

# Projeto de Tags

## Questão 1

Relatório técnico.....	02
Codigo.....	03
Teste com DevTool.....	04
Teste Com Visual Code.....	05
Resposta Email 1.....	06

## Questão 2

Relatório técnico .....	07
Codigo.....	07
Teste com DevTool.....	08
Teste Com Visual Code.....	08
Reposta Email 2.....	09

## Questão 3

Relatório técnico .....	09
Código .....	10
Teste com DevTool.....	10
Teste Com Visual Code.....	10
Reposta Email 3.....	11

## Relatório técnico

### Problemas apontados:

Produtos duplicados;

Produtos zerado no estoque;

Criar variável com os produtos corretos;

### Solução:

Verifiquei que as variáveis “product\_price”, “product\_stock\_quantity” estão no formato string, converti os mesmos para número usando a “função parseFloat” para número flutuantes e “função parseInt.” para número real, usando a “função forEach” para percorrer o array.

```
currentProduct.product_price =parseFloat(currentProduct.product_price);  
currentProduct.product_stock_quantity  
=parseInt(currentProduct.product_stock_quantity);
```

### Criando variável do tipo lista:

Nova variável para receber os atributos tratados do array “cart\_products” conforme solicitado pelo cliente.

```
var cart_products_sanitized = [];
```

### Produtos zerados:

Fiz uma comparação usando uma tomada de decisão para tirar os produtos zerados array, com os elemento maior que zero, usei “função forEach” para percorrer o array.

```
cart_products.forEach((currentProduct) => {  
  
    if(currentProduct.product_stock_quantity > 0)
```

### Produtos duplicados:

Usei a “função findIndex” para pegar o índice do elemento e comparei com array usando uma tomada de decisão para não adicionar as duplicatas.

Usei outra tomada de decisão para adicionar os produtos não duplicados com “função push” para adicionar os elementos no novo array.

### Código:

```
var cart_products_sanitized = [];  
cart_products.forEach((currentProduct) => {  
  
    if(currentProduct.product_stock_quantity > 0){  
        var duplicated = cart_products_sanitized.findIndex(id =>{  
            return currentProduct.product_id == id.product_id;  
        })>-1;  
        if(!duplicated) {  
            cart_products_sanitized.push(currentProduct);  
        }  
    }  
    currentProduct.product_price  
=parseFloat(currentProduct.product_price)  
    currentProduct.product_stock_quantity  
=parseInt(currentProduct.product_stock_quantity);  
});  
  
console.log(cart_products_sanitized)
```

### Teste:

Foi desenvolvido usando a IDE Visual Studio Code. Testei no DevTool assim como solicitado e os resultados foram os mesmos.

## Teste no DevTool

```
▼ (3) [{...}, {...}, {...}] ⓘ  
  ▼ 0:  
    product_brand: "Rozar"  
    product_cart_quantity: "1"  
    product_category: "peripherals"  
    product_id: "IN2350"  
    product_name: "Teclado Mecânico Rozar K600"  
    product_price: 650  
    product_stock_quantity: 30  
    product_url: "http://www.foo.com/p/IN2350"  
    ► [[Prototype]]: Object  
  ▼ 1:  
    product_brand: "GL"  
    product_cart_quantity: "1"  
    product_category: "monitor"  
    product_id: "IN4566"  
    product_name: "Monitor Gamer LED GL UltraWide 25\""  
    product_price: 1399.99  
    product_stock_quantity: 5  
    product_url: "http://www.foo.com/p/IN4566"  
    ► [[Prototype]]: Object  
  ▼ 2:  
    product_brand: "Fenix"  
    product_cart_quantity: "5"  
    product_category: "office"  
    product_id: "ES7112"  
    product_name: "Pacote de Post-it Fenix 50 unidades"  
    product_price: 14.95  
    product_stock_quantity: 22  
    product_url: "http://www.foo.com/p/ES7112"  
    ► [[Prototype]]: Object  
  length: 3  
  ► [[Prototype]]: Array(0)
```

