

Processo de Stringification. A documentação completa para essas funções é No §11.6.

## 6.9 Métodos de objeto

Como discutido anteriormente, todos os objetos JavaScript (exceto aqueles explicitamente criado sem um protótipo) herdar propriedades de `Object.prototype`. Essas propriedades herdadas são principalmente métodos, e porque eles estão disponíveis universalmente, eles são de Interesse particular para programadores JavaScript. Nós já vimos o Métodos `HASOWNPROPERTY ()` e `PROPRIEDISENUMELE ()`, por exemplo. (E também já cobrimos alguns estáticos funções definidas no construtor de objeto, como `Object.Create ()` e `Object.Keys ()`.) Esta seção explica um punhado de métodos de objetos universais que são definidos em `Object.prototype`, mas que devem ser substituídos por Outras implementações mais especializadas. Nas seções a seguir, Mostramos exemplos de definição desses métodos em um único objeto. Em Capítulo 9, você aprenderá como definir esses métodos de maneira mais geral para uma classe inteira de objetos.

### 6.9.1 O método `ToString ()`

O método `tostring ()` não leva argumentos; ele retorna uma string que De alguma forma, representa o valor do objeto em que ele é chamado. JavaScript invoca esse método de um objeto sempre que precisar converta o objeto em uma string. Isso ocorre, por exemplo, quando você usa O operador `+` para concatenar uma string com um objeto ou quando você passa um objeto para um método que espera uma string.