Explicação dos Códigos

1. Código Javascript.

No início do código foi declarada a biblioteca ‘fs’ do node.js que foi fundamental para a exportação dos arquivos corrigidos, após sua declaração foi feita a chamada da função leitura() que faz a leitura e converte os arquivos JSON corrompidos em objetos, no final os retornando e atribuindo-os nas variáveis w e y.

Então é feita a chamada da função Correção_Nomes(w, y) que leva consigo as variáveis w e y contendo os dados corrompidos. Nesta função são feitos dois FOR um para cada variável, eles passam e corrigem, por meio de variáveis auxiliares, cada um dos nomes de veículos e marcas corrompidos, retornando no final os dados com nomes corrigidos para as variáveis w e y fora da função.

Depois é feita a chamada da função Correção_Vendas(w) que leva a variável w contendo os dados corrompidos das vendas, os corrigi e os devolve ao final para a variável w fora da função.

E no final é chamado a função Exportar(w, y), que leva as duas variáveis contendo os dois bancos já corrigidos em formato de objeto, as transforma em string (Json) e as exporta no formato JSON, utilizando a função fs.WriteFile() proveniente da biblioteca ‘fs’ declarada no início do código.

Na função Correção_Nomes(w, y) é importante apontar que como o array aux2 sempre recebia null em todo início do primeiro loop nos dois FOR principais, para impedimento de bugs foi adicionado um if com a condição (q == 0) em que dentro dele a letra era atribuída ao invés de incrementada no array, para que não somente que se a primeira letra fosse corrompida ela seria substituída pelo ‘A’ ou ‘O’ maiúsculo, mas também para que a letra substituísse o null na posição 0, e não fosse apenas incrementada no array, deixando o null e corrompendo o nome.

2. Código SQL.

Para o início, a partir do comando SELECT foram estabelecidas as cinco colunas que montam a tabela, depois com o comando FROM conectou-se por meio de um JOIN a segunda coluna da database_corrigida_1 e a primeira coluna da database_corrigida_2, juntando os dados dos dois arquivos e os estabelecendo como as fonte dos dados das colunas da tabela, e para finalizar, utilizando o GROUP BY foi estabelecido na última linha que a tabela fosse agrupada em marcas e veículos seguindo ordem alfabética.