

Explicação dos códigos

Javascript:

Função fixDatabaseName:

Recebe um banco de dados (json) como parâmetro e percorre todos os valores, verificando se a chave “nome” possui o caractere “æ” ou “ø” e caso essa verificação for satisfeita a função substitui os caracteres por “a” e “o” respectivamente.

A função também verifica se os caracteres “æ” ou “ø” aparecem na primeira posição do nome e caso a condição for verdadeira os caracteres serão substituídos por “A” e “O” (maiúsculo).

Após realizadas todas as verificações/substituições, a função retorna o banco de dados alterado.

Função fixDatabaseBrand:

Recebe um banco de dados (json) como parâmetro e percorre todos os valores, verificando se a chave “marca” possui o caractere “æ” ou “ø” e caso essa verificação for satisfeita a função substitui os caracteres por “a” e “o” respectivamente.

A função também verifica se os caracteres “æ” ou “ø” aparecem na primeira posição da marca e caso a condição for verdadeira os caracteres serão substituídos por “A” e “O” (maiúsculo).

Após realizadas todas as verificações/substituições, a função retorna o banco de dados alterado.

Função fixDatabaseSells:

Recebe um banco de dados (json) como parâmetro e percorre todos os valores, verificando se o valor da chave “vendas” é do tipo string. Caso essa verificação for satisfeita a função converte esse valor para o tipo number.

Após realizadas todas as verificações/substituições, a função retorna o banco de dados alterado.

Função fixDatabase:

Recebe um banco de dados (json) como parâmetro e verifica se ela possui ou não a chave “nome”. Caso o banco possua essa chave, a função “fixDatabaseName” e “fixDatabaseSells” serão chamadas, caso contrário a função chamada será “fixDatabaseBrand”.

Função createJSON:

Recebe dois parâmetros, o nome desejado do arquivo que será salvo e o conteúdo do arquivo (banco de dados) e cria um arquivo “.json” com o nome e conteúdo passados.

SQL:

Primeiro importei os dois bancos de dados corrigidos (fixed_database_1 e fixed_database_2) na plataforma SQL Online. Em seguida alterei o nome das colunas utilizando o código abaixo:

```
ALTER TABLE fixed_database_1 RENAME COLUMN c1 TO data;  
ALTER TABLE fixed_database_1 RENAME COLUMN c2 TO id_marca_;  
ALTER TABLE fixed_database_1 RENAME COLUMN c3 TO vendas;  
ALTER TABLE fixed_database_1 RENAME COLUMN c4 TO valor_do_veiculo;  
ALTER TABLE fixed_database_1 RENAME COLUMN c5 TO nome;  
ALTER TABLE fixed_database_2 RENAME COLUMN c1 TO id_marca;  
ALTER TABLE fixed_database_2 RENAME COLUMN c2 TO marca;
```

Após alterar os nomes das colunas verifiquei que precisaria de todas as informações das duas tabelas para responder as 5 perguntas e elaborar o relatório de vendas, por isso unifiquei as duas tabelas utilizando o INNER JOIN. Utilizei as colunas “id_marca_” (fixed_database_1) e “id_marca” (fixed_database_2) como chaves primária e estrangeira respectivamente para uni-las, como demonstrado no código abaixo:

```
SELECT * FROM fixed_database_1  
INNER JOIN fixed_database_2  
on fixed_database_1.id_marca_ = fixed_database_2.id_marca;
```

Logo após exportei a tabela unificada resultante no formato CSV para elaborar o relatório.

