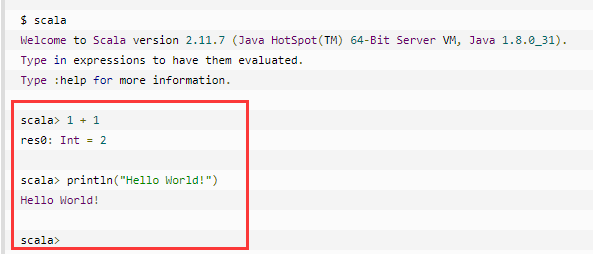
# 1 Scala

## 编程方式

### 交互式编程

图1



### 脚本编程

编写\*.scala脚本，然后scalac编译成\*.class，然后scala \*.class运行。

图1



## 1.2 变量

### 1.2.1 变量声明

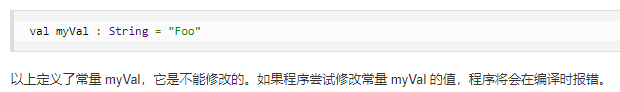
在Scala中，用关键字var声明变量，用关键字val声明常量。

（注意：val声明的常量是不能被程序修改的，否则会编译时报错）

图1



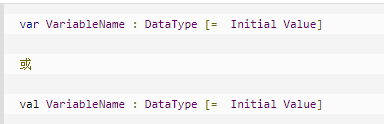
图2



### 1.2.2 变量类型声明

变量的类型在变量名之后，等于号之前声明。

图1

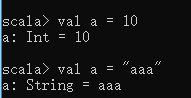


#### 1.2.2.1 类型推测

变量的类型不一定要声明，如果不声明，Scala会根据变量或常量的初始值推测出来。

（注意：如果既不声明变量类型，又没有初始值，则会编译报错）

图1



## 1.3数据类型

原文链接：<https://www.cnblogs.com/qingyunzong/p/8858465.html>

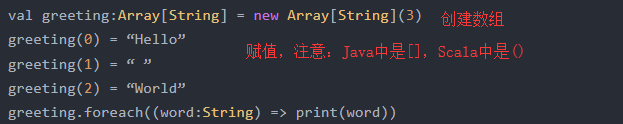
Scala支持的数据类型如下：

图1



### 1.3.1 数组（Array）

图1

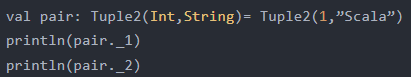


### 1.3.2 元组（Tuple）

元组可以包含不同类型的元素，然后用“.\_基于1的索引”访问元组的元素。

（注意：元组最多支持22个元素，超过22个元素会编译时报错）

图1



### 1.3.3 映射（map）

映射就是K/V的结构。

图1



### 1.3.4 Set

Set是没有重复的元素的集合。

图1



## 1.4 方法和函数

Scala的方法是组成类的一部分。Scala的方法的语法如下：

图1

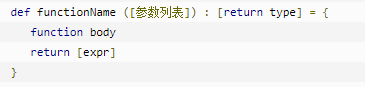
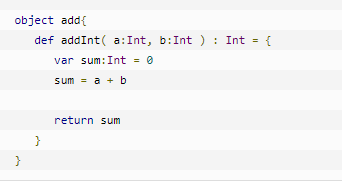


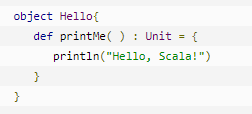
图2



### 1.4.1 无返回值的方法

如果方法无返回值，return type为Unit，相当于Java的void。

图1

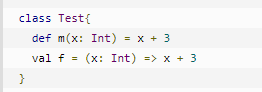


### 1.4.2 函数和方法的区别

方法是组成类的一部分。而函数是一个完整的对象，函数实际上是继承了Trait类的对象。

方法用def定义，函数用val定义。

图1



### 1.4.3 方法的简写

原文链接：<https://blog.csdn.net/u010916338/article/details/80454625>

图1（完整方法声明）

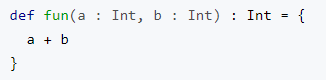


图2（没有参数，()可以省略）

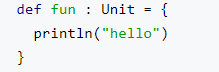


图3（没有返回值，Unit也可以省略）

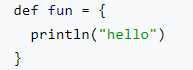


图4（方法体只有一行时，{}也可以省略）



### 1.4.4 =>

=>是创建匿名函数的语法糖。

A,B是参数的类型，T是返回值的类型。

图1（=>创建匿名函数的完整版）



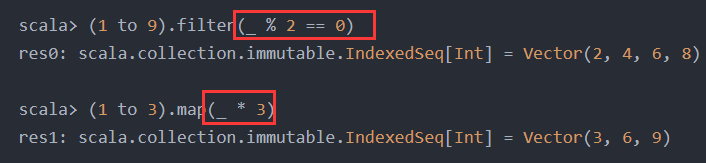
图2（参数为空）



图3（返回值为空）



图4（\_表示参数，=>的左边可以整个省略）



## 1.5 类和对象

类是抽象的，不占用内存；对象是具体的，占用内存。对象是类的具体实例。

图1（Scala中定义类）

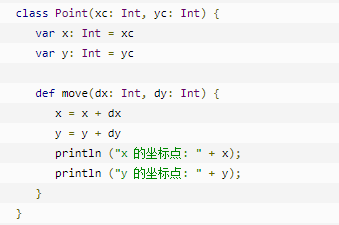
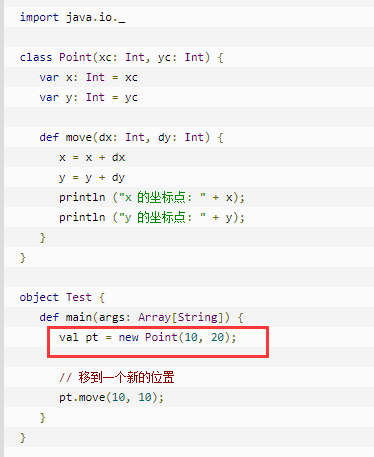


图2（Scala中定义对象）



### 1.5.1 单例对象

Scala中没有static，所以类的单例模式（只有一个实例的类）就是单例对象。

（注意：单例对象不能带参数）

图1

