

Código em JavaScript

Esse código em JavaScript tem o propósito de corrigir problemas específicos em arquivos JSON que representam bases de dados. Ele realiza duas operações principais:

1. Correção de Caracteres Especiais:

- A função `fixCharacters` percorre os registros do arquivo JSON e substitui os caracteres especiais 'æ' por 'a' e 'ø' por 'o' nas propriedades nome e marca.

2. Correção de Valores Numéricos:

- A função `fixNumbers` percorre os registros do arquivo JSON e converte os valores da propriedade vendas de string para número. Antes da conversão, substitui possíveis vírgulas por pontos para garantir que a conversão seja feita corretamente.

3. Leitura e Exportação de Arquivos:

- A função `readJson` lê um arquivo JSON e o converte para um objeto JavaScript. Caso ocorra um erro na leitura ou na conversão, uma mensagem de erro é exibida.
- A função `exportFile` exporta um objeto JavaScript como JSON para um novo arquivo. Caso ocorra um erro na exportação, uma mensagem de erro é exibida.

4. Função Principal:

- A função `correctDb` é a função principal que chama as outras funções para corrigir os dados. Ela lê o arquivo corrompido, aplica as correções e exporta os dados corrigidos para um novo arquivo.

5. Uso do Código:

- O código é usado para corrigir dois arquivos JSON diferentes (`broken_database_1.json` e `broken_database_2.json`) e gerar novos arquivos corrigidos (`fixed_database.json` e `fixed_database_2.json`).

Código em SQL

Este código SQL cria uma tabela chamada `resultado_final` e popula essa tabela com informações resultantes de várias consultas relacionadas ao volume de vendas, receitas, médias, rentabilidade e veículos mais vendidos por marca. Vamos dar uma breve explicação para cada parte do código:

1. Questão 1: Contagem do Volume de Vendas por Marca
 - A primeira consulta calcula o total de vendas (TOTAL) para cada modelo (NOME/MODELO) agrupado por marca (B.c2). A tabela resultante inclui uma coluna adicional chamada RELATORIO com a descrição "Análise de Vendas por Marca".
2. Questão 2: Veículos com Maior Receita
 - A segunda consulta calcula o total de receitas (TOTAL) para cada modelo (MODELO) agrupado por marca (B.c2). A tabela resultante inclui uma coluna adicional chamada RELATORIO com a descrição "Top Veículos por Receita".
3. Questão 3: Veículos com Menor Receita
 - A terceira consulta é semelhante à segunda, mas agora busca os veículos com menor receita. A tabela resultante também inclui a coluna RELATORIO com a descrição "Veículos com Menor Receita".
4. Questão 4: Média de Vendas Anual por Marca
 - A quarta consulta calcula a média de vendas anuais (MEDIA) para cada marca (MARCA) ao longo do tempo (ANO). A tabela resultante inclui a coluna RELATORIO com a descrição "Média de Vendas por Ano".
5. Questão 5: Rentabilidade de Vendas por Marca
 - A quinta consulta calcula a rentabilidade de vendas (RESULTADO_DIVISAO) para cada marca (MARCA). A tabela resultante inclui a coluna RELATORIO com a descrição "Rentabilidade de Vendas".
6. Questão 6: Veículos Mais Vendidos por Marca
 - A sexta consulta calcula a quantidade total de veículos vendidos (`QUANTIDADE_DE_VEICULOS_VENDIDOS`) para cada modelo (`MODELO`) agrupado por marca (`B.c2`). A tabela resultante inclui a coluna `RELATORIO` com a descrição "Top Veículos por Quantidade Vendida e Marca".
7. Exibição do Relatório Final
 - Por fim, a última parte do código seleciona todos os registros da tabela `resultado_final` para exibição.

