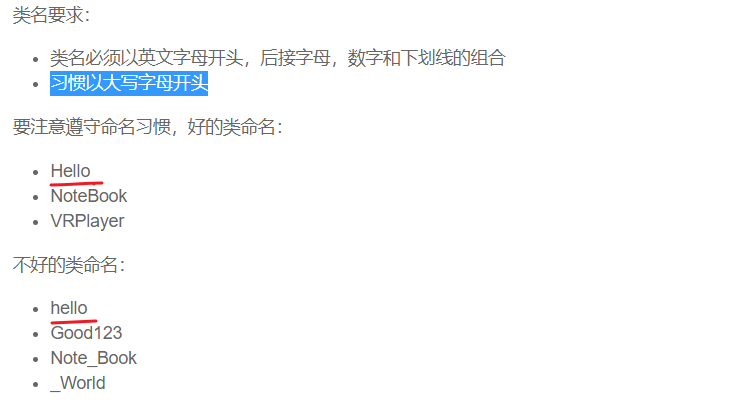
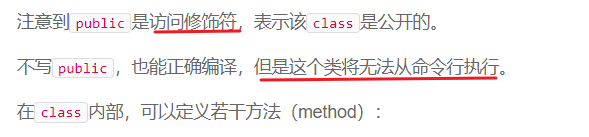
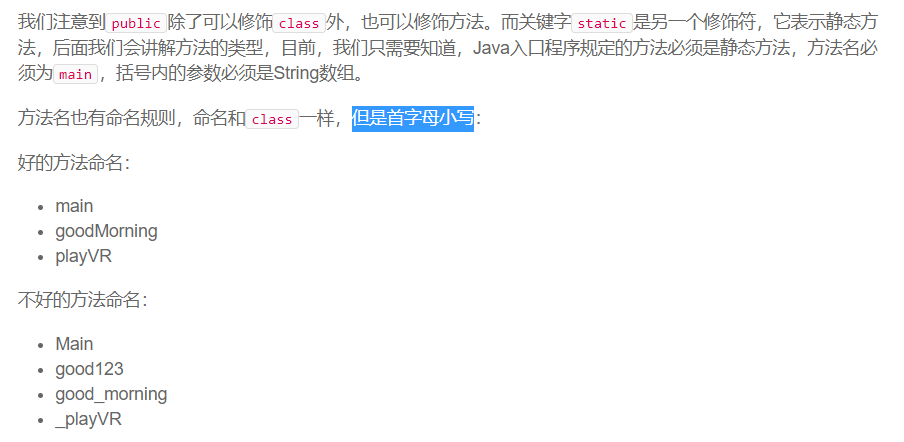
# 变量有关





