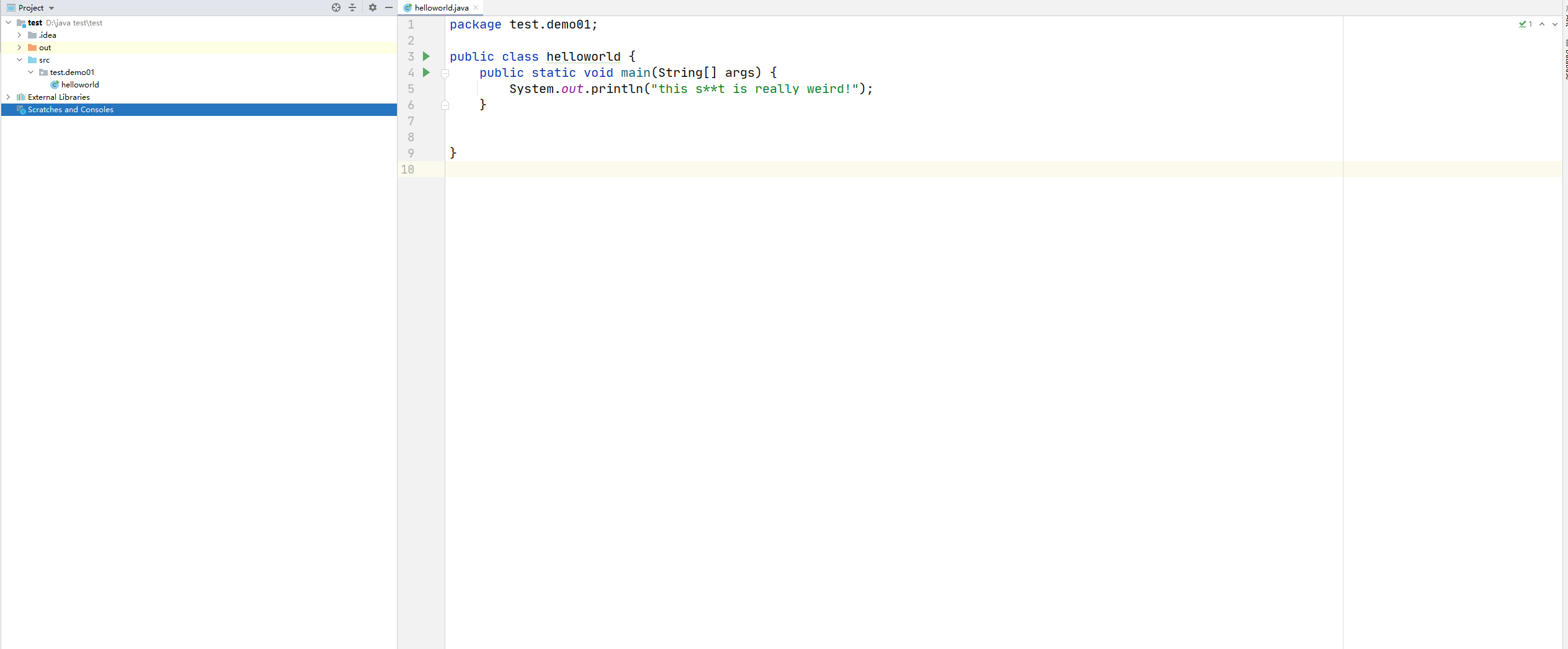
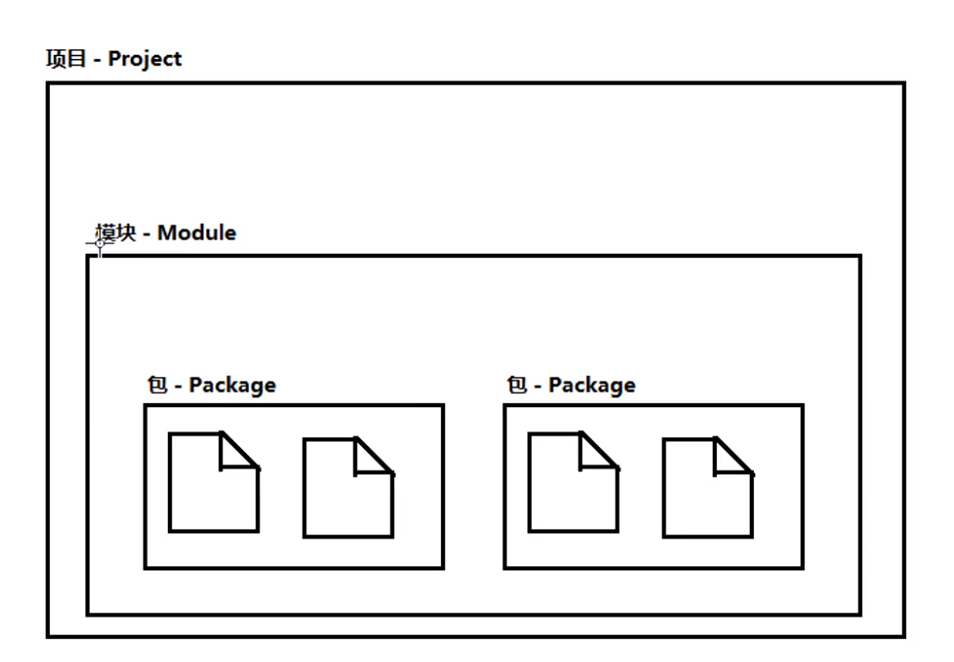
1. java同csharp一样，是完全面向对象的语言。所有的方法都应写于类体中。
2. idea的项目结构：

