337,00 mais barato


vendido e entregue por **Mundo Bag**
[excluir da sacola](#)



óculos de sol aviador ace dourado
único

R\$ 69,99
 - 2 +
 R\$ 139,98

vendido e entregue por **CeA**
[excluir da sacola](#)



suéter de pelúcia sherpa com zíper
bege m

R\$ 199,99
 - 1 +
 R\$ 199,99

vendido e entregue por **CeA**
[excluir da sacola](#)

subtotal
 R\$ 652,86

código do consultor

 adicionar

entrega
 calcular

adicionar cupom de desconto

Se você possui um Vale-troca ou Cartão Presente eles devem ser informados na etapa de pagamento do seu pedido.

total
 em até **5x de R\$ 130,57**
no Cartão C&A sem juros
ou **R\$ 652,86** à vista

FECHAR PEDIDO

dinheiro de volta garantido, devolva pelo site o que não servir e receba seu dinheiro de volta.

[escolher mais produtos](#)

9 - Crie as funções acima utilizando array functions

Manipulando Listas e Objetos

10 - Faça uma função que retorne o dobre dos números do array abaixo.

[0, 2, 98, 75, 23, 18, 31, 7, 1]

11 - Faça uma função que retorne todas as frutas que tenham a letra "o"

["Abacaxi", "Uva", "Maçã", "Goiaba", "Abacate", "Acerola"]

12 - Faça uma função que retorne a primeira fruta que tenha a letra "o"

["Abacaxi", "Uva", "Maçã", "Goiaba", "Abacate", "Acerola"]

13 - Faça o somatório de todos os valores do array abaixo, utilizando reduce

[1, 2, 3, 4, 5, 6]

14 - Faça uma função que retorne apenas as idades das pessoas, dado o objeto abaixo:

```
[
  { name: 'Camila', age: 17 },
  { name: 'Jorge', age: 13 },
  { name: 'Marcia', age: 56 },
  { name: 'Barbara', age: 24 },
  { name: 'Natan', age: 67 },
  { name: 'Pablo', age: 65 }
];
```

15 - Resolver as funções anteriores agora utilizando apenas array function

16 - Crie uma função que una os objetos abaixo em um outro objeto chamado `pessoa`

```
primeiroNome: { nome: 'João' }  
sobrenome: { sobrenome: 'da Silva' }
```

Crie uma função que crie um novo objeto alterando apenas as propriedades abaixo:

```
produto: {  
  nome: "X-Tudo",  
  ingredientes: ["Pão", "Carne", "ovo", "bacon", "Mussarela", "Presunto", "Alface", "Tomate",  
    "Batata Palha"],  
  servePessoas: 1,  
  valor: 24.98,  
  valorPromocional: 16.98,  
  maisVendido: true  
}
```

Alterar o valor do produto para 35.00 e o valor promocional deve ser o valor do produto com 10% de desconto.

17 - Faça uma função que retorne o primeiro produto com valor entre 50 e 60 reais.

Itens pedidos	Preço
1 unidade(s) de: <i>Desafios e Enigmas: uma Forma Descontraída de Colocar à Prova seu Raciocínio</i> , Niederauer, Juliano Vendido por: Amazon Serviços de Varejo do Brasil, Ltda.	R\$ 47,20
Condição: Novo	
1 unidade(s) de: <i>Mulheres que correm com os lobos</i> , Estés, Clarissa Pinkola Vendido por: Amazon Serviços de Varejo do Brasil, Ltda.	R\$ 37,90
Condição: Novo	
1 unidade(s) de: <i>O oráculo da noite: A história e a ciência do sonho</i> , Ribeiro, Sidarta Vendido por: Amazon Serviços de Varejo do Brasil, Ltda.	R\$ 66,31
Condição: Novo	
1 unidade(s) de: <i>Seja homem: a masculinidade desmascarada</i> , Bola, JJ Vendido por: Amazon Serviços de Varejo do Brasil, Ltda.	R\$ 53,71
Condição: Novo	
1 unidade(s) de: <i>Mulheres incríveis</i> , Cultural, Ciranda Vendido por: Amazon Serviços de Varejo do Brasil, Ltda.	R\$ 14,37
Condição: Novo	
1 unidade(s) de: <i>A ciranda das mulheres sábias</i> , Estés, Clarissa Pinkola Vendido por: Amazon Serviços de Varejo do Brasil, Ltda.	R\$ 13,90
Condição: Novo	
1 unidade(s) de: <i>Sociedade do cansaço</i> , Han, Byung-Chul Vendido por: Amazon Serviços de Varejo do Brasil, Ltda.	R\$ 23,20
Condição: Novo	
1 unidade(s) de: <i>Como o racismo criou o Brasil</i> , Souza, Jessé Vendido por: Amazon Serviços de Varejo do Brasil, Ltda.	R\$ 24,99

