

# **Projeto Classificatório**

## **Explicação dos Códigos**

Processo seletivo - Tech  
Louise Queiroz da Silva Bezerra

**2024**

## Situação problema

Você é responsável por um software de gestão de uma concessionária multimarcas, sua gestão solicitou um relatório de desempenho das vendas dos veículos no último ano.

Ao verificar o banco de dados você percebeu que algumas tabelas foram corrompidas em uma atualização. Todos os nomes de veículos e marcas tiveram alguns caracteres modificados, houve substituição de todos os "a" por "æ", "o" por "ø",.

É preciso reverter essas substituições para recuperar os nomes originais antes de construir o relatório.

Você encontrou um problema com os valores de vendas também, eles devem ser sempre do tipo number, mas alguns deles estão no tipo string. É necessário transformar as strings novamente em number.

## Resolução

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

#### [Código em JS.](#)

Para a criação do código em JS foi solicitado a construção de 4 funções:

- Ler os arquivos Json;  
A função **readJson** é usada para ler um arquivo JSON de um caminho fornecido. Foi utilizado **'const fs = require('fs')'** para importar o módulo fs (file system) do Node.js facilitando a integração do arquivo. A função aceita como parâmetro o nome de um arquivo e inicia um bloco try para prevenir riscos de leitura. Com isso, ele usa o método `readFileSync()` do módulo fs para ler o arquivo e armazenar em uma variável. Foi utilizado o `JSON.parse()` para transformar a string do arquivo em um objeto JavaScript para facilitar a manipulação do mesmo. Os blocos de catch servem para capturar algum erro durante a execução.
- Corrigir nomes de marca e veículo;  
A função **correctStrings** resolve um problema de padronização no JSON, pois vários caracteres estavam modificados, ela recebe como parâmetro o objeto criado na função de abertura do arquivo, e chama uma função interna para processar a string e usar o método `replaceAll()` para resolver todos os casos especiais de caracter estranho, deste modo corrigimos a string. A ideia principal era fazer um map e mapear os caracteres e usar o `replaceAll()`, porém com o OBJ ficou mais simples,

então foi utilizado o `typeof` para concluir se a função havia sido finalizada e o OBJ tratado.

- Corrigir vendas;

A abordagem para a função **correctSales** é similar a de strings, porém ele mapeia os objetos e chega o tipo do dado de vendas, usando o método `typeof` é possível concluir se o número é um string ou um int, caso seja um string ele é convertido utilizando o método `parseInt()` e passa para o próximo objeto.

- Exportar um arquivo JSON com o banco corrigido.

A função **exportJson** é a mais simples do código, determina o objeto JSON que será exportado e o caminho de output, logo usa o método `JSON.stringify()` que converte o objeto para uma string formatada e utiliza a função `writeFileSync()` do módulo `fs` para criar um novo arquivo.

Para evitar bugs no código foi ativamente utilizado um tratamento de exceções com os blocos de `catch` e `try` para conseguir desviar possíveis problemas de leitura de arquivos e manipulação de dados.

## 2. Criar relatório de venda

Para a manipulação de dados, foi utilizado o SQLite Online, embora tenha enfrentado inicialmente alguns desafios de adaptação, uma vez que estou mais familiarizada com o PostgreSQL e diversos comandos via terminal não estão disponíveis na plataforma, como o `COPY` para carregar dados diretamente de arquivos JSON. No entanto, essa lacuna foi superada, pois ao importar os dois arquivos para a plataforma, as tabelas do banco de dados foram automaticamente geradas. Posteriormente, criei uma nova tabela consolidando as informações das colunas das tabelas originais e utilizei uma operação de `join` para combinar as duas chaves primárias das tabelas, representadas pelas PKs (id).

Em seguida, exportei os dados para um arquivo CSV e escolhi trabalhar com a criação de um arquivo [.ipynb](#), onde utilizei Python para extrair insights a partir dos dados brutos do novo banco de dados. O arquivo conta com a utilização das bibliotecas `Pandas` e `Matplotlib`, a `Pandas` para manipular os arquivos do CSV e a `Matplotlib` para criação de gráficos.