1. 什么是异常？

答：有异于常态，和正常情况不一样，有错误出现。阻止当前方法或作用域，称之为异常。异常具有不确定性。如：用户输入错误，除数为0，需要的文件不存在或打不开，数组下标越界，传入参数为空或不符合指定范围等。异常处理机制是JAVA语言的重要特征之一。通过异常处理机制可防止程序执行期间因出现错误而造成不可预料的结果。

1. 什么是抛出异常？

答：抛出异常有两种方式：由系统自动抛出和利用语句抛出。

1. 由系统自动抛出异常：

在程序运行中，如果出现了可被Java运行系统识别的错误，系统会自动产生与该错误相对应的异常类的对象，即自动抛出。

1. 人为异常抛出：

两种方式：

① 在方式头写出需要抛出的异常（利用throws语句）

② 在方法体内抛出异常

1. 捕捉异常

答：在没有给出处理异常的代码的程序中，抛出的异常都被Java运行系统捕捉，由Java运行系统进行相应处理。一般来说，在设计程序过程中，如果能够预测程序中可能发生的异常，则应在程序中给出处理的代码，而不交给Java运行系统处理。

要由程序自己捕捉和处理异常，需要建立try-catch-finally语句块。

1. Throw语句和Throws语句的区别
2. throw是语句抛出一个异常（明确了这个地方要抛出一个异常）

语法：throw(异常对象)； 如 throw e;

1. throws是方法可能抛出异常的声明

（用在声明方法的时，表示该方法可能要抛出异常）

语法：[(修饰符)](返回值类型)(方法名)([参数列表])[throws(异常类)]{......}

public void doA(int a) throws Exception1,Exception3{......}

1. Try-catch-finally语句执行过程

try{

//在此区域内或能发生异常;}

catch(异常类1 e1)

{//处理异常1;}

...

catch(异常类n en)

{//处理异常类n;}

finally

{//不论异常是否发生，都要执行的部分;}