Java 7.27 学习

今日目标：今天学习 方法重载 和 可变参数 及 作用域

三．作用域VarScope

变量作用域是非常重要的知识点，需要深刻掌握

1.基本使用

1）在java编程中，主要的变量就是属性（成员变量）和局部变量

属性（成员变量）：

* 定义在类中，方法外
* 在整个类中都可以访问
* 有默认值（如果不初始化）
* 生命周期与对象实例相同

局部变量：

* 定义在方法内
* 只在定义它的方法中可见
* 必须在使用前初始化
* 当方法执行完毕后就会销毁

2）我们说的局部变量一般是指在成员方法中定义的变量（举例：Cat类：cry）

3）Java中作用域的分类

全局变量：也就是属性，作用域为整个类体

局部变量：也就是除属性之外的其他变量，作用域为定义它的代码块中

4) 属性（成员变量）可以不赋值，直接使用，因为有[默认值](做题过程中的总结记录.docx)，局部变量在使用必须先赋值，因为没有默认值

2.作用域的使用事项和细节VaribleScopeDetail

1）属性和局部变量可以重名，访问时遵循就行原则

2）在同一个作用域中，不可以重名。比如在同一个成员方法中，两个局部变量，不可以重名

3）属性生命周期较长，伴随着对象的创建而创建，伴随着对象的销毁而销毁。

4）局部变量生命周期较短，伴随着它的代码块的执行而创建，伴随着代码块的结束而销毁，即在一次方法调用过程中。

5）作用域范围不同

全局变量/属性：可以被本类使用，或其他类使用(通过对象调用)

局部变量：只能在本类中对应的方法中使用

6）修饰符不同：

全局变量/属性前面可以加修饰符（public protected private）

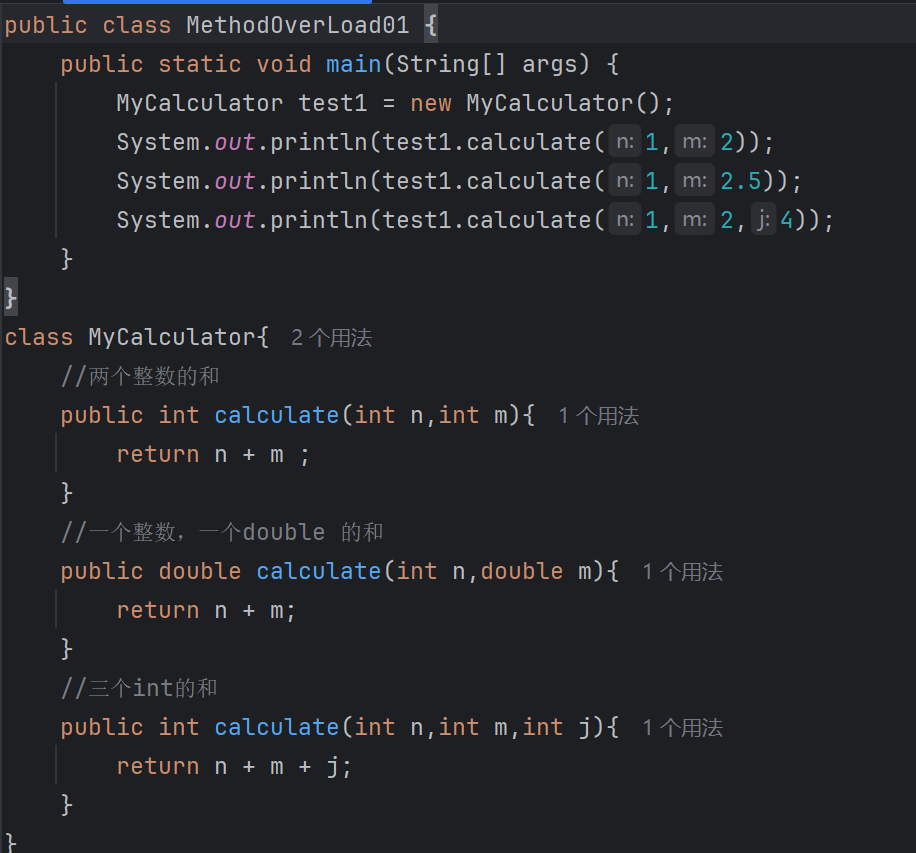
局部变量前面不可以加修饰符

一．方法重载

1. 方法重载基本介绍

在Java中,允许同一个类中存在多个同名方法,这种特性称为方法重载(Method Overloading)。但要求形参列表不一致，意味着这些同名方法必须在参数上有所区别

