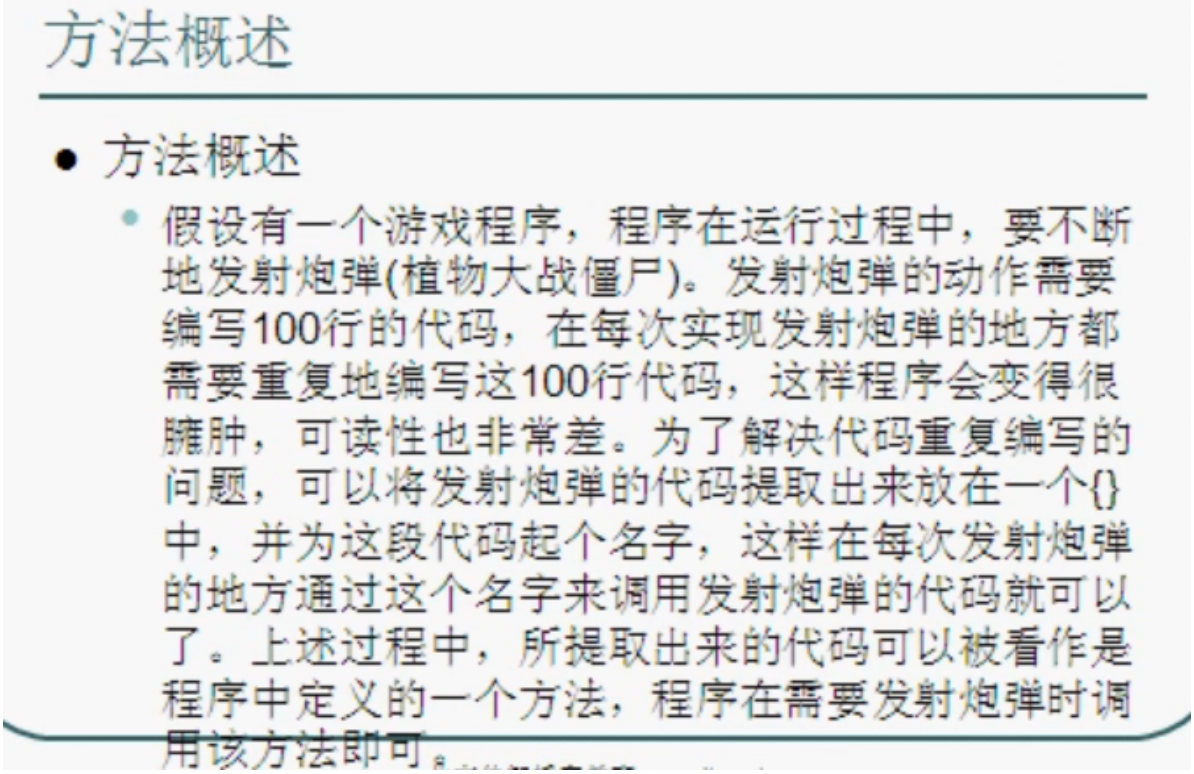
Day5

1、方法定义

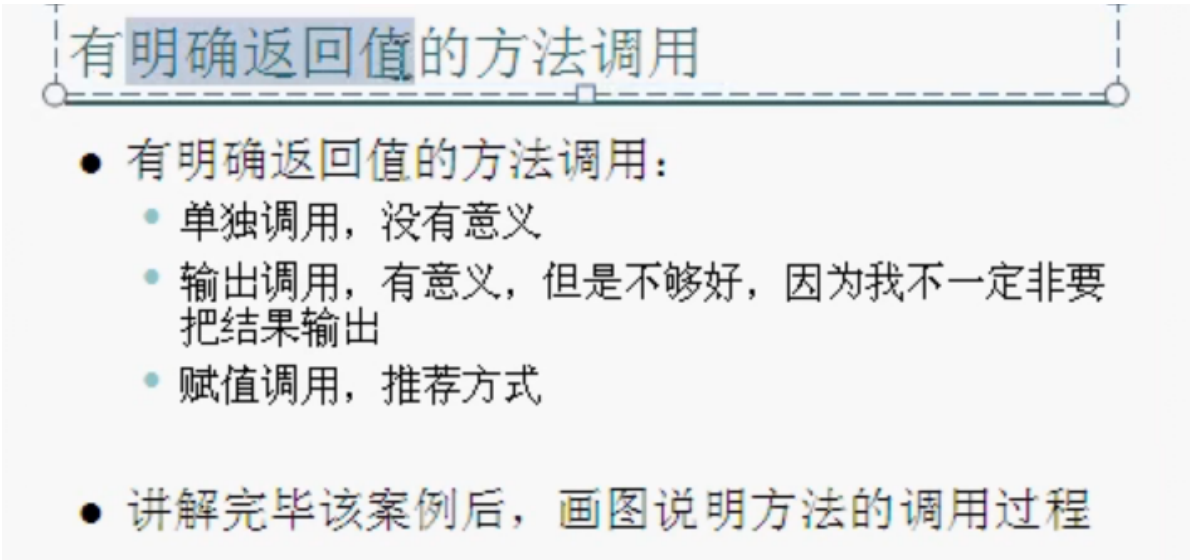
完成特定功能的代码块

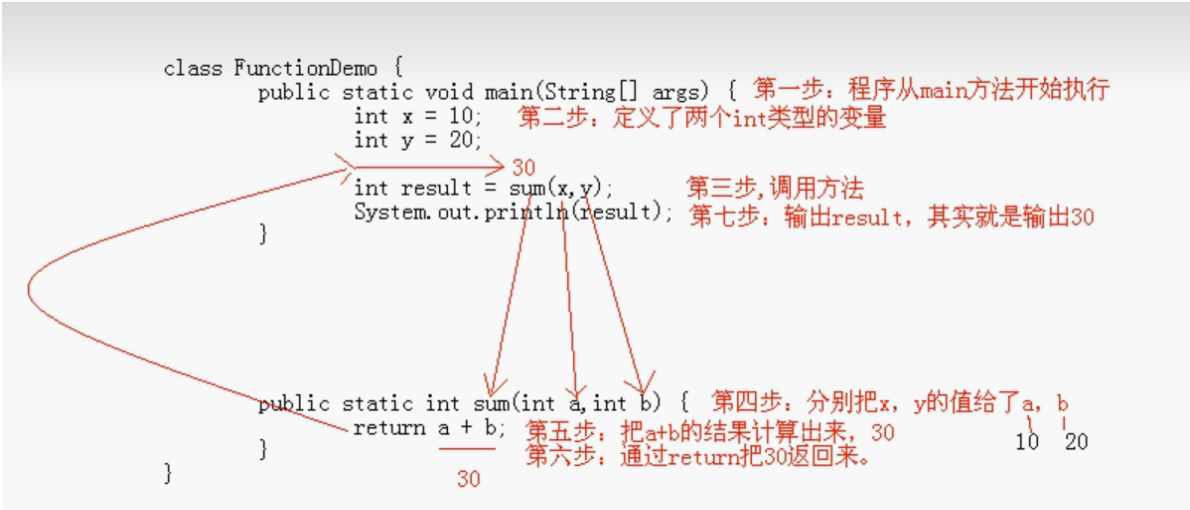
函数在java中称为方法

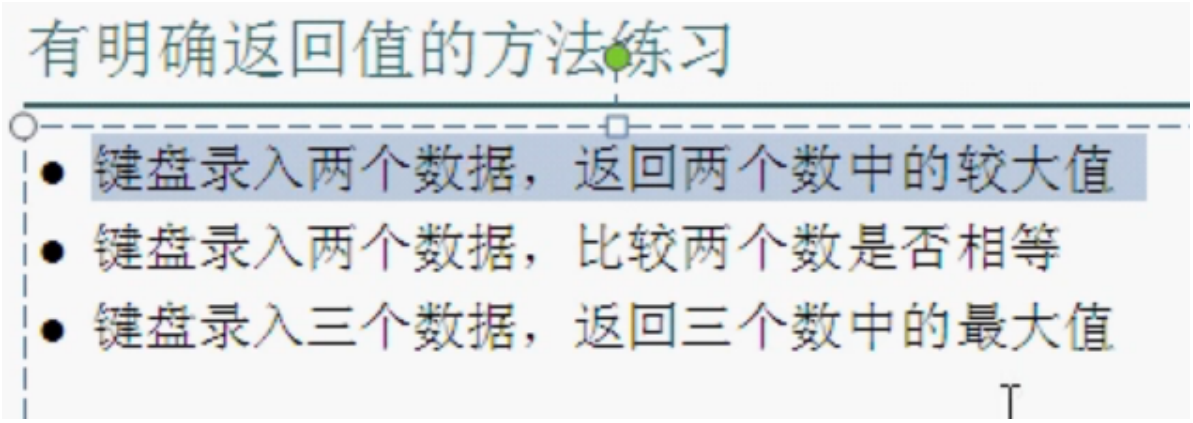



```
格式：
修饰符 返回值类型 方法名(参数类型 参数名1，参数类型 参数名2，参数类型 参数名3){
    方法体语句；
    return 返回值；
}

修饰符：目前就用public static，后面再讲其他的
返回值类型：就是功能结果的类型
方法名：符合命名规则即可
参数：
    实际参数：实际参与运算的
    形式参数：就是方法定义上的，用于接受实际参数的
方法体语句：就是完成功能的语句
return：返回方法的结果
```







