

Se você deseja criar um objeto vazio comum (como o objeto retornado por {} ou novo objeto ()), aprove o `Object.prototype`:

Seja `o3 = Object.create (Object.prototype);` // `o3` é como {} ou novo objeto {}.

A capacidade de criar um novo objeto com um protótipo arbitrário é um poderoso, e usaremos `Object.create ()` em vários lugares ao longo deste capítulo. (`Object.create ()` também leva um segundo argumento opcional que descreve as propriedades do novo objeto. Este segundo argumento é um recurso avançado coberto no §14.1.)

Um uso para `Object.create ()` é quando você deseja se proteger Modificação não intencional (mas não maliciosa) de um objeto por uma biblioteca função que você não tem controle. Em vez de passar o objeto Diretamente para a função, você pode passar um objeto que herda dela. Se A função lê propriedades desse objeto, verá o herdado valores. Se definir propriedades, no entanto, essas gravações não afetarão o objeto original.

Seja `o = {x: "Não altere este valor"};`

`Library.function (Object.create (o));` // se protege contra modificações acidentais

Para entender por que isso funciona, você precisa saber como são as propriedades consultado e definido em JavaScript. Estes são os tópicos da próxima seção.

6.3 Propriedades de consulta e definição

Para obter o valor de uma propriedade, use o ponto (.) Ou suporte quadrado