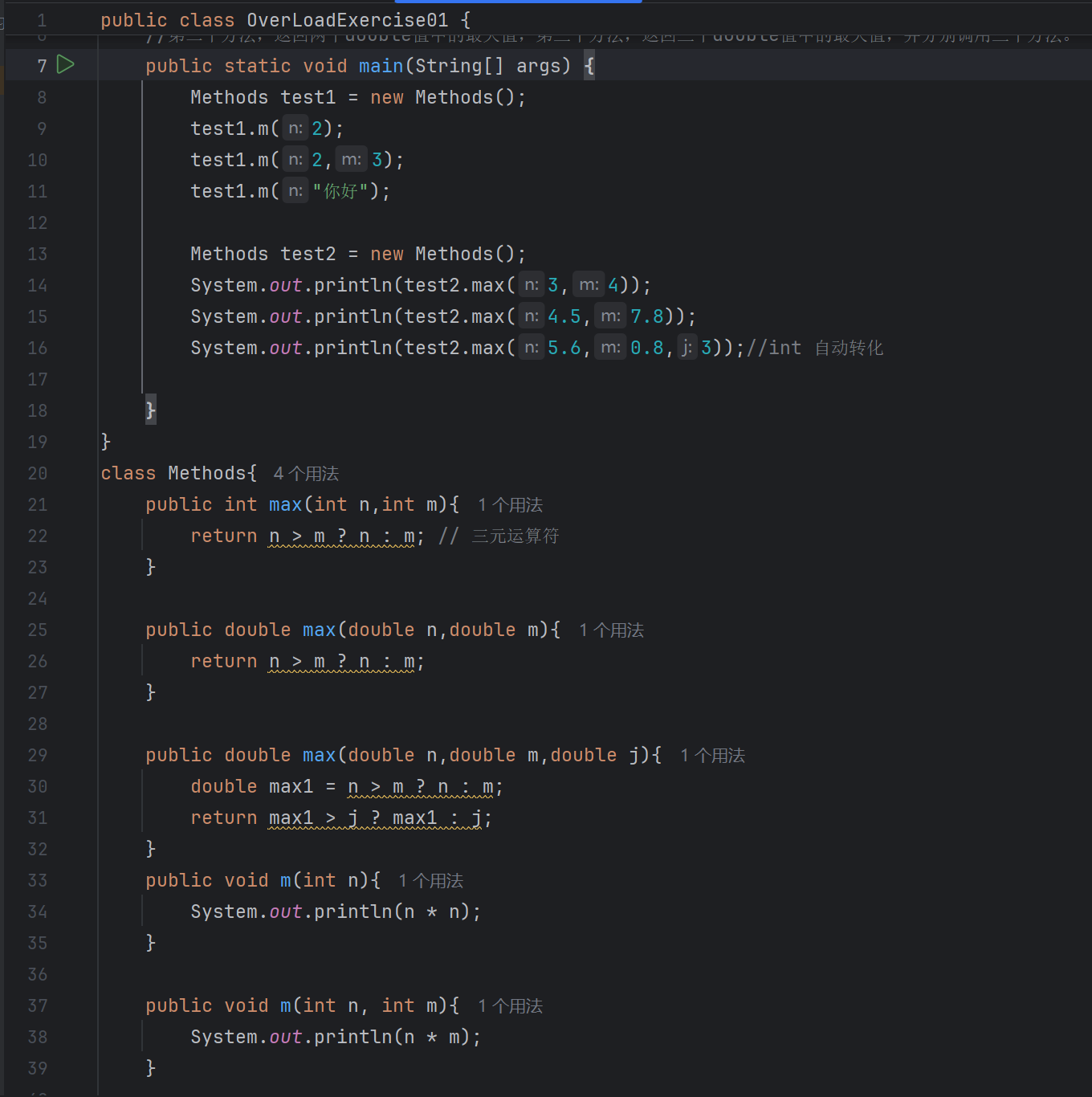
1. 方法重载的举例：MethodOverLoad01



1. 注意事项和使用细节
2. 方法名：必须相同
3. 形参列表：参数数量、参数类型、参数顺序不同，参数名无要求
4. 返回类型：无影响
5. 练习







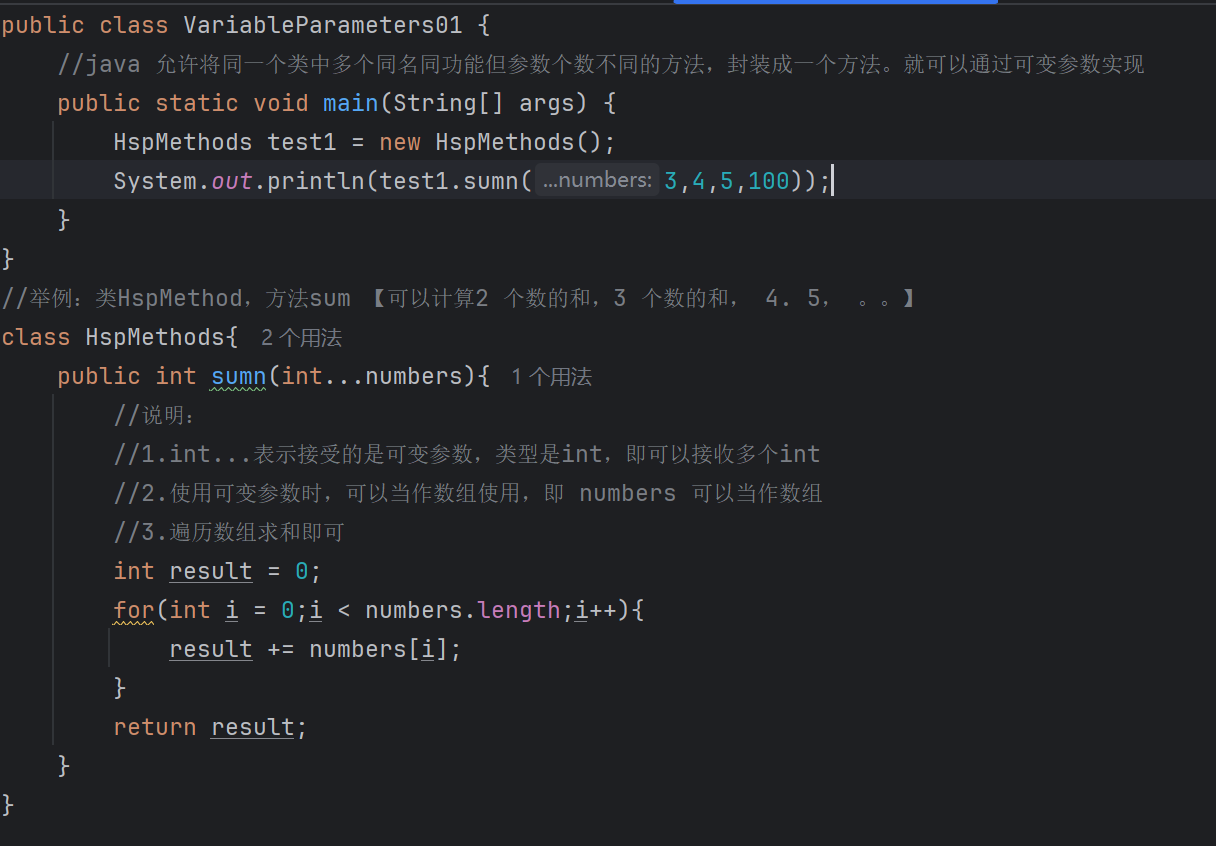
二．可变参数

1可变参数基本介绍

Variable Parameter

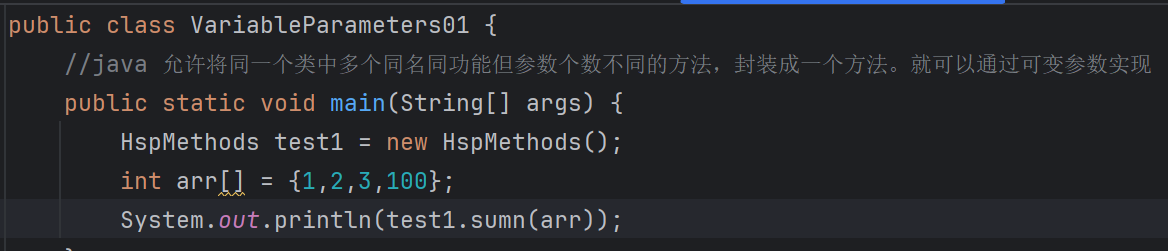
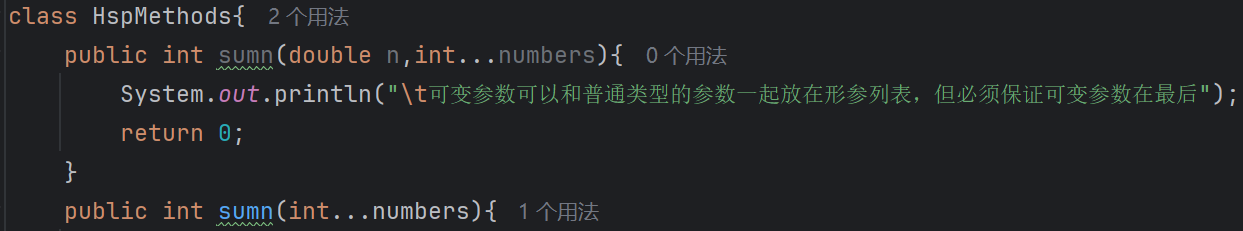
Java允许将同一个类中多个 同名 同功能 但 参数个数不同 的方法，封装成一个方法。就可以通过可变参数实现

2.举例



3.补充说明：

class HspMethods{  
 public int sumn(int...numbers){ //🡨格式  
 //说明：  
 //1.int...表示接受的是可变参数，类型是int，即可以接收多个int  
 //2.使用可变参数时，可以当作数组使用，即 numbers 可以当作数组  
 //3.遍历数组求和即可  
 int result = 0;  
 for(int i = 0;i < numbers.length;i++){  
 result += numbers[i];  
 }  
 return result;  
 }  
}

1. 注意事项和使用细节
2. 可变参数的实参可以为0个或者任意多个
3. 可变参数的实参可以为数组
4. 可变参数的本质就是数组
5. 可变参数可以和普通类型的参数一起放在形参列表，但必须保证可变参数在最后
6. 一个形参列表最多只能出现一个可变参数
7. 练习