注意事项：

方法不执行不调用
方法与方法是平级关系，不能嵌套定义
方法调用时候不用再传递数据类型
方法如果有明确的返回值，要用`return`返回

```
class FunctionDemo4 {
    public static void main(String[] args) {
        //for循环输出图形
        for(int x = 0;x < 4;x++){
            for(int y = 0;y < 5;y++){
            System.out.print('*');
            }
            System.out.println();
        }
    }
}
```

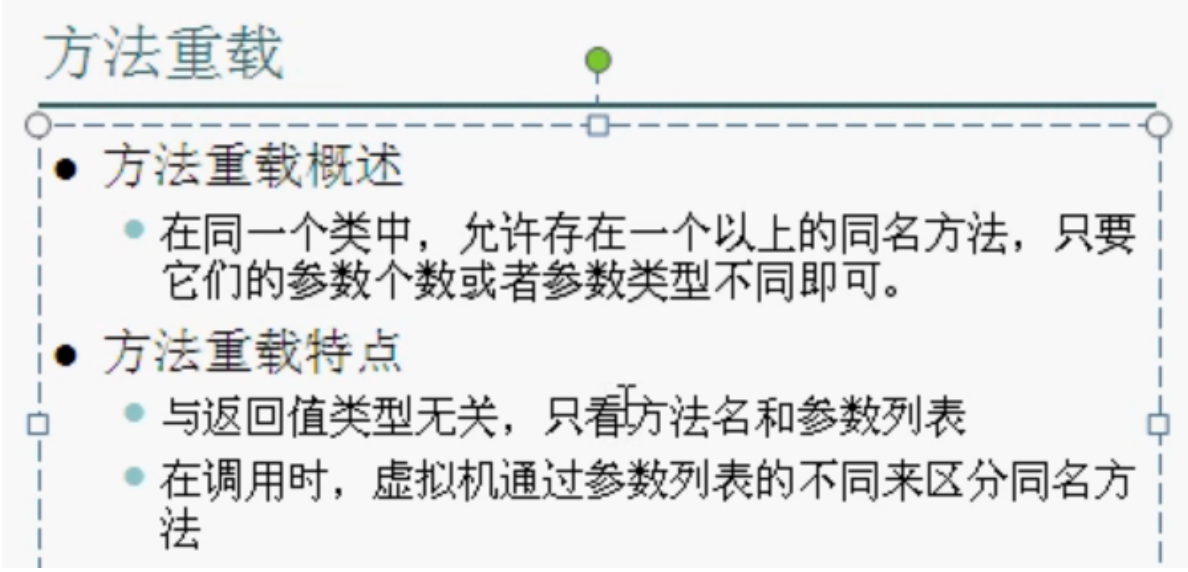
2、方法重载

在同一个类中允许定义多个同名的方法

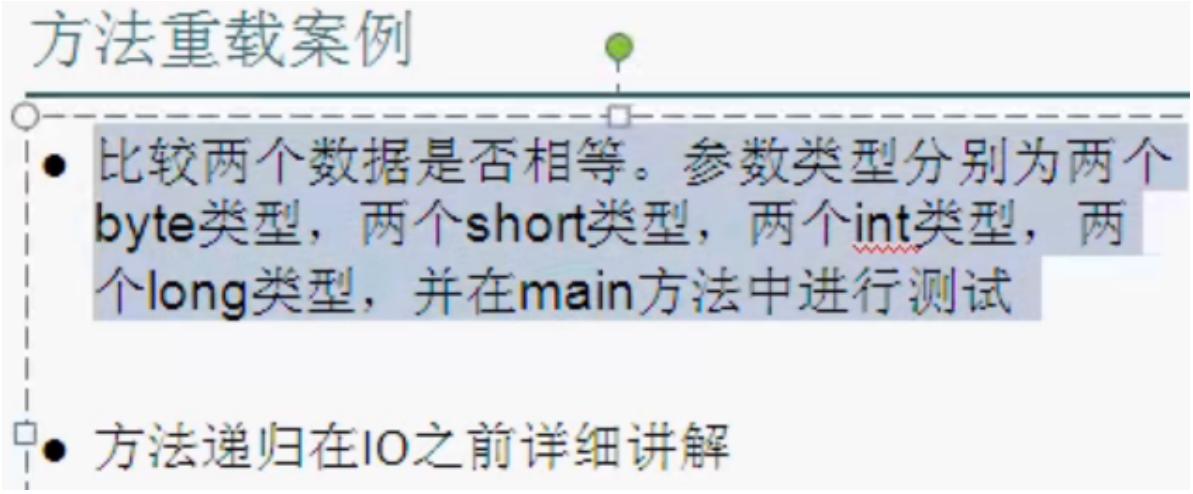
方法名相同，参数列表不同，与返回值无关

可以参数个数不同，也可以参数类型不同

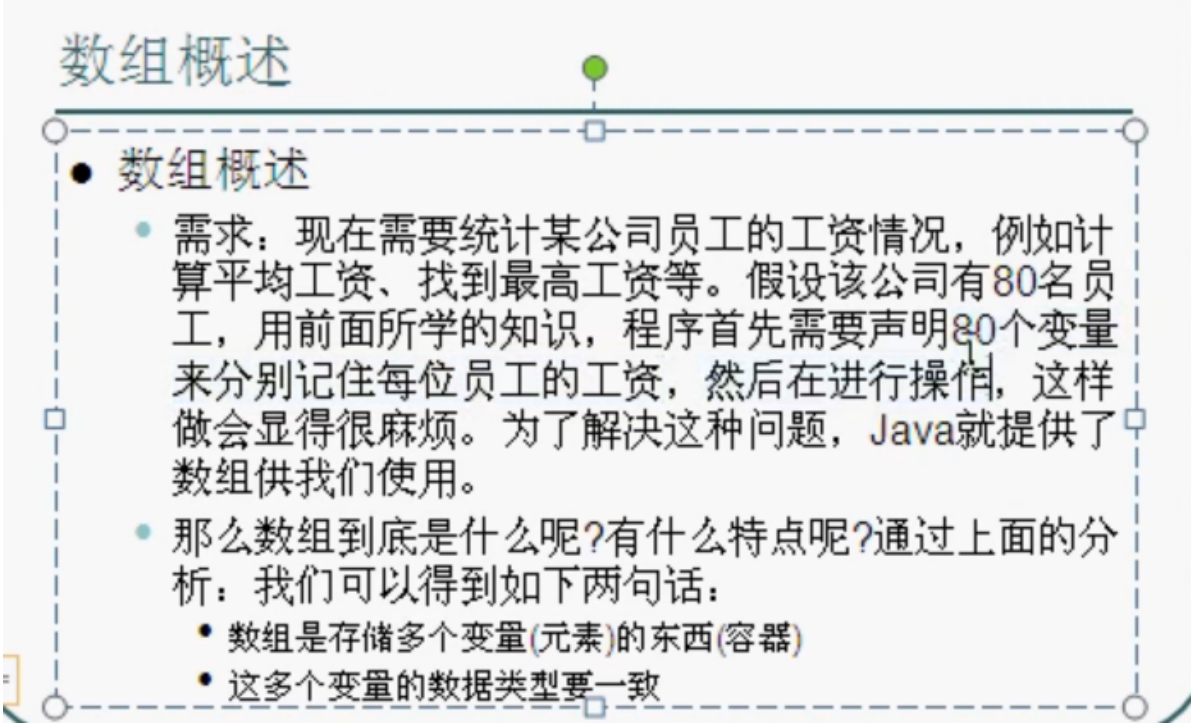
```
    我们的需求不断的发生改变，我们就对应的提供了多个求和的方法。
    但是呢，他们的名字是不一样的。
    而我们又要求方法命名做到：见名知意。
    但是，很明显，现在没有做到。
    那么，肿么办呢？
    针对这种情况：方法的功能相同，参数列表不同的情况，为了见名知意，Java允许它们起一样的名字。
*/
```



