

UTF-16 como uma sequência (conhecida como "par substituta") de dois valores de 16 bits. Isso significa que um JavaScript String of Comprimento 2 (dois valores de 16 bits) pode representar apenas um único caractere unicode:

```
deixe euro = "?";
```

```
Let Love = "?";
```

Euro.length // => 1: Este personagem tem um elemento de 16 bits

Love.Length // => 2: UTF-16 A codificação de ? é "\ud83d\udc99"

A maioria dos métodos de manipulação de cordas definidos pelo JavaScript opera em valores de 16 bits, não caracteres.

Eles não tratam pares substitutos, especialmente, não realizam normalização da corda e nem mesmo

Verifique se uma string está bem formada UTF-16.

No ES6, no entanto, as cordas são iteráveis, e se você usar o loop ou ... operador com uma corda, ele iterará os caracteres reais da string, não os valores de 16 bits.

3.3.1 Literais de cordas

Para incluir uma string em um programa JavaScript, basta incluir o caracteres da string dentro de um par correspondente de solteiro ou duplo citações ou backticks ('ou "ou").

Backticks podem estar contidos em strings delimitados por uma única citação personagens, e da mesma forma para as cordas delimitadas por citações duplas e backticks. Aqui estão exemplos de literais de cordas:

```
"" // a sequência vazia: tem zero caracteres
```

```
'Teste'
```

```
"3.14"
```

```
'nome = "myform"'
```

```
"Você não prefere o livro de O'Reilly?"
```

```
"? é a proporção da circunferência de um círculo e seu raio"
```

```
`" Ela disse 'oi' ", ele disse.
```

Strings delimitadas com backsticks são uma característica do ES6 e permitir

Expressões JavaScript a serem incorporadas (ou interpoladas em)

String literal. Esta sintaxe de interpolação de expressão é coberta em §3.3.4.

As versões originais do JavaScript exigiam que os literais fossem escritos