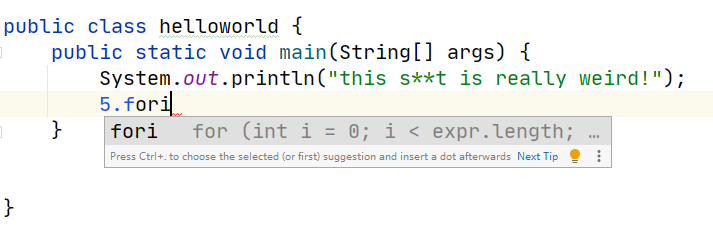


Class建于package内。方法（包括main主入口）写在类体内。

1. 常用快捷键：
2. 导入modele

首先将文件夹放在project目录下；file-project structure-modules-+号-import modules

1. 快捷输入循环： 5.fori



1. 函数重载overload：

和c++基本相同。

多个方法的名称一样，但参数列表不一样。

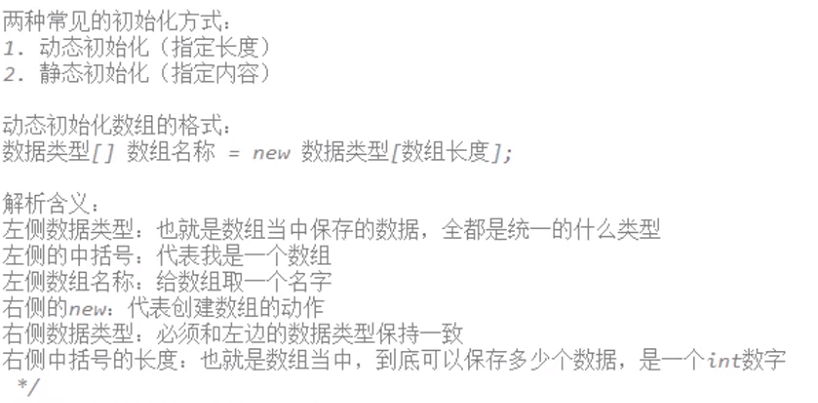
重载与以下因素无关：参数名称（废话）、方法返回值类型.