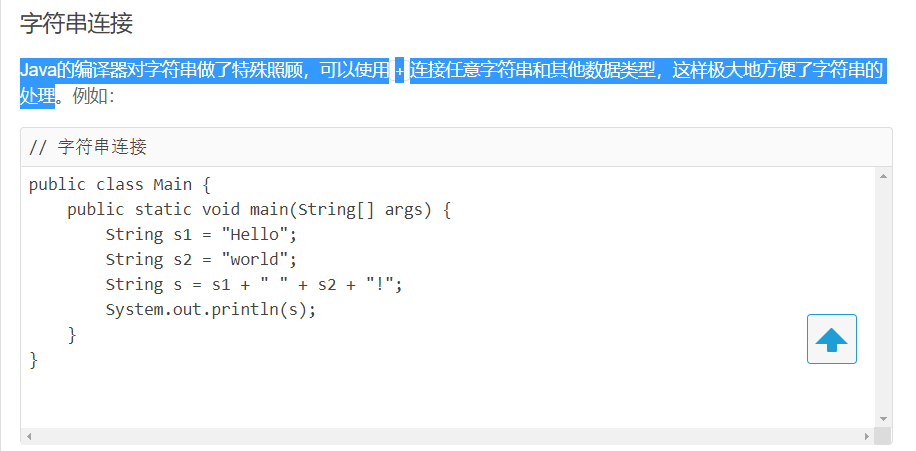




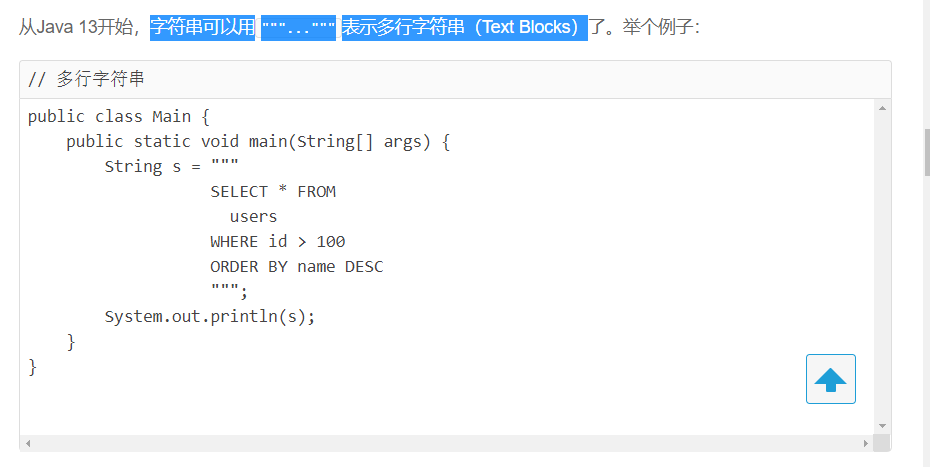
Java的final关键字就是C中的const

就是一个字符串数组的变量名，不是关键字args是arguments的缩写默认名，一般都习惯性照写你可以改成其他名。、

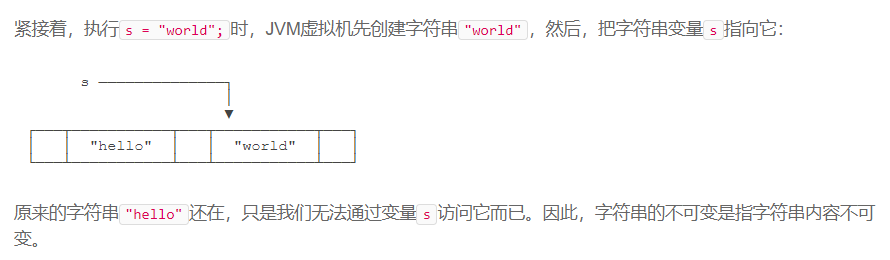
## 1.2 字符串



Ps：=为赋值，-可能也有去子串的功能



### 字符串不可变性解释—字符串不变。仅仅是变量的指向变了



## 数组

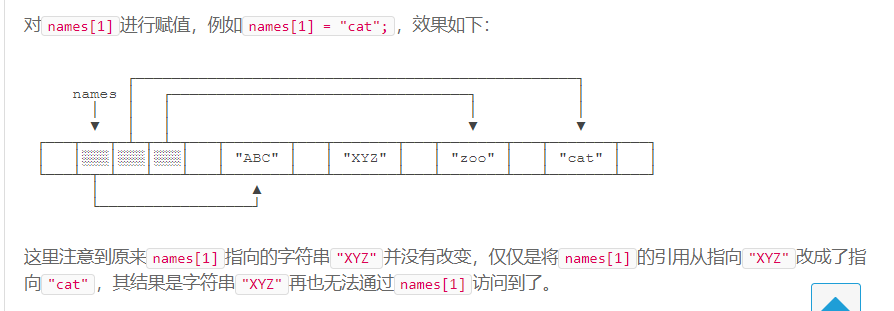
int [] array = new int[5];//万物皆对象，我喜欢，就不用比照两套了

int[] ns = new int[] { 68, 79, 91, 85, 62 };//编译器自动推算数组大小

int[] ns = { 68, 79, 91, 85, 62 };

