字符串最大的特点：一但初始化，不可更改 在java中 String类由Final修饰

Final修饰的类不可被继承，且类中成员默认为final修饰

Final修饰的方法，该方法不能被覆盖

Final修饰变量：修饰基本数据类型，该数据初始化后不可被更改，如果是引用类型变量，对象初始化为不能让其指向另一个对象

String 类代表字符串。Java 程序中的所有字符串字面值（如 "abc" ）都作为此类的实例实现。

{

String S1=”sdad”; //S1代表一个对象变量，”sdad”为一个String实例对象，为一个常量

S1=”aaaa”;

Sysytem.out.println(“S1:”+S1);

}

运行结果：S1:aaaa “sdad”和”aaaa”是不可更改的 ，此处改变的是S1的指向，从字符串常量1指向字符串常量2

S1为一个String类型的类变量

转义字符： 注意：\数字 (\141) 斜线加数字，

代表的该数字所表示的ASCII码值对应的字符

String a="\\141\141abc\t";

该字符串长度为多少？

首先我们来分析一下，\\是一个，1，4，1是3个，已经4个了，\141是一个，5个了，a,b,c是3个，总共8个了，最后\t是一个，总共9个

该怎么做？首先要特别注意\这个转义字符，后面跟着的和它组成一个字符，在上面这道题上最主要的是\141很有迷惑性，这'\141'表示的是字符'a'，它是以3位8进驻的ascii码表示的，所以以后做题时要注意\后面的字符和3位数字，其实他们表示的是一个字符

String s1=”abc” 和String s1=new String(“abc”)区别：

前者S1在内存中只有一个对象”abc”

后者S2在内存中有两个对象，”abc”和String(“abc”)这两个对象

S1==S2和S1.equals(S2)的区别：

String类没有增删操作，但是可以改查拆分替换等 操作。

StringBuffer和StringBuilder可以追加和插入操作，用于生成字符串

StringBuffer和StringBuilder区别：

StringBuilder不同步，效率高。 StringBuffer同步，效率低

基本数据类型包装类：

byte Byte

int Integer

short short

float Float

long Long

double Double

boolean Boolean

char Character

基本数据类型对象包装类的最常见作用，

就是用于基本数据类型和字符串类型之间做转换

基本数据类型转成字符串。

基本数据类型+""

基本数据类型.toString(基本数据类型值);

如： Integer.toString(34);//将34整数变成"34";

字符串转成基本数据类型。

xxx a = Xxx.parseXxx(String);

int a = Integer.parseInt("123");

double b = Double.parseDouble("12.23");

boolean b = Boolean.parseBoolean("true");

部分基本数据类型 十进制转成其他进制。

toBinaryString();

toHexString();

toOctalString();

其他进制转成十进制。

parseInt(string,radix);