# 基本语法

## 字符串

这里的 String 就是 java 的 String 。所以方法通用

### 声明

val msg1: java.lang.String = “hello world”

var msg2:String = “hello”

var msg3 = “world”

这里，val， var 表示是否 const ， 指定变量或者表达式类型。

### 创建多个字符串

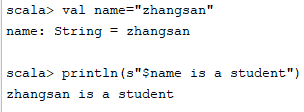
Val msg = “””lsdld

Sdlsld”””

用三个 “”” 可以创建出多行的字符串

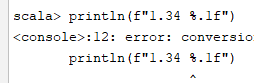
### 插值器

s 插值器是一个函数， ${ } 大括号内任意表达式

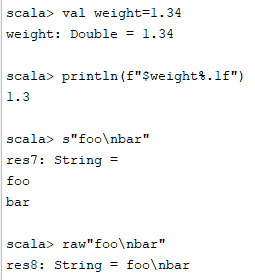




f插值器直接格式化了变量， raw插值器原样输出。

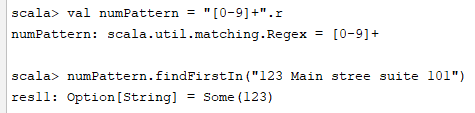




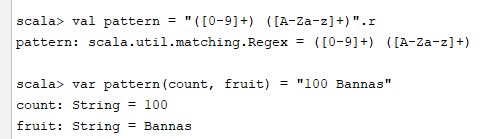


### 正则表达式

.r 方法产生正则表达式，这里返回了 Option 可能是 null，也可能不是。



从 pattern 获取匹配值



## 区间

### To



### By



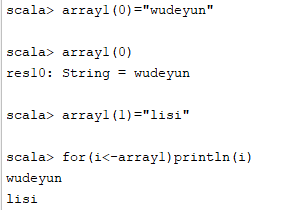
### Until

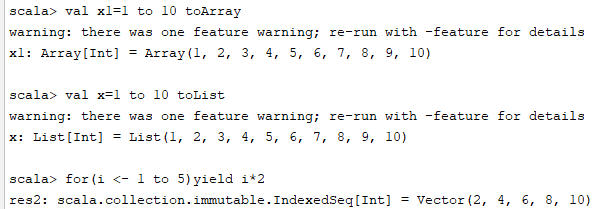


## 数组

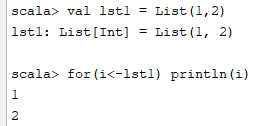








## List



数组合并

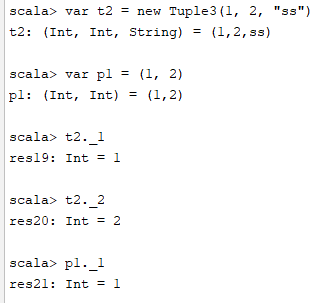




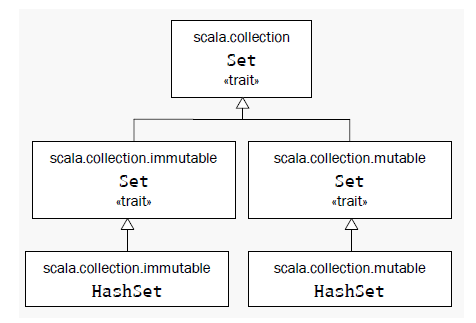
空表



## 向量

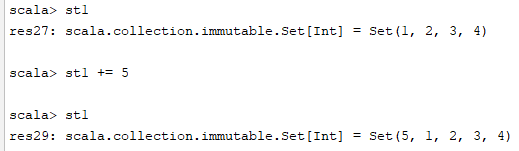


