区别一：throw 是语句抛出一个异常；throws 是方法抛出一个异常；  
  
throw语法：throw <异常对象>  
  
在方法声明中，添加throws子句表示该方法将抛出异常。如果一个方法会有异常，但你并不想处理这个异常，就在方法名后面用throws,这样这个异常就会抛出，谁调用了这个方法谁就要处理这个异常，或者继续抛出.  
  
throws语法：[<修饰符>]<返回值类型><方法名>（[<参数列表>]）[throws<异常类>]  
  
其中：异常类可以声明多个，用逗号分割。  
  
区别二：throw要么和try-catch-finally语句配套使用，要么与throws配套使用。但throws可以单独使用，然后再由处理异常的方法捕获。

联系一：try .... catch 就是用catch 捕获 try 中的异常，并处理；catch是捕获异常。也就是说某一个操作有可能会抛出什么异常。throw 就是不处理异常，直接抛出异常

调用一个会throws exception的方法（在方法定义的时候可以用throws实现）时，需要把这个方法放在try里，然后用catch破获这个exception，做相应的处理。

throw new exception（）是抛出一个exception，由别的method来破获它。

也就是说try....catch是为破获别人的exception用的，而throw是自己抛出exception让别人去破获的。

public void input() throws Exception

{ }

public void outputException()

{

   try {

       input();

}

catch(Exception e)

{

System.out.println("Exception");

}

}

简单的说，如果一个方法中用了throws，那么这个方法会向上抛出一个异常，那么在调用这个方法的时候必须把这个调用放在try...catch块里处理这个异常。

对于异常有两种处理方法

1、要么声明异常，也就是在方法名后面加上throws exceptionname,..., 方法本身只是抛出异常，由函数调用者来捕获异常。若产生异常，异常会沿着调用栈下移，一直找到与之匹配的处理方法，若到达调用栈底仍未找到，程序终止。

2、要么捕获异常。通过try catch方法，catch子句中放置处理异常的语句。

联系二：对于会觉得会有异常抛出的程序块，用try{}包住，然后用catch来抓住这个异常，在catch中对异常做处理， 在try中如果有异常的话，程序会转到catch而不会中断，通常这两个是配合使用的,如果你不想因为程序有错，而抛出一大堆异常的话，你就把该程序try起来，try和catch只能获取程序运行时引发的异常，而throw语句可以引发明确的异常，程序到了throw语句这后就立即停止，不会执行后面的程序,

大家也许在编程的过程中，会意识到这里出错，但不知道会抛出什么异常，也是一件头疼的事，现将这部分总结如下：  
  
算术异常类：ArithmeticExecption  
空指针异常类：NullPointerException  
类型强制转换异常：ClassCastException  
数组负下标异常：NegativeArrayException  
数组下标越界异常：ArrayIndexOutOfBoundsException  
违背安全原则异常：SecturityException  
文件已结束异常：EOFException  
文件未找到异常：FileNotFoundException  
字符串转换为数字异常：NumberFormatException  
操作数据库异常：SQLException  
输入输出异常：IOException  
方法未找到异常：NoSuchMethodException

、throw关键字通常用在方法体中，并且抛出一个异常对象。程序在执行到throw语句时立即停止，它后面的语句都不执行。通过throw抛出异常后，如果想在上一级代码中来捕获并处理异常，则需要在抛出异常的方法中使用throws关键字在方法声明中指明要跑出的异常；如果要捕捉throw抛出的异常，则必须使用try—catch语句

class MyException extends Exception { //创建自定义异常类 String message; //定义String类型变量 public MyException(String ErrorMessagr) { //父类方法 message = ErrorMessagr; } public String getMessage(){ //覆盖getMessage()方法 return message; }}public class Captor { //创建类static int quotient(int x,int y) throws MyException{//定义方法抛出异常if(y < 0){ //判断参数是否小于0 throw new MyException("除数不能是负数");//异常信息 } return x/y;//返回值 }public static void main(String args[]){ //主方法 try{ //try语句包含可能发生异常的语句 int result = quotient(3,-1);//调用方法quotient() }catch (MyException e) { //处理自定义异常 System.out.println(e.getMessage()); //输出异常信息 } catch (ArithmeticException e) { //处理ArithmeticException异常 System.out.println("除数不能为0");//输出提示信息 } catch (Exception e) { //处理其他异常 System.out.println("程序发生了其他的异常"); //输出提示信息 } }}

throws是用来声明一个方法可能抛出的所有异常信息throw则是指抛出的一个具体的异常类型。通常在一个方法（类）的声明处通过throws声明方法（类）可能抛出的异常信息，而在方法（类）内部通过throw声明一个具体的异常信息。throws通常不用显示的捕获异常，可由系统自动将所有捕获的异常信息抛给上级方法；throw则需要用户自己捕获相关的异常，而后在对其进行相关包装，最后在将包装后的异常信息抛