

```
padrão: // Se tudo mais falhar ...  
// Executar o bloco de código #4.  
quebrar; // Pare aqui  
}
```

Observe a palavra -chave quebrada usada no final de cada caso neste código.

A declaração de quebra, descrita mais adiante neste capítulo, causa o intérprete para pular até o fim (ou "quebrar") do interruptor declaração e continue com a declaração que a segue. O caso cláusulas em uma declaração de interruptor especificam apenas o ponto de partida do código desejado; Eles não especificam nenhum ponto final. Na ausência de Declarações de quebra, uma declaração de interruptor começa a executar seu bloco de código na etiqueta do caso que corresponde ao valor de sua expressão e continua executando declarações até chegar ao final do bloco. Sobre ocasiões raras, é útil escrever um código como esse que "cai" de uma etiqueta de caso para a seguinte, mas 99% do tempo você deve ser Cuidado para encerrar todos os casos com uma declaração de quebra. (Ao usar Mudar dentro de uma função, no entanto, você pode usar uma declaração de retorno em vez de uma declaração de quebra. Ambos servem para encerrar o interruptor declaração e impedir a execução de cair para o próximo caso.)

Aqui está um exemplo mais realista da instrução Switch; ele converte Um valor para uma string de uma maneira que depende do tipo de valor:

```
função convert (x) {  
  Switch (tipo de x) {  
    caso "número": // converte o número para um  
    Inteiro hexadecimal  
    retornar x.toString (16);  
    case "string": // retorna a string fechada  
    em citações
```