

```
Catch (ex) { // Se a entrada do usuário não foi válida, terminamos  
aqui em cima  
alerta (ex); // Diga ao usuário qual é o erro  
}
```

Este exemplo é uma declaração de tentativa/captura sem cláusula finalmente. Embora finalmente não seja usado com a mesma frequência que pode ser útil. No entanto, seu comportamento requer explicação adicional. O finalmente é garantido que a cláusula seja executada se alguma parte do bloco de tentativa for executado, independentemente de como o código no bloco de tentativa é concluído. Isso é Geralmente usado para limpar após o código na cláusula de tentativa. No caso normal, o intérprete JavaScript chega ao final do tente blocos e depois prossegue para o bloco finalmente, que executa qualquer limpeza necessária. Se o intérprete deixou o bloco de tentativa por causa de uma declaração de retorno, continuação ou interrupção, o bloco finalmente é Executado antes que o intérprete salte para seu novo destino. Se ocorrer uma exceção no bloco de tentativa e há um associado Catch bloqueio para lidar com a exceção, o intérprete executa primeiro o Pegue o bloco e depois o bloco finalmente. Se não houver captura local bloco para lidar com a exceção, o intérprete executa primeiro o Finalmente bloqueie e depois pula para a captura mais próxima que contém cláusula. Se um bloco finalmente causar um salto com um retorno, continue, quebrar, ou fazer uma declaração, ou chamar um método que lança um exceção, o intérprete abandona qualquer salto pendente e Executa o novo salto. Por exemplo, se uma cláusula finalmente lança um