18 - Utilizando o objeto criado no exercício anterior, faça uma função que retorne a soma dos itens na lista de compras (utiliza o `reduce`)

19 - Utilizando o objeto criado no exercício anterior, faça uma função que retorne o nome dos produtos que tenham o valor maior do que 25 reais.

Desafio

Busca por voo:

Vamos construir um conjunto de funções que vai salvar os dados de entrada sobre voos que estão para acontecer e depois podemos buscar algumas informações sobre os dados armazenados.

- Deve ser criada uma função que quando chamada seja passado os seguintes valores do voo como parâmetro: código, horário de partida, quantidade de passageiros, valor da passagem, origem e destino. Lembrete: qual a melhor estrutura para representar esses dados? Deve ser possível armazenar mais de um objeto. Sempre que a função for chamada, deve ser adicionado um item ao final, com valores corretos.

Exemplo:

Entrada:

```
cadastrarVoo(1, "10:32", 120, 1220, "SP", "RE")
```

- Deve ser criada uma função que quando chamada deve retornar todos os voos disponíveis para a origem informada.

Exemplo:

Entrada:

```
buscarVooOrigem("SP")
```

- Deve ser criada uma função que quando chamada deve retornar todos os voos disponíveis para o destino informado.

Exemplo:

Entrada:

```
buscarVooOrigem("SP")
```

Jogo da Velha:

Vamos construir um jogo da velha, mas para isso algumas funções devem ser criadas. Siga cada exercício dessa etapa, para concluir o desafio

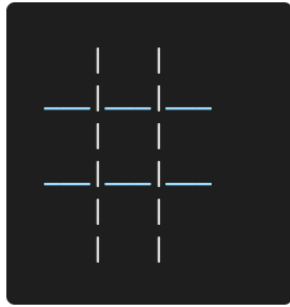
- Para cada jogada feita o programa deve ser capaz de escrever como está a visão do tabuleiro. Crie uma função que irá "desenhar" o jogo da velha sempre que for chamada.

Exemplo:

Entrada:

```
jogoVelha = [
  ["", "", ""],
  ["", "", ""],
  ["", "", ""]
]
desenhaJogoVelha()
```

Saída:

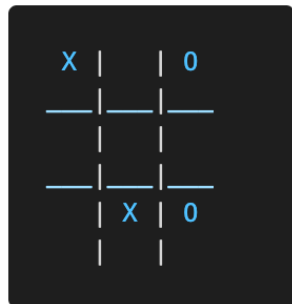


Exemplo 2:

Entrada:

```
jogoVelha = [['X', ' ', 'O'], [' ', ' ', ' '], [' ', 'X', 'O']]  
desenhaJogoVelha()
```

Saída:



- Faça uma função chamada jogar, que deve receber como parametro o número da linha (1 a 3), o número da coluna (1 a 3) e qual vai ser a jogada no momento ("X" ou "O"). Toda vez que essa função for chamada, a matriz jogoDaVelha deve ser preenchida corretamente, e o jogo da velha deve ser "desenhado" no console.
- Faça uma função que retorne uma mensagem de erro com o texto "Jogada Inválida" sempre que a função jogar for chamada descumprindo alguma das regras abaixo:
 - Número da linha com valor diferente de 1, 2 ou 3
 - Número da coluna com valor diferente de 1, 2 ou 3
 - Valor da jogada diferente de "X" ou "O"
 - Informar uma posição que não esteja vazia.
- Faça funções que identifiquem ao final de uma jogada qual a situação do Jogo:
 - "Jogo em Andamento!"
 - "O Jogador X ganhou"
 - "O Jogador O ganhou"
 - "O jogo deu Velha"