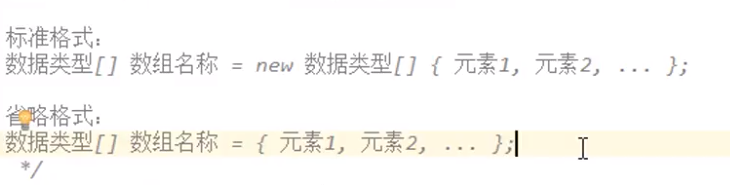
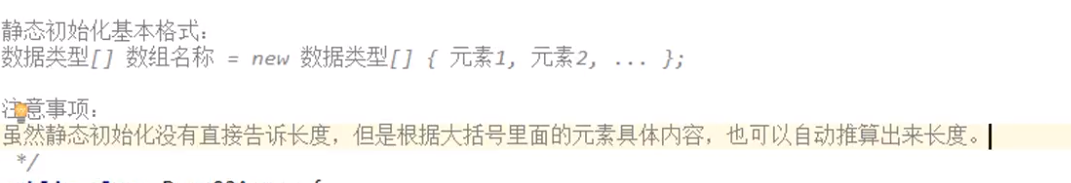
脑部为啥一个print（）随便放参数啥都能输出。

1. 数组：

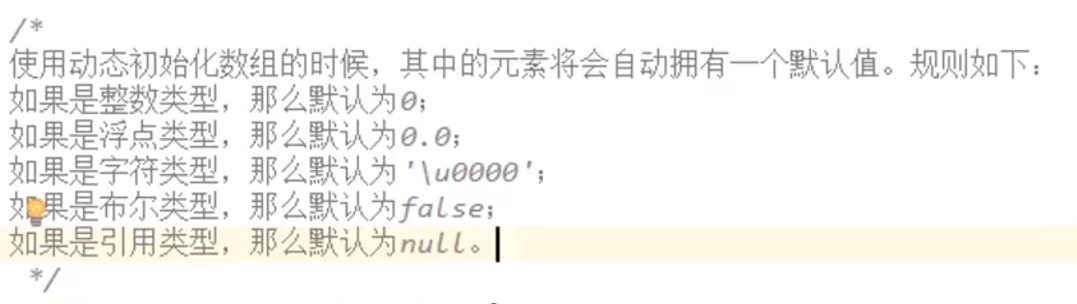
和c++一样的静态数组，无法改变长度。存储和引用方法相同。

初始化语句略有不同：

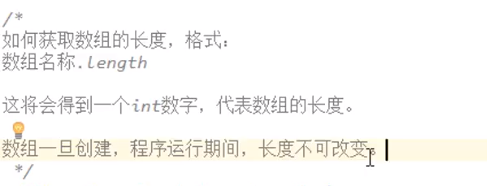


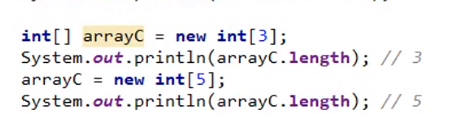


1. 数组的default：



1. 获取数组长度：

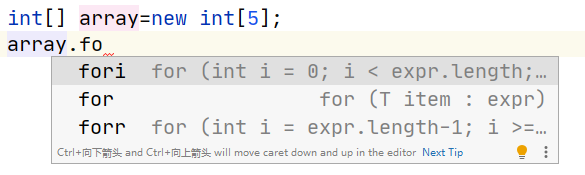


易错点：

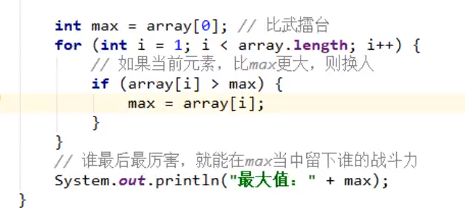
这种情况下数组长度好似改变了，其实不然；这其实是两个数组（两次new），只不过第一个数组叫arrayC，第二个数组抢了第一个数组的名字。