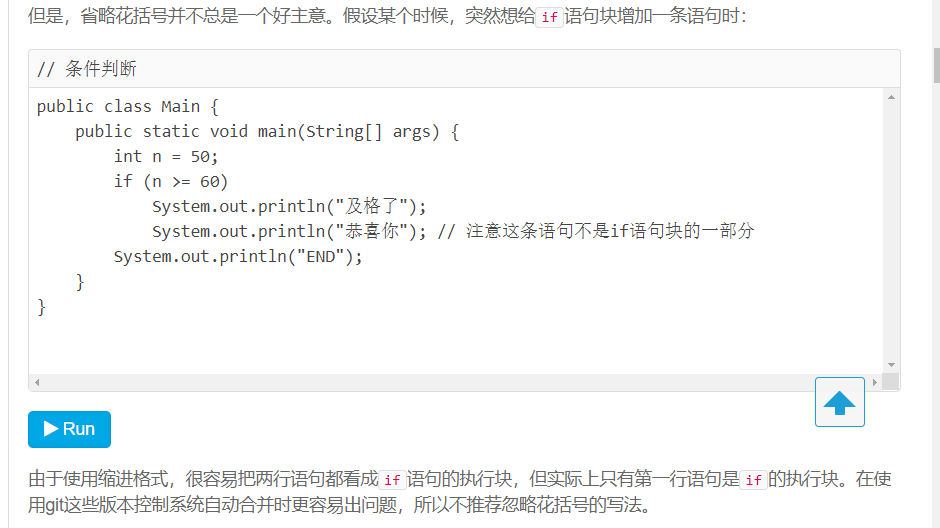
System.out.println(ns.length); // 5

字符串不变。仅仅是变量的指向变了

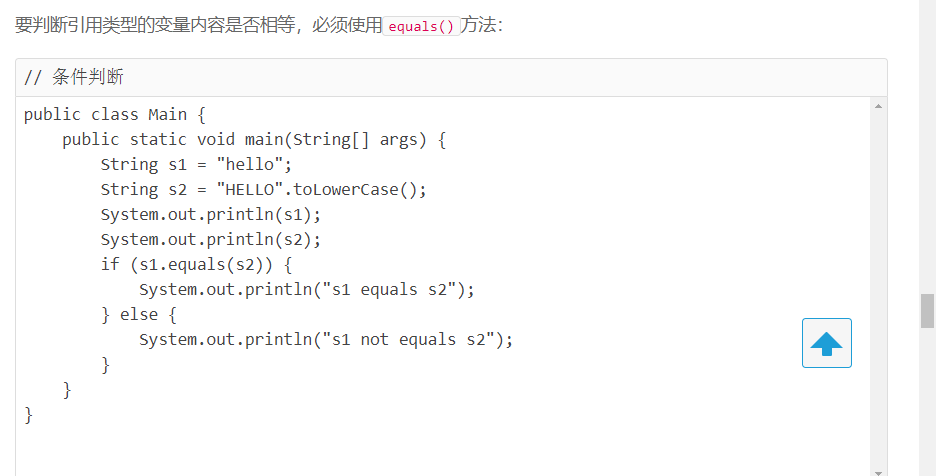


# 2 流程控制

省略花括号并不是好主意，建议单行也写花括号（ps：java不使用缩进表示逻辑，python才，j和c缩进是让代码块看起来思路清晰）



**字符串方法**



Switch复选，然后可以在case里面进行跳转

例如：

option = 输入选择 //好像只有int后面可以看看有没有string的

switch（option）{

case 1:

功能接口，例如开始；

break;

case 2:

功能接口，例如设置；

break;

...

default://其他所有情况 ，缺失，就是没有符合选择的意思

System.out.println("Not selected");

break;

}

有的！有的！有string

public class Main {

public static void main(String[] args) {

String fruit = "apple";

switch (fruit) {

case "apple"://也就是case+参数可能值

System.out.println("Selected apple");

break;

case "pear":

System.out.println("Selected pear");

break;

case "mango":

System.out.println("Selected mango");

break;

default:

System.out.println("No fruit selected");

break;

}

}

}



**Java12后的->引入就相当于把靠缩进形成的类python语句的代码逻辑块变成了用{}形成的代码逻辑块；返回值就是直接int value = switch（option）{…}的结构。**

**while（1）{…}怎么跳出：**

用break语句，首先这条语句是无条件跳出循环的。不过你可以设置一个条件，条件满足时，用这break语句，就出来了（**while里面写一个if条件，满足条件break出来**）

好像还有goto;return;exit(0);等，break最常用



**# for each**

**public class Main {**

**public static void main(String[] args) {**

**int[] ns = { 1, 4, 9, 16, 25 };**

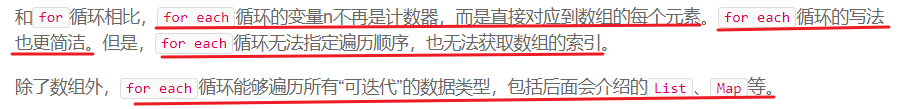
**for (int n : ns) {//int n；n在ns中顺序遍历**

**System.out.println(n);**

**}**

**}**

**}**



注意：在for (int n : ns)循环中，变量n直接拿到ns数组的元素，而不是索引。

**while和for都能用，if也可以，而且常常就是搭配if来的，因为无条件的跳出往往没什么意义**<https://www.liaoxuefeng.com/wiki/1252599548343744/1259542531392800>



**break就是跳出整个循环到上一级去，continue就是跳出本次循环到下一次循环去。（跟之前学的一样）**

#### 关于输出的一个小点

System.out.println("j = " + j);

这样就不用System.out.println("j =%d " , j);用+号直接实现，减少语言包袱

# 3 输入输出

System.out.println();

1 如果是变量就什么都不加

2 如果是字符串加“”

3 如果是混合就或需占位符%d int pi =3.14159; System.out.println(“n = %.2d”，pi);

4 用+拼接输出