StringBuilder概述：内部拥有一个数组用来存放字符串内容，进行字符串拼接时，直接在数组中加入新内容。StringBuilder会自动维护数组的扩容。

Java中的StringBuffer类是动态字符串数组。

StringBuffer类包含append（）方法，append（）方法相当于“+”，将指定的字符串追加到此字符序列。

StringBuffer.append()追加的字符串在同一个内存地址。

public class Test{

public static void main(String args[]){

StringBuffer sBuffer = new StringBuffer("aaa");

sBuffer.append("bbb");

sBuffer.append("ccc");

sBuffer.append("ddd");

System.out.println(sBuffer);

}

}

运行结果：aaabbbcccddd

substring()方法在截取字符串子串时，含头不含尾

next()和nextLine()的区别

next():1.一定要读到有效字符才可以结束

2.对输入之前遇到的空白，会自动去掉

3.只有输入有效字符后才可以将后面的空白作为分隔符或者结束符

4.不能得到带有空格的字符串

nextLine():以Enter为结束符，返回Enter前的所有字符，包括空白

字符比较A.equals(B)

contains() 方法用于判断字符串中是否包含指定的字符或字符串。

返回True,False

Arrays.Sort(“”)排序字符

toCharArray —— 将字符串中的字符转换为字符数组

字符串格式化

本地语言环境：String.format(String format, Object... args)

本地语言环境：String.format(Locale locale, String format, Object... args)







Java中的math.max()方法返回指定参数中的最大值

比较（相同类型）：compareTo() 方法

Interface —— 申明接口

进制转换：Integer.toString(参数,进制) \_ 字符类型String



String转换为int的方法有：Integer.parseInt(String)方法、Integer.valueOf(String)方法。

parseInt()是Integer包装类里的一个方法，可以将字符串解析为带符号的整数。

Integer.valueOf()和Integer.parseInt()之间还是存在差异的：valueOf(String)方法会返回Integer类的对象，而parseInt(String)方法返回原始的int值。

注：使用parseInt()方法时，字符串中的所有字符都必须是数字，但第一个字符可以是减号“ - ”。例：String str="-1234";

Enum(枚举):枚举的类型需要定义在类之外

Java 枚举是一个特殊的类，一般表示一组常量，比如一年的 4 个季节，一个年的 12 个月份，一个星期的 7 天，方向有东南西北等。

Java 枚举类使用 enum 关键字来定义，各个常量使用逗号来分割。