

数组

JAVA中数组有着自己的特色

数组是由同一类型的对象或者基本数据构成，并且封装在同一个标识符下。JAVA中，数组是**对象**，是动态初始化。这个内容的具体细节在后面会给出。

数组元素

数组中的变量被称为数组的元素，数组的元素没有名字，通过数组的名字和非负整数下标值引用数组元素。

创建和使用数组

JAVA中数组的创建和使用分为以下四步：

1. 数组的声明
2. 数组的创建
3. 数组元素的初始化
4. 使用数组

JAVA的字符串String是一串Unicode字符。JAVA并没有内置的String数据类型，正如前面所说，JAVA中String数据类型是通过类来实现的。数组通过如下代码声明和创建：

```
String e = "";  
String greeting = "Hello";
```

Substrings

Substring方法是String类中的一个方法，它可以提取原字符串中第一个子串：

```
String greeting = "Hello";  
String s = greeting.substring(0, 3);
```

最后的结果为：

```
"Hel"
```

substring方法中有两个参数，第一个参数是想要复制的子串在父串中的开始位置，第二个参数是其结束位置。

Concatenation

JAVA和很多高级语言一样，允许使用"+"来连接两个字符串：

```
String expletive = "Expletive";
String PG13 = "deleted";
String message = expletive + PG13;
```

最后得到的结果是

```
"Expletivedelete"
```

另外，String字符串和别的类型的数据也可以通过"+"来进行连接:

```
int age = 13;
String rating = "PG" + age
```

如果要构造一个按字母分开的字符串，可以使用join方法:

```
String all = String.join(",", "S", "M", "L", "XL");
System.out.println(all);
```

最后得到的结果是:

```
"S,M,L,XL"
```

String的不可变性

JAVA中String不提供任何更改已经存在的字符串的方法，但是可以通过重新赋值的方法来对字符串进行改动:

```
String greeting = "Hello";
greeting = greeting.substring(0, 3) + "p!";
```

由于JAVA中String中任何一个字符都不可以改变，因此在文件中，称String对象有着不可变性(Immutable)。

测试String是否相等

JAVA中String类提供equals方法来判断两个字符串是否相等:

```
String A = "HaHaHa";
String B = "HaHaHa";
```

```
String C = "HaHa";

System.out.println(A.equals(B));
System.out.println(A.equals(C));
System.out.println("HaHaHa".equals(A));
```

最后结果为:

```
true
false
true
```

和C系列一样，不能通过"=="来判断字符串是否相等。在JAVA中，"=="只能用来判断两个字符串是不是在同一个位置上(Same Location)。