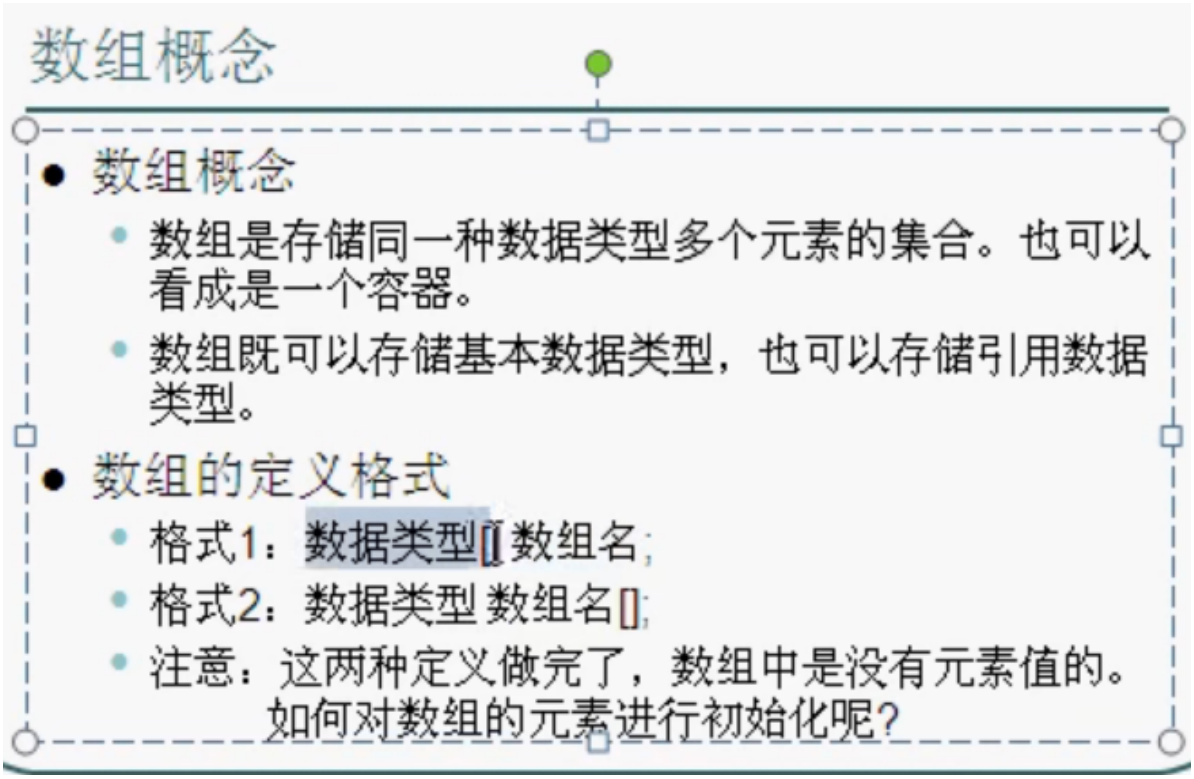
方法重载案例

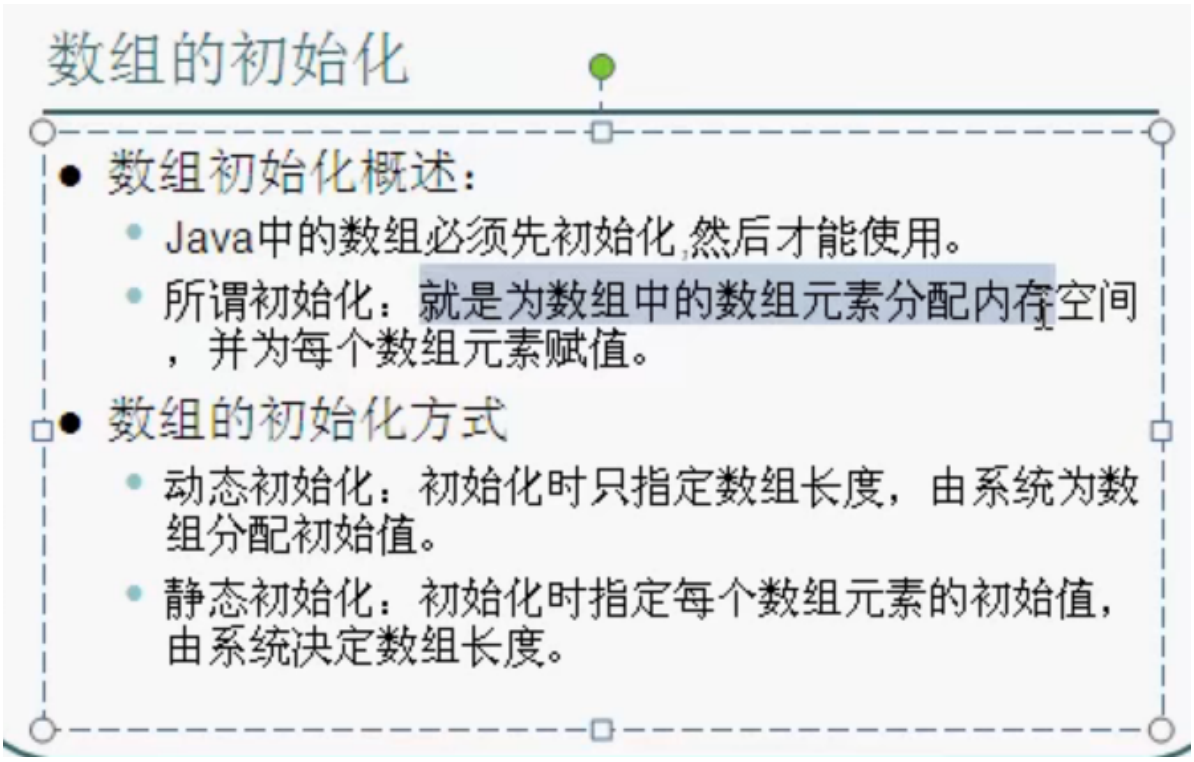


3、数组概述

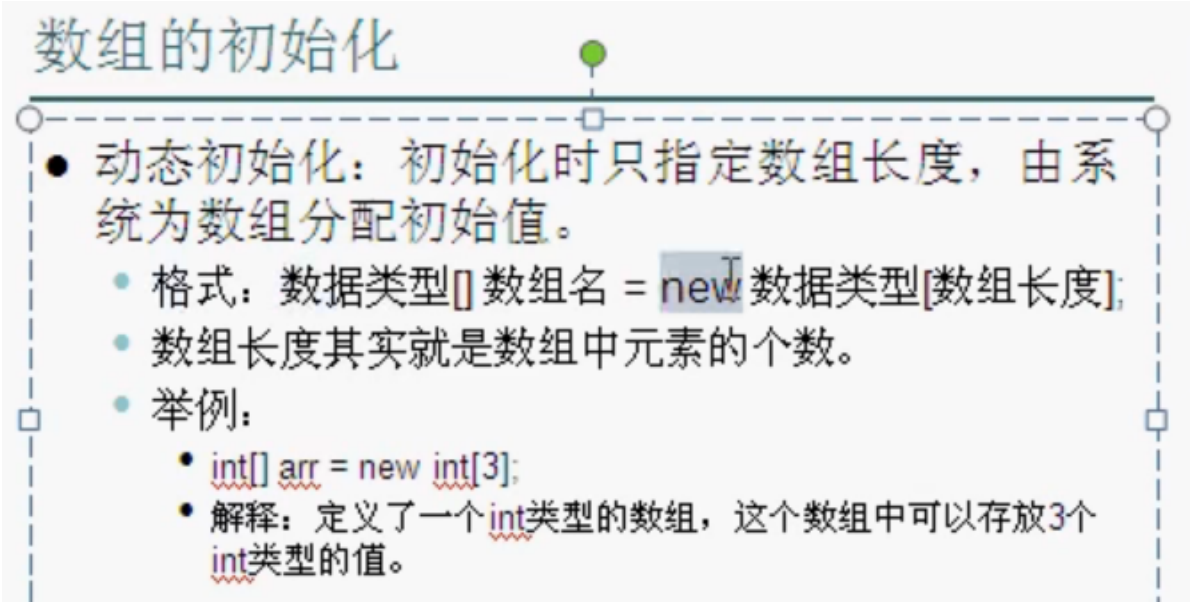


存储同一类型多个数据的集合

