

Candidato: João Pedro Nunes Silva

Vaga estágio: Projeto Rocky – Web Analytics

#### 1- Módulo File System (fs)

```
1 const fs = require("fs"); // permite salvar e abrir arquivos JSON, através do módulo filesystem
```

É um módulo que permite a abertura de arquivos, além disso permite ao desenvolvedor modificar e salvar esses arquivos.

#### 2- Função Read file Sync

```
3 // função que permite ler o arquivo de maneira síncrona e retornar seu conteúdo — https://www.geek  
4 function readFile() {  
5     const originalData = fs.readFileSync("./broken-database.json", "utf8", (err, jsonString) => {  
6         if (err) {  
7             console.log("File read failed:", err);  
8             return;  
9         }  
10    });
```

Essa função faz com que o arquivo possa ser lido de maneira síncrona e também tem a capacidade de retornar seu conteúdo.

#### 3- constante dataObject

```
12     const dataObject = JSON.parse(originalData)  
13     return dataObject;  
14 }
```

Cria uma constante chamada dataObject, que transforma as informações do arquivo JSON do tipo string para o tipo objeto.

#### 4- Função fixName

```
15 //função que corrige os nomes que estavam corrompidos  
16 function fixName(name) {  
17     return name.replace(/æ/g, "a").replace(/ç/g, "c").replace(/ø/g, "o").replace(/ß/g, "b")  
18 }
```

Cria uma função que através do comando replace corrige as letras que estavam corrompidas, para isso o parâmetro global utilizado é o (/g)

#### 5- Função fixPrice

```
19 //função que corrige os preços, transformando os valores que estavam no formato string para number  
20 function fixPrice(price) {  
21     return Number(price)  
22 }
```

Através da função, os dados armazenados na variável “price” são convertidos de string para number.

#### 6- Função fixQuantity

```
23 //função que corrige as quantidades, caso a quantidade seja 0 ela retorna
24 function fixQuantity(quantity) {
25     return quantity || 0
26 }
```

Função que retorna o próprio valor de 'quantity' e caso não haja nenhum valor, a função retorna 0.

#### 7- Função writeFile

```
27 //salva o arquivo com o nome saida.json
28 function writeFile(dataObject) {
29     const jsonString = JSON.stringify(dataObject, null, 2);
30     fs.writeFile('./saida.json', jsonString, err => {
31         if (err) {
32             console.log('Error writing file', err)
33         } else {
34             console.log('Successfully wrote file')
35         }
36     })
37 }
```

Transforma o objeto JavaScript novamente em arquivo tipo .JSON e realiza o salvamento do arquivo com o nome "saida.json".

#### 8- Função showSorted

```
38 // ordena as categorias por ordem alfabética e após isso por ID crescente
39 function showSorted(dataObject) {
40     const sortedData = dataObject.sort((a, b) => {
41         if (a.category > b.category) return 1;
42         if (a.category < b.category) return -1;
43         return a.id - b.id;
44     })
45     const sortedNames = sortedData.map(function (item) {
46         return item.name
47     })
48
49     console.log(sortedNames)
50 }
51 }
```

Faz o ordenamento das categorias, primeiramente por ordem alfabética e logo após isso por ID crescente.

## 9- Função categoryValue

```
52 //função que calcula o valor total do estoque considerando a categoria e quantidade de cada produto
53 function categoryValue(dataObject) {
54     const categories = []
55
56     for (let i = 0; i < dataObject.length; i++) {
57         if (!categories.includes(dataObject[i].category)) {
58             categories.push(dataObject[i].category)
59         }
60     }
61
62     for (let i = 0; i < categories.length; i++) {
63         let value = 0;
64
65         const filteredData = dataObject.filter(product => product.category == categories[i])
66         for (let j = 0; j < filteredData.length; j++) {
67             value += filteredData[j].price * filteredData[j].quantity;
68         }
69
70         console.log(categories[i] + ': R$ ' + value);
71     }
72 }
```

Cria uma função, que calcula o valor total do estoque, a partir de sua categoria e quantidade disponível de cada produto.

## 10-Função main

```
73 //função que organiza as outras funções criadas
74 function main() {
75     const dataObject = readFile()
76
77     for (let i = 0; i < dataObject.length; i++) {
78         dataObject[i].price = fixPrice(dataObject[i].price)
79         dataObject[i].quantity = fixQuantity(dataObject[i].quantity)
80         dataObject[i].name = fixName(dataObject[i].name)
81     }
82     writeFile(dataObject);
83     showSorted(dataObject);
84     categoryValue(dataObject);
85 }
```

Uma função “principal” que armazena as outras funções desenvolvidas, deixando o código mais organizado e adequado, além disso executa as funções “writeFile”, “showSorted” e “categoryValue”.

Referências utilizadas:

[https://heynode.com/tutorial/readwrite-json-files-nodejs/#:~:text=The%20simplest%20way%20to%20read,require\(%22.%2Fconfig](https://heynode.com/tutorial/readwrite-json-files-nodejs/#:~:text=The%20simplest%20way%20to%20read,require(%22.%2Fconfig)

<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Statements/for>

<https://love2dev.com/blog/javascript-for-loop-foreach/#:~:text=The%20Basic%20For%20Loop,with%20an%20index%20of%2000.&text=As%20you%20can%20see%20the,condition%2C%20and%20the%20final%20expression>

<https://dev.to/sanchithasr/7-ways-to-convert-a-string-to-number-in-javascript-4l>

<https://ricardo-reis.medium.com/o-m%C3%A9todo-sort-do-array-javascript-482576734e0a#:~:text=Ordenando%20um%20array%20de%20n%C3%BAmeros,-Suponha%20que%20voc%C3%AA&text=Para%20classificar%20um%20array%20de,scores%20numericamente%20em%20ordem%20crescente>

<https://gomakethings.com/sorting-an-array-by-multiple-criteria-with-vanilla-javascript/>

[https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global\\_Objects/Array/map](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/Array/map)

[https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global\\_Objects/JSON/parse](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/JSON/parse)

<https://www.geeksforgeeks.org/node-js-fs-readfilesync-method/>