1. Java有哪些基本数据类型？描述其分类情况，编写一个简单的程序，各声明一个变量，初始化并输出其值。

2. 比较break与continue语句的区别。

3. 资料查阅题：比较String与StringBuffer类型的主要区别。

4. Java数组的特点是什么？如何创建和使用对象数组？

5. 编写一个程序，利用除法运算和取余运算把整型变量的每位数字都提出来并打印。例如n值为1678，则输出结果为：n ＝ 1，6，7，8； n值为39，则输出结果为：n ＝ 3，9。

6. 编写Java程序，接受用户输入的1~12之间的整数，若不符合条件则重输入，利用switch语句输出对应月份的天数。

7. 编写Java程序计算一个整数之内的素数并输出。

8. 编写Java程序实现：输入一组整数存放在数组中，比较并输出其中的最大值和最小值；再将数组元素从小到大排序并输出。

抱歉，我没有在任何一个代码中throw异常

1. 代码形式提交
2. break是完全结束当前循环，continue是终止本次循环。
3. String 是不可变的对象, 因此在每次对String类型的对象进行改变的时候其实都等同于生成了一个新的 String 对象，然后将指针指向新的 String 对象

而StringBuffer就是对象本身进行操作，而不是像前者一样通过生成新的对象，再改变对象引用的间接操作

由于String的这种特性，在定义经常更改的字符串时，我们应使用StringBuffer类型，以避免内存的浪费，而在5.0还新增了StringBuilder类型，我感觉还是要优先用StringBuilder型

1. Java数组最大的特点是，它是一种类（至少我是这样认为……）,工作原理和其他语言不一样（swift也是这样），其他特性和其他语言大同小异，比如下标从零开始，声明之后长度不可变等。

声明方面，我们使用dataType dataName[]或者dataType[] dataName声明数组变量，然后new生成数组

使用方法就很多了,最简单的array.length可以查看数组元素的个数，array[i]可以访问数组的成员等，如果数组的成员不是基本数据类型，而是一种其他类型，那它就是对象数组了，可以用例如：student[3].id的方式访问其元素的成员，其他在声明和使用时和基本数据类型的数组大同小异

1. 代码形式提交
2. 代码形式提交
3. 代码形式提交
4. 代码形式提交