## Teste no Visual Code

```
PROBLEMAS  SAÍDA  TERMINAL  CONSOLE DE DEPURAÇÃO

[
  {
    product_id: 'IN2350',
    product_name: 'Teclado Mecânico Rozar K600',
    product_price: 650,
    product_cart_quantity: '1',
    product_url: 'http://www.foo.com/p/IN2350',
    product_category: 'peripherals',
    product_stock_quantity: 30,
    product_brand: 'Rozar'
  },
  {
    product_id: 'IN4566',
    product_name: 'Monitor Gamer LED GL UltraWide 25"',
    product_price: 1399.99,
    product_cart_quantity: '1',
    product_url: 'http://www.foo.com/p/IN4566',
    product_category: 'monitor',
    product_stock_quantity: 5,
    product_brand: 'GL'
  },
  {
    product_id: 'ES7112',
    product_name: 'Pacote de Post-it Fenix 50 unidades',
    product_price: 14.95,
    product_cart_quantity: '5',
    product_url: 'http://www.foo.com/p/ES7112',
    product_category: 'office',
    product_stock_quantity: 22,
    product_brand: 'Fenix'
  }
]
```

**Resposta Email 1:**

Analizamos e corrigimos os erros, criamos a variável solicitada pelo cliente, que apresenta apenas os itens com estoque acima de zero e as duplicatas foram solucionadas.

Foram corrigidos os elementos “product\_cart\_quantity” e “product\_price” que estavam no formato de texto, colocando-os como números.

Qualquer dúvida estamos à disposição!

Att. Equipe.

## **Relatório técnico**

### **Problemas apontados:**

Produto no formado errado;

Tipo de dado com formado errado;

Criar variável com os produtos corretos;

### **Solução:**

Declarei a nova variável como solicitado para receber os elementos corrigidos, usei a “função map” para adicionar na nova variável já alterando o nome do atributo “price” para “item\_price”.

```
var new_products_list = products_list.map(p =>{
  return {
    id: p.id,
    nome: p.name,
    item_price: p.price
  };
});
```

Com o novo array criado com os elementos corrigindo usei a “função forEach” para percorrer o vetor, a “função parseInt” para converter String para número, usei a “função replace” para tirar símbolo de reais (R\$).

### Código:

```
var new_products_list = products_list.map(p =>{
  return {
    id: p.id,
    nome: p.name,
    item_price: p.price
  };
});
new_products_list.forEach((element)=>{
  element.item_price =
  Number(parseInt(element.item_price.replace("R$", "")));
});

console.log(new_products_list)
```

### Teste:

Foi desenvolvendo usando IDE Visual Studio Code testei no DevTool assim como solicitado o resultado foram o mesmo.



## Teste no DevTool:

```
▼ (6) [{...}, {...}, {...}, {...}, {...}, {...}] ⓘ  
  ▶ 0: {id: '35', nome: 'Miniatura Homem de Ferro', item_price: 50}  
  ▶ 1: {id: '30', nome: 'Miniatura Magneto', item_price: 57}  
  ▶ 2: {id: '25', nome: 'Miniatura Super Homem', item_price: 37}  
  ▶ 3: {id: '71', nome: 'Miniatura Bernard', item_price: 71}  
  ▶ 4: {id: '56', nome: 'Miniatura Batman', item_price: 23}  
  ▶ 5: {id: '56', nome: 'Miniatura Darth Vader', item_price: 89}  
    length: 6  
  ▶ [[Prototype]]: Array(0)
```

## Teste no Visual Code:

```
PROBLEMAS  SAÍDA  TERMINAL  CONSOLE DE DEPURAÇÃO  Code  
[Running] node "c:\xampp\htdocs\RaccomTag\questao2.js"  
[  
  { id: '35', nome: 'Miniatura Homem de Ferro', item_price: 50 },  
  { id: '30', nome: 'Miniatura Magneto', item_price: 57 },  
  { id: '25', nome: 'Miniatura Super Homem', item_price: 37 },  
  { id: '71', nome: 'Miniatura Bernard', item_price: 71 },  
  { id: '56', nome: 'Miniatura Batman', item_price: 23 },  
  { id: '56', nome: 'Miniatura Darth Vader', item_price: 89 }  
]
```

## Resposta Email 2:

Olá, boa noite!

