

Código JavaScript.

```
// converter json para objeto js
fetch('./broken_database_1.json')
  .then(r => r.json())
  .then(json => {
    json.forEach(objeto_jason => {
      // Corrigir nome do carro
      objeto_jason.nome = objeto_jason.nome.replace(/æ/g, 'a').replace(/ø/g, 'o');

      // Corrigir valor das vendas
      if (typeof objeto_jason.vendas === 'string') {
        objeto_jason.vendas = parseInt(objeto_jason.vendas, 10);
      }
    });
  });
```

Utilizando o **fetch** é possível pegar um arquivo json externo e converter para um objeto js (nesse código é necessário que o arquivo json esteja no mesmo local que o código js).

Após converter utilizei o método **forEach** que percorre todos os elementos presentes num array um de cada vez e guarda temporariamente na variável **objeto_jason**, e executa os comandos, nesse caso irá usar o **replace** nos nomes substituindo as letras necessárias para corrigir os dados. Utilizei uma expressão regular para ser possível corrigir todas as letras necessárias, após isso utilizando um **if** comparei o tipo de dado armazenado para que quando fosse encontrado um do tipo **string** fosse corrigido para **int**.

```
// Converter objeto para string JSON  
const jsonString = JSON.stringify(json);  
  
// Escrever string JSON em um novo arquivo  
const blob = new Blob([jsonString], { type: 'application/json' });  
const url = URL.createObjectURL(blob);  
const a = document.createElement('a');  
a.href = url;  
a.download = 'database1_corrigido.json';  
a.click();
```

Após ter corrigido todos os dados, utilizando **JSON.stringify** o objeto js foi convertido para um arquivo json, e usando o objeto **blob** armazenei o arquivo convertido anteriormente, para ser possível criar um objeto URL com o **createObjectURL()**, depois cria um elemento html utilizando o **createElement (a)**, salva a url no elemento criado, nomeia o arquivo que será realizado o download e por fim simula um click no elemento para ser feito o download.