

eles também. Um objeto de erro tem uma propriedade de nome que especifica o tipo de erro e uma propriedade de mensagem que mantém a string passada para a função do construtor. Aqui está uma função de exemplo que lança um Objeto de erro quando invocado com um argumento inválido:

```
função fatorial (x) {  
  // Se o argumento de entrada for inválido, faça uma exceção!  
  se (x < 0) lançar um novo erro ("x não deve ser negativo");  
  // caso contrário, calcule um valor e retorne normalmente  
  deixe f;  
  for (f = 1; x > 1; f *= x, x--)/ *vazio * /;  
  retornar f;  
}
```

Fatorial (4) // => 24

Quando uma exceção é lançada, o intérprete JavaScript imediatamente interrompe a execução normal do programa e salta para a exceção mais próxima manipulador. Os manipuladores de exceção são escritos usando a cláusula de captura do Tente/Catch/Finalmente declaração, descrita no próximo seção. Se o bloco de código em que a exceção foi lançada não tem uma cláusula de captura associada, o intérprete verifica o próximo Bloco de código mais alto para ver se ele tem um manipulador de exceção associado a ele. Isso continua até que um manipulador seja encontrado. Se um a exceção é lançada em uma função que não contém um Tente/Catch/Finalmente declaração para lidar com isso, a exceção propaga -se para o código que chamou a função. Desta maneira, Exceções se propagam através da estrutura lexical do JavaScript Métodos e subir a pilha de chamadas. Se nenhum manipulador de exceção for encontrado, A exceção é tratada como um erro e é relatada ao usuário.

5.5.7 Tente/Catch/Finalmente