

altere o valor de um número. Não é tão óbvio para as cordas, no entanto.

Como as cordas são como matrizes de personagens, você pode esperar ser capaz

Para alterar o caractere em qualquer índice especificado. De fato, JavaScript não

Permitir isso, e todos os métodos de string que parecem retornar uma string modificada

estão, de fato, retornando um novo valor de string. Por exemplo:

Seja `s = "olá";` // Comece com algum texto em minúsculas

`S.ToupPercase ();` // retorna "Hello", mas não altera `S`

`s` // => "Hello": a string original não tem

mudado

Os primitivos também são comparados pelo valor: dois valores são os mesmos apenas se

Eles têm o mesmo valor. Isso parece circular para números, booleanos,

nulo e indefinido: não há outra maneira de eles poder

comparado. Novamente, no entanto, não é tão óbvio para as cordas. Se dois

Os valores distintos de string são comparados, JavaScript os trata como igual se,

e somente se eles tiverem o mesmo comprimento e se o personagem em cada índice

é o mesmo.

Objetos são diferentes dos primitivos. Primeiro, eles são mutáveis ??- seu

Os valores podem mudar:

Seja `o = {x: 1};` // Comece com um objeto

`O.x = 2;` // sofre -o alterando o valor de um

propriedade

`O.Y = 3;` // sofre novamente adicionando um novo

propriedade

Seja `a = [1,2,3];` // matrizes também são mutáveis

`a [0] = 0;` // altere o valor de um elemento de matriz

`a [3] = 4;` // Adicione um novo elemento de matriz

Os objetos não são comparados pelo valor: dois objetos distintos não são iguais