**Exercise 1：**

**import java.util.InputMismatchException;  
import java.util.Scanner;  
  
public class Exercise1 {  
 public static void main(String[] args) {  
 boolean determine=false;  
 Scanner input=new Scanner(System.in);  
 do{  
 try {  
 determine=false;  
 System.out.print("Please input two integers: ");  
 int a=input.nextInt();//如果输入不是int，则这一行的读取无效  
 int b=input.nextInt();  
 System.out.println(a + " + " + b + " = " + (a + b));  
 } catch (InputMismatchException e) {  
 String c= input.nextLine();//一定要消除之前用户输入的两个字符  
 System.out.println("Try again.(Incorrect input: an integer is required)");  
 e.printStackTrace();  
 determine=true;  
 }  
 }while (determine);  
 }  
}**



**Exercise 2：**

**import java.util.Scanner;  
  
public class CommonExceptionDemo {  
 enum ExceptionEnum {  
 ARITHMETIC(new ArithmeticException("/ by zero")),  
 INDEXOUTOFBOUNDS(new IndexOutOfBoundsException()),  
 NEGATIVEARRAYSIZE(new NegativeArraySizeException()),  
 NULLPOINTER(new NullPointerException()),  
 NUMBERFORMAT(new NumberFormatException()),  
 CLASSCAST(new ClassCastException());  
  
 private Exception e;  
  
 ExceptionEnum(Exception e){  
 this.e=e;  
 }  
  
 public Exception getException() {  
 return e;  
 }  
  
 public static ExceptionEnum getExceptionByOrdinal(int ord) throws IllegalArgumentException {  
 ExceptionEnum e;  
 switch (ord) {  
 case 0:  
 e= ExceptionEnum.ARITHMETIC;return e;  
 case 1:  
 e= ExceptionEnum.INDEXOUTOFBOUNDS;return e;  
 case 2:  
 e= ExceptionEnum.NEGATIVEARRAYSIZE;return e;  
 case 3:  
 e= ExceptionEnum.NULLPOINTER;return e;  
 case 4:  
 e= ExceptionEnum.NUMBERFORMAT;return e;  
 case 5:  
 e= ExceptionEnum.CLASSCAST;return e;  
 default:throw new IllegalArgumentException();//如果有case抛出异常，抛出异常就会终止程序，那么这里就不能也不需要再return（否则会出现unreachable statement）  
 }  
 }  
 }  
  
 public static void main(String[] args) {  
 System.out.println("Exception:");  
 for(ExceptionEnum c:ExceptionEnum.values()) {  
 System.out.println(" "+c.name() +"("+c.ordinal()+")");//九个空格  
 }  
  
 System.out.print("Please input an integer to select the TYPE OF EXCEPTION(0~5): ");  
 Scanner input=new Scanner(System.in);  
 try {  
 int a = input.nextInt();  
 throw ExceptionEnum.getExceptionByOrdinal(a).getException();  
 }catch (ArithmeticException e0){  
 e0.printStackTrace();  
 }catch (Exception e){  
 e.printStackTrace();  
 System.out.println("Here is the end.");  
 }  
 }  
  
}**





