# 简介

运行于JVM之上的一种函数式语言。

# 特点

分布式、高性能、易拓展、JVM生态

# 性能

高

# 适用

分布式、函数

# 概念

## 函数是第一等公民

## call by value：及时求值、替换形参

对函数实参求值，且仅求值一次。

## call by name：用时求值、惰性求值

每次在函数体调用都会求值。动态。

如果函数形参类型以=>开头，那么会使用call by name。

入参实际是函数

## 柯理化

## Actor

## object与class区别

1. Object内方法、变量可以直接访问（静态）
2. Class需要由实例来承载，使用时需要new
3. Class可以包含构造器，object不可以
4. Object与class位于同一个scala源码文件中，且名称相同时，成为该类的伴生对象

## 伴生对象

# 安装

# 编译

# 工具

Idea（推荐使用）

Eclipse

# 语法

## 类型

Byte

Int

Short

Long

Boolean

Float

Double

Char

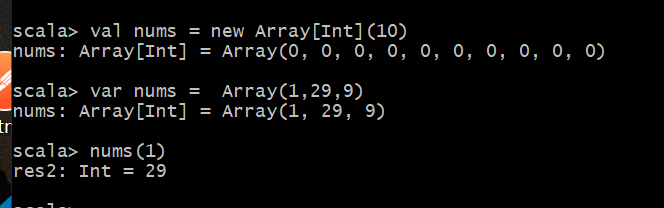
Any

class

Object

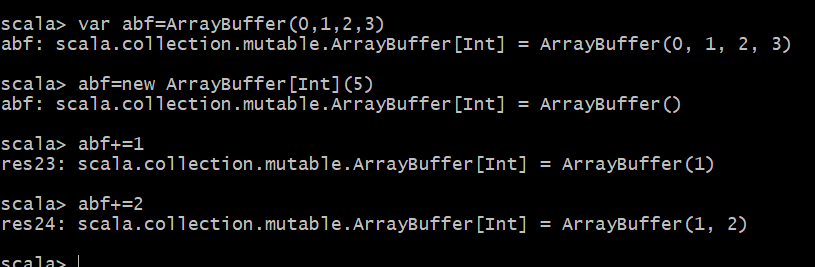
### 数组

#### 定长数组Array



#### 变长数组ArrayBuffer

声明



+ 操作一个值

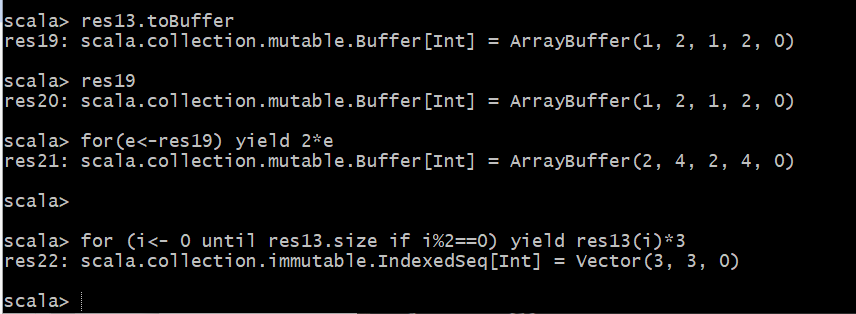
++ 结合、数组

= 修改源缓冲区

: 以:结尾通常是前导操作

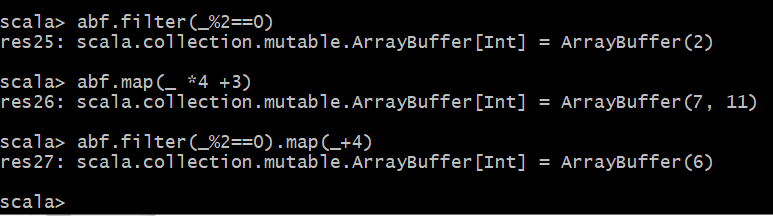
#### 数组变形

##### Yield



类似功能：python里面的列表解析表达式

##### filter



## 变量

## 常量

**lazy**

## 操作符

没有++，--

+

-

\*

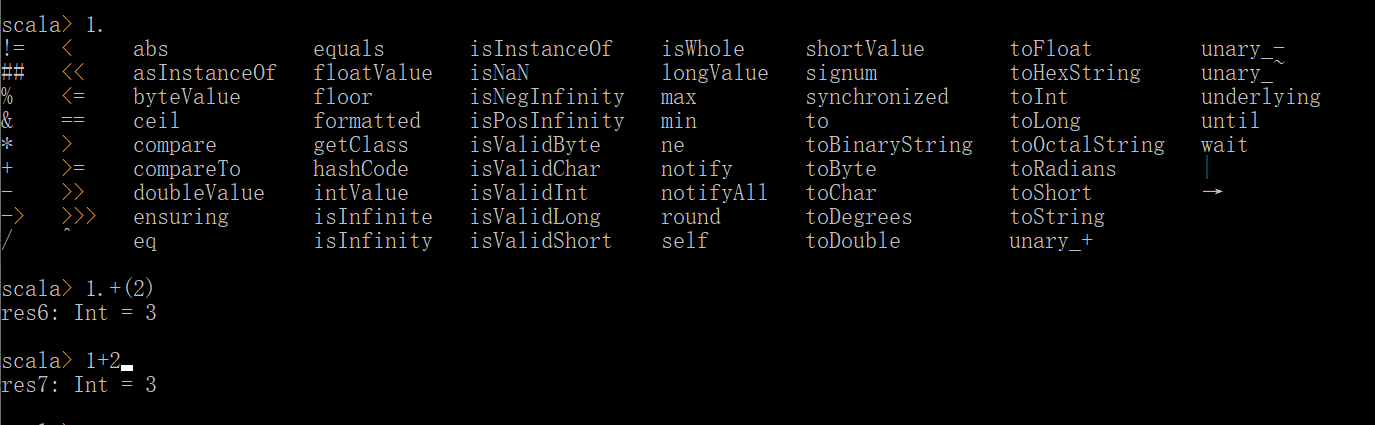
/

%

+=

-+