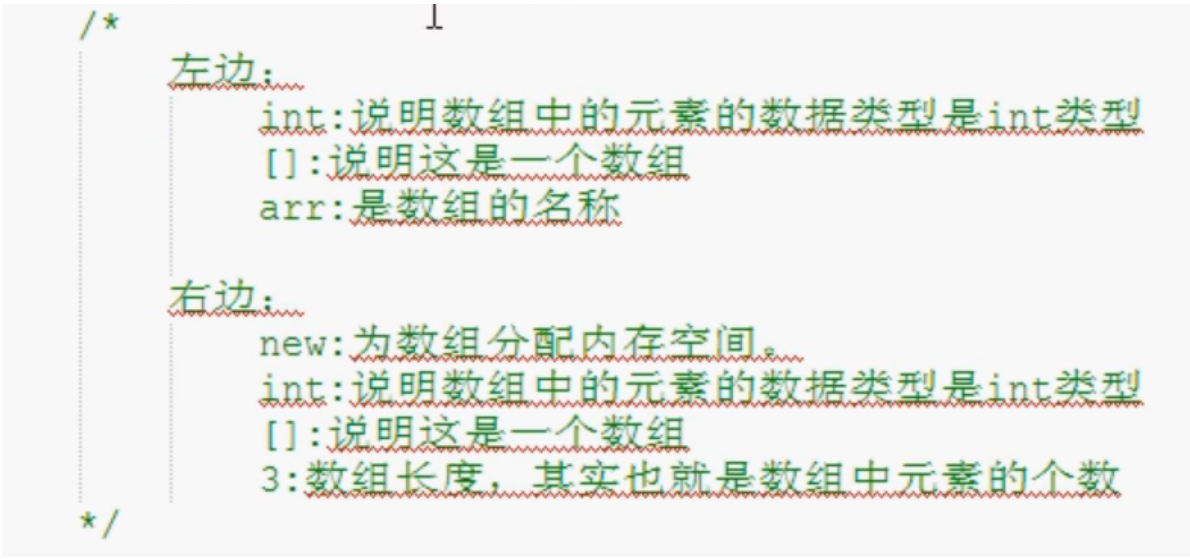




初始化方式：静态初始化和动态初始化



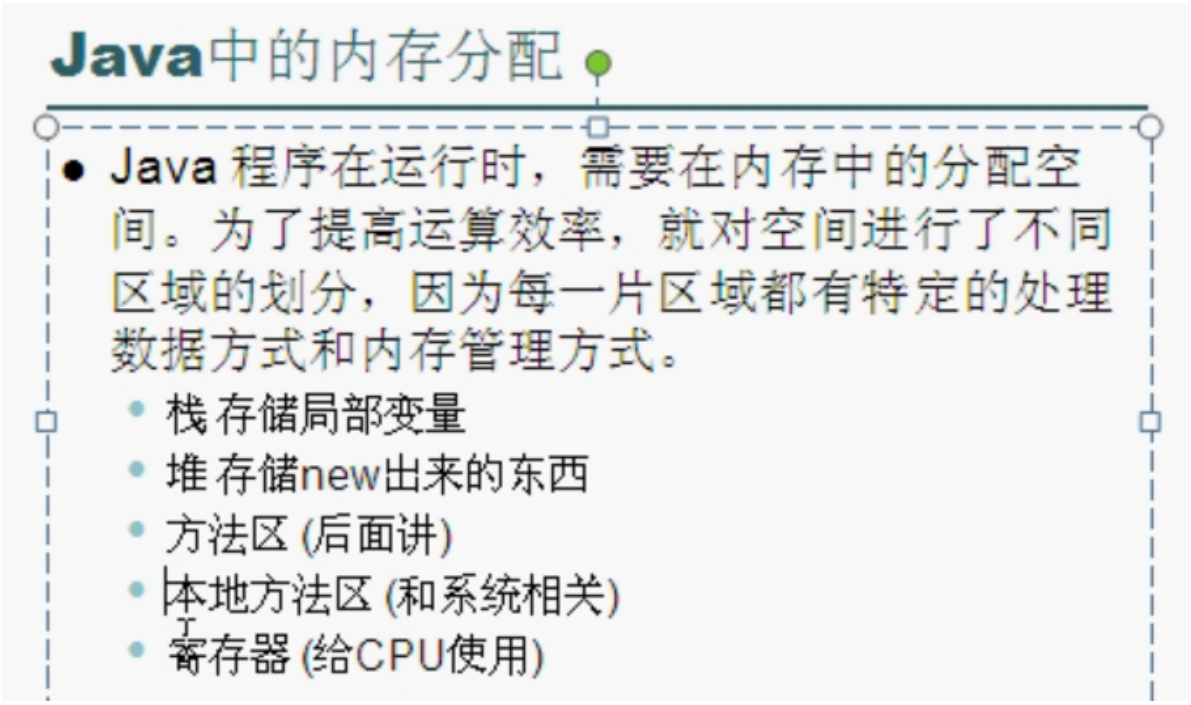

```
public class ArrayDemo {
    public static void main(String[] args) {
//        int[] a;//会报错，因为没有初始化
//        System.out.println(a);
        int[] a = new int[3];
        System.out.println(a); //[I@4554617c地址值
    }
}
```