Analizamos e corrigimos os erros, criamos variável “new\_products\_list” e alterado o nome do elemento “price” para “item\_price” e passado o valor para número como solicitado pelo cliente.

Qualquer dúvida estamos à disposição!

Att. Equipe.

## Relatório técnico

### Problemas apontados:

Criar variável com valor total do produto;

Elementos com o valor undefined;

Informação do frete com tipo booleano;

### Solução;

Verifiquei que as variáveis “product\_price”, “product\_stock\_quantity” estão no formado String, converti os mesmo para número usando a função “parseFloat” para número flutuantes e “parseInt” para número real, usando a função forEach para percorrer o array.

```
cart_products.forEach((element) => {  
  element.product_price =  
  Number(parseFloat(element.product_price).toFixed(2));  
  element.product_quantity = Number(element.product_quantity);  
});
```

Para resolver os problema apontados pelo cliente criei um function getCalcular e usei a função “reduce()” no elementos “product\_price” e “product\_quantity” para calcular o total dos produtos.

Para resolver a questão do frete usei uma tomada de decisão com operador ternário, com a variável “freeShipping” recebe “true” se o valor for maior que 300 ou “false” se o total for menor que 300.

```
function getCalcula(){  
  totalPrice = cart_products.reduce((preVel, element)=>{  
    return preVel + element.product_price * element.product_quantity;  
  },0);  
  var freeShipping = (totalPrice > 300) ? true: false  
  return{  
    totalPrice,  
    freeShipping  
  };  
};
```

Crie a variável “cartPrincesInfo” para receber a função criada e usei o método de desconstrução atribuindo as variáveis “totalPrice” e “freeShipping” direto no “console.log”.

```
var cartPricesInfo = getCalcula();  
console.log("cartPrincesInfo = ",cartPricesInfo);
```

```
console.log("totalPrice=", cartPricesInfo.totalPrice);
console.log("freeShipping =", cartPricesInfo.freeShipping);
```

### Código:

```
cart_products.forEach((element) => {
    element.product_price =
    Number(parseFloat(element.product_price).toFixed(2));
    element.product_quantity = Number(element.product_quantity);
});

function getCalcula(){
    totalPrice = cart_products.reduce((preVel, element)=>{
        return preVel + element.product_price * element.product_quantity;
    },0);
    var freeShipping = (totalPrice > 300) ? true: false
    return{
        totalPrice,
        freeShipping
    };
};

var cartPricesInfo = getCalcula();
console.log("cartPrincesInfo = ", cartPricesInfo);
console.log("totalPrice=", cartPricesInfo.totalPrice);
console.log("freeShipping =", cartPricesInfo.freeShipping);
```

### Teste:

Foi desenvolvendo usando IDE Visual Studio Code testei no DevTool assim como solicitado o resultado foram o mesmo.

### Teste no DevTool:

```
cartPrincesInfo = {totalPrice: 2137.74, freeShipping: true}
totalPrice= 2137.74
freeShipping = true
← undefined
```

### Teste no Visual Code:

```
PROBLEMAS  SAÍDA  TERMINAL  CONSOLE DE DEPURAÇÃO

[Running] node "c:\xampp\htdocs\RaccomTag\questao3.js"
cartPrincesInfo = { totalPrice: 2137.74, freeShipping: true }
totalPrice= 2137.74
freeShipping = true
```

### Resposta Email 3:

Olá, boa noite!

Analizamos e corrigimos os erros, criamos a variável solicitada pelo cliente, apresentado o valor total dos produtos do carrinho e o frete com valor true para compras acima de 300 reais.

Qualquer dúvida estamos à disposição!

Att. Equipe.

