

Aqui, por exemplo, é como podemos usar para/de para fazer um loop através do Elementos de uma variedade de números e calcule sua soma:

```
Deixe dados = [1, 2, 3, 4, 5, 6, 7, 8, 9], soma = 0;
```

```
para (Let of Data of Data) {
```

```
soma += elemento;
```

```
}
```

```
soma // => 45
```

Superficialmente, a sintaxe parece regularmente para o loop: o para

A palavra -chave é seguida por parênteses que contêm detalhes sobre o que o

Loop deve fazer. Nesse caso, os parênteses contêm uma variável

declaração (ou, para variáveis ??que já foram declaradas, simplesmente

o nome da variável) seguido pela palavra -chave e um

expressão que avalia um objeto iterável, como a matriz de dados em

Este caso. Como em todos os loops, o corpo de um loop segue o

Parênteses, normalmente dentro de aparelhos encaracolados.

No código acabado de ser mostrado, o corpo do loop é executado uma vez para cada elemento do

matriz de dados. Antes de cada execução do corpo do loop, o próximo elemento

da matriz é atribuída à variável elemento. Os elementos da matriz são

iterado em ordem do primeiro ao último.

Matrizes são iterados "ao vivo" - as mudanças feitas durante a iteração podem

afetar o resultado da iteração. Se modificarmos o código anterior por

adicionar os dados da linha.push (soma); dentro do corpo do loop, então nós

criar um loop infinito porque a iteração nunca pode atingir o último

elemento da matriz.

Para/de com objetos