数组的遍历：

数组名称.fori 即可



数组求最值：（没有库函数）

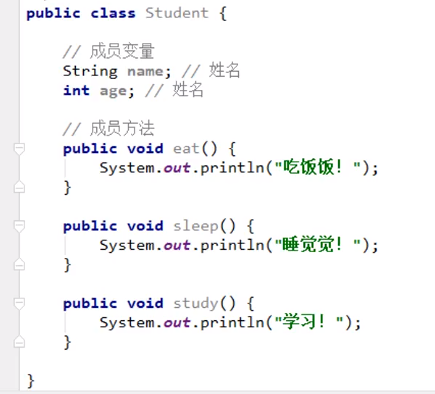


1. 输入输出：

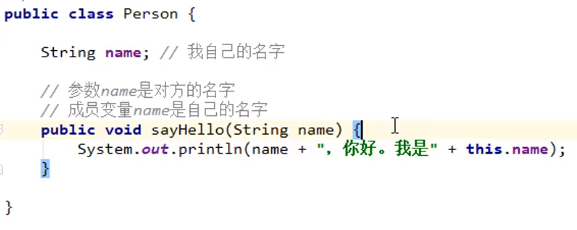


1. 类的定义：

成员方法一般不public

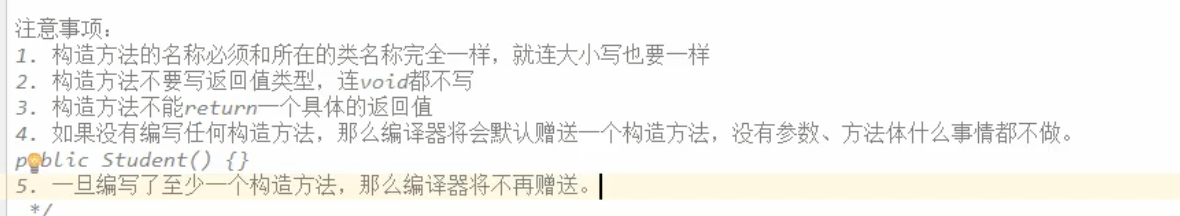


This指针：同c++



This肯定是写在方法内部的，this指的作用域就是本方法所在类。

构造方法：和c++构造函数一样。

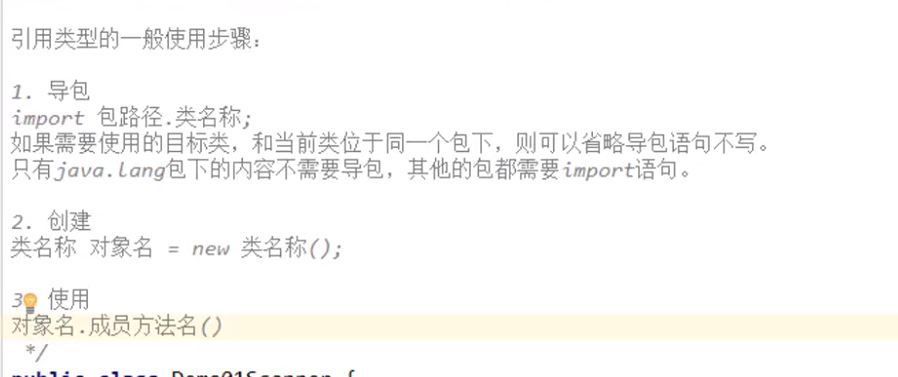


1. 常用API



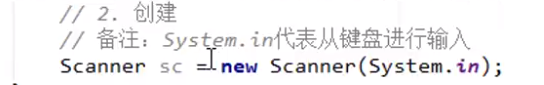
Api就是一大堆类、一大堆方法。文档就是字典。文档的话可以先看1.6中文版。

使用api的方法：



Scanner：





查api文档得知：