### 操作符重载



## 函数

无返回值函数：

def 函数名(参数名:参数类型,...)={

函数体

}

带返回值函数：

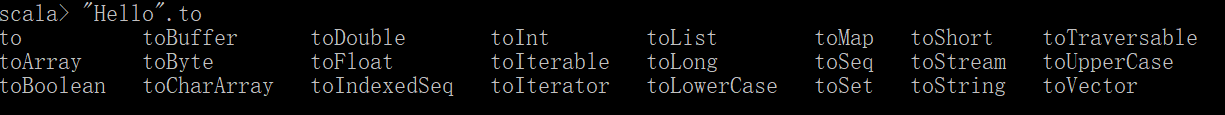
def 函数名(参数名:参数类型,...):返回值类型={

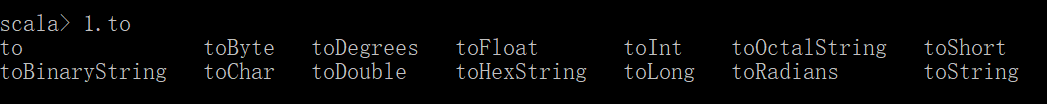
函数体

返回值

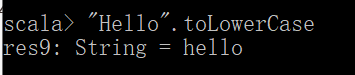
}

#### 没有值类型，都是对象

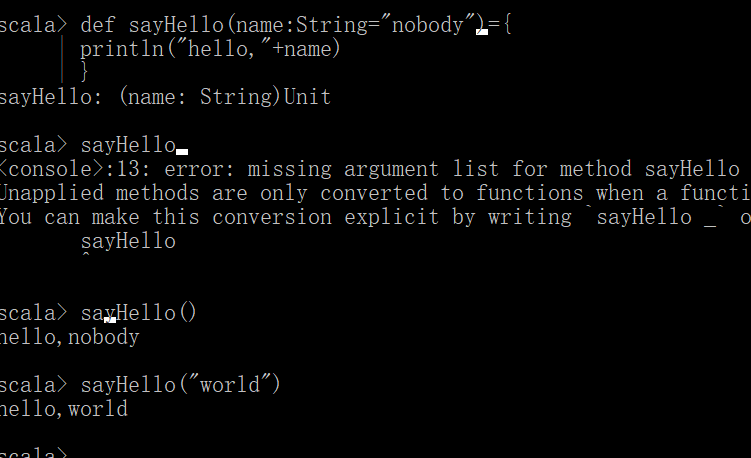




#### 无参函数调用无需()

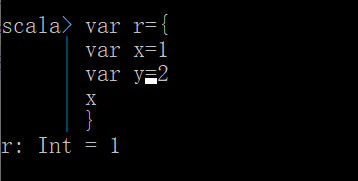


#### 参数默认值

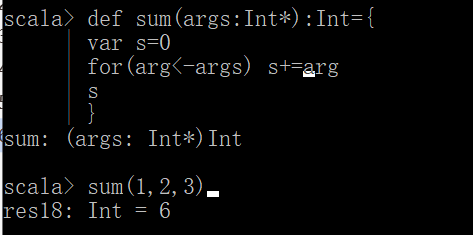


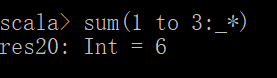
#### Return

函数的最后一行就是返回值



#### 变长参数





## Procedures

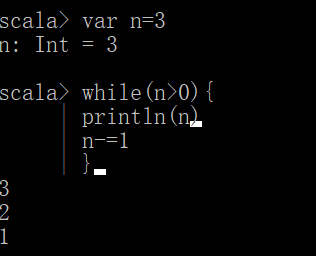
和函数很像，没有返回值，不用=来定义

## 赋值

## 逻辑

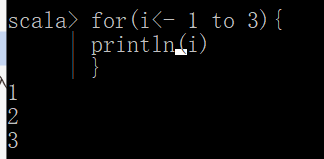
## 循环

### while

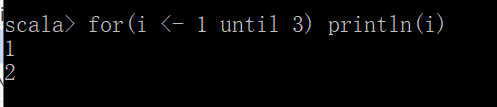


### For

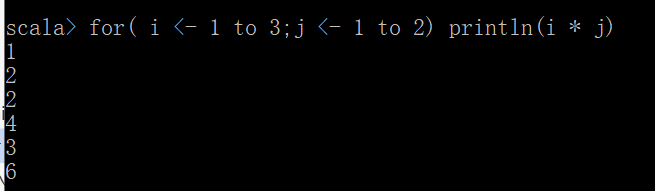
#### to



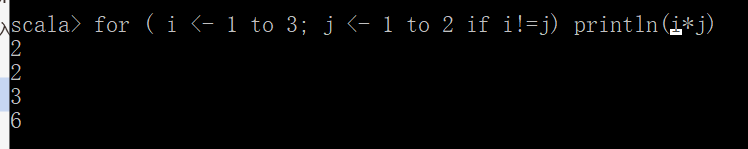
#### until

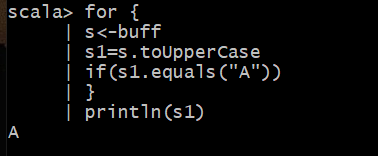


#### 嵌套扁平化



#### 加逻辑





## 依赖导入

## 异常

### try catch表达式

## match

exp match{

}

# 并发

# 设计模式

# 框架