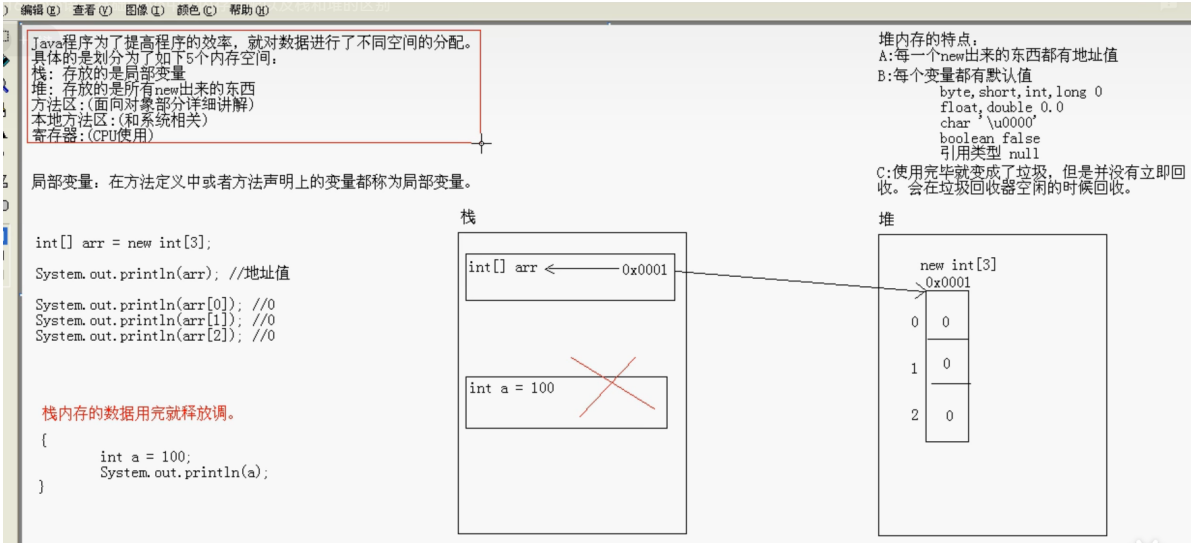

```
System.out.println(arr); //[I@175078b 地址值。
//我要地址值没有意义啊，我就要数据值，怎么办呢？
//不用担心，java为你考虑到了。
//其实数组中的每个元素都是有编号的，并且是从0开始。最大编号是数组的长度-1。
//用数组名和编号的配合就可以获取数组中的指定编号的元素。这个编号的专业叫法：索引
//通过数组名访问数据的格式是：数组名[索引];
```

动态初始化默认值为0

4、Java中的内存分配



后两个了解即可



栈内存在脱离作用域之后就释放

不要同时静态和动态初始化

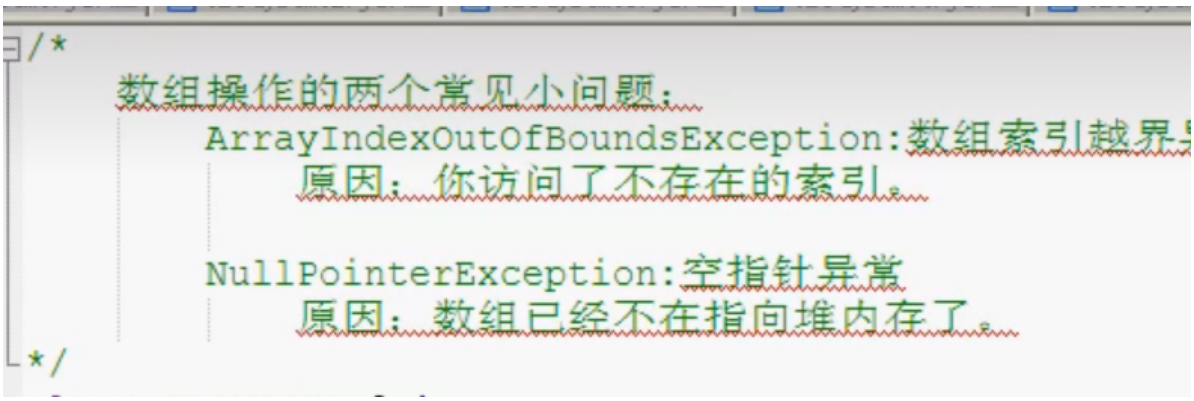
```
int[] arr = new int[3]{1,2,3}
```

这样初始化是错误的

常见问题：

1、数组索引越界

2、空指针异常


/*
数组操作的两个常见小问题：
ArrayIndexOutOfBoundsException:数组索引越界
原因：你访问了不存在的索引。
NullPointerException:空指针异常
原因：数组已经不在指向堆内存了。
*/

p141