

## **Projeto Classificatório – Processo Seletivo TI**

No projeto solicitado, escolhi utilizar a linguagem JavaScript pois tinha interesse prévio na linguagem, estava começando a estudar seus fundamentos na faculdade e paralelamente na internet, também já tinha softwares necessários para auxiliar na criação do projeto. Seguem as explicações das lógicas utilizadas em cada função solicitada.

### **1. Recuperação dos dados originais do banco de dados**

#### **a) Ler o arquivo Json:**

Inicialmente a função “lerArquivo()” armazena os dados do arquivo “broken-database.json” na variável “obj” que é um Array de objetos, buscando-os através do método “require”.

#### **b) Corrigir nomes:**

A função “corrigirNome()” executa “lerArquivo()” e com um loop percorre o Array “obj” até sua última posição, um loop interno percorre as Strings do atributo “name” e através do método “replace()”, substitui os caracteres solicitados.

#### **c) Corrigir preços:**

A função “corrigirPreco()” executa “lerArquivo()”, percorre “obj” e converte todos os atributos “price” para o tipo “float”.

#### **d) Corrigir quantidades:**

A função “corrigirQuantidade()” executa “lerArquivo()”, percorre “obj” e testa se o valor do atributo “quantity” é igual a “undefined”, e se sim, armazena 0 no valor do atributo.

#### **e) Exportar um arquivo JSON com o banco corrigido:**

A função “exportar()” executa as quatro funções anteriores, chama o módulo “fs” através do método require() e o atribui a variável “fs” possibilitando utilizar o método “writeFileSync()” que escreve todo o conteúdo de “obj” no arquivo externo “saída.json”.

## **2. Validação do banco de dados corrigido**

a) Imprimir a lista de nomes ordenada por categoria e depois por id:

Primeiro, a função “lerSaida()” funciona exatamente como “lerArquivo()”, porém utilizando o arquivo “saída.json” e o Array “obj2”.

Para a ordenação por categoria, na função “listaOrdenada()”, inicialmente executa “lerArquivo()” e utiliza o algoritmo “selection sort”, que contém um loop para percorrer “obj2” e indicar o objeto que terá o atributo “category” testado e um loop interno percorre “obj2” desde a próxima posição até o fim do Array procurando o menor valor. De volta ao loop externo, uma variável auxiliar recebe o menor valor, o objeto que armazena o menor valor recebe o valor o que está no índice atual e o objeto atual recebe o valor da variável auxiliar.

Para ordenar os id’s em ordem crescente, o mesmo algoritmo é utilizado, porém, é testado se um id é menor que outro e se o valor da categoria não é maior que o outro, mantendo assim a precedência da ordenação. Ao final, imprimem-se todos os valores com atributo “name”.

b) Quantidade em estoque de cada categoria:

A função “totalCategoria()” executa “lerSaida()” inicializa as variáveis “panelas”, “eletrodomesticos”, “eletronicos” e “acessorios” todas em 0, percorre “obj2” e no primeiro caso testa se a String de “category do objeto atual é, por exemplo, igual à “Panelas”, se sim adiciona 1 à panelas, o mesmo ocorre com os outros, adicionando os valores em suas variáveis correspondentes.