

o oposto: permite que você não inclua declarações onde alguém está esperando. A declaração vazia é assim:

```
;
```

O intérprete JavaScript não toma medidas quando executa um vazio declaração. A declaração vazia é ocasionalmente útil quando você quiser

Para criar um loop que tenha um corpo vazio. Considere o seguinte para

O loop (para loops será coberto no §5.4.3):

```
// inicialize uma matriz a
```

```
para (vamos i = 0; i < a.Length; a[i++] = 0);
```

Neste loop, todo o trabalho é feito pela expressão `a[i++] = 0` e

Nenhum corpo de loop é necessário. A sintaxe JavaScript requer uma declaração como um corpo de loop, no entanto, uma declaração vazia - apenas um semicolon - é usado.

Observe que a inclusão acidental de um ponto de vírgula após o direito parênteses de um loop `for`, enquanto o loop, ou se a declaração pode causar

Bugs frustrantes que são difíceis de detectar. Por exemplo, o seguinte

O código provavelmente não faz o que o autor pretendia:

```
if ((a === 0) || (b === 0)); // opa! Esta linha faz
```

nada...

```
o = nulo; // e esta linha é sempre
```

executado.

Quando você usa intencionalmente a declaração vazia, é uma boa ideia

Comente seu código de uma maneira que deixa claro que você está fazendo isso propósito. Por exemplo: