

uma operação $O(n)$. Isso significa que se você se encontrar usando ... dentro de um loop ou função recursiva como uma maneira de acumular dados em Um objeto grande, você pode estar escrevendo um algoritmo $O(n)$ ineficiente que não será escalonado bem como n cresce.

6.10.5 Métodos de abreviação

Quando uma função é definida como uma propriedade de um objeto, chamamos de funcione um método (teremos muito mais a dizer sobre os métodos em Capítulos 8 e 9). Antes do ES6, você definiria um método em um objeto literal usando uma expressão de definição de função como faria Defina qualquer outra propriedade de um objeto:

```
Deixe Square = {  
  área: function () {return this.side * this.side;},  
  lado: 10  
};  
square.area () // => 100
```

No ES6, no entanto, a sintaxe literal do objeto (e também a definição de classe Sintaxe que veremos no capítulo 9) foi estendida para permitir um atalho onde a palavra -chave da função e o cólon são omitidos, resultando em código como este:

```
Deixe Square = {  
  área () {return this.side * this.side;},  
  lado: 10  
};  
square.area () // => 100
```

Ambas as formas do código são equivalentes: ambos adicionam uma propriedade nomeada área para o objeto literal e ambos definem o valor dessa propriedade para o