## Set 和 Map

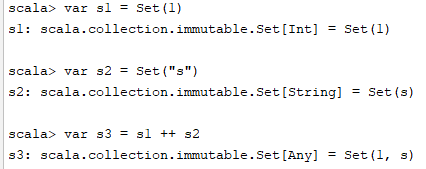


添加元素





合并集合

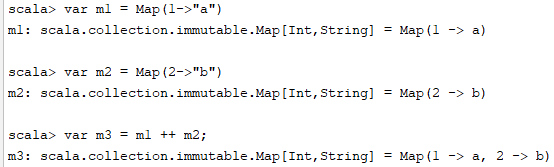


交集

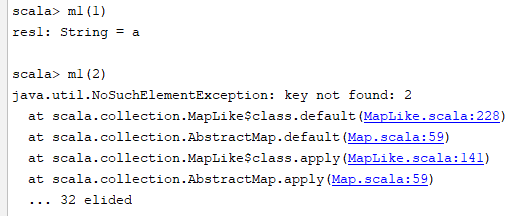




map合并

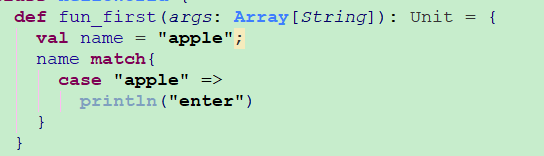


获取值



# 控制语句

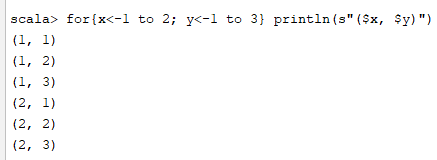
## Match



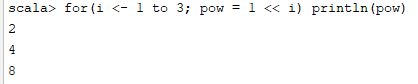
## For



两层循环

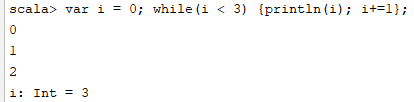


值绑定 pow



## While





# 函数

## 定义

1. 标准定义

**def** addInt(a:Int, b:Int):Int = { **return** a + b}

2. 不指定返回值类型，推断返回类型，因为返回类型是推断的，不允许 **return** 关键词

**def** addInt(a:Int, b:Int) = { a + b}

3. 单句

**def** addInt(a:Int, b:Int) =  a + b

4. 无返回值，没有 "=" 就表示无返回值 。显示指定无返回值用关键词 Unit ，则会个时候，可以用 return 关键字，也可以不用

**def** addInt(a:Int, b:Int){a + b}

**def** addInt(a:Int, b:Int):Unit= {a + b}

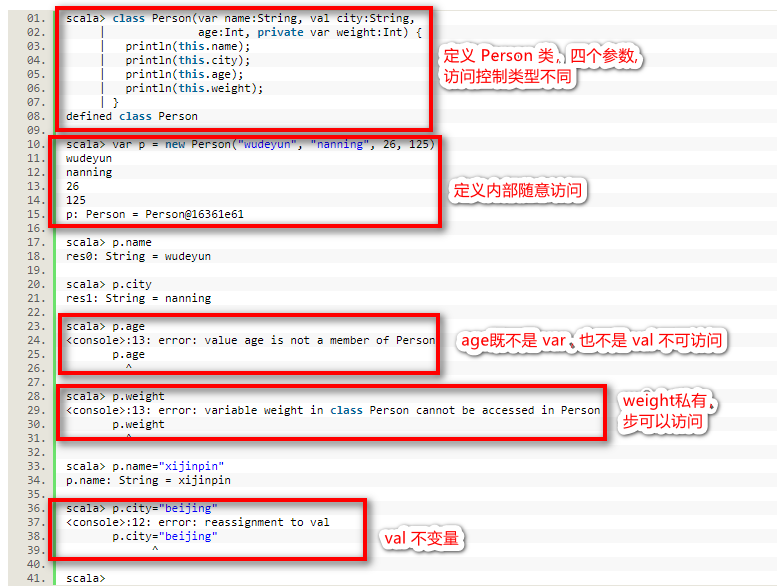
**def** addInt(a:Int, b:Int):Unit= {**return** a + b}

5. lamda表达式

**def** addInt = (a:Int, b:Int) =>  a + b

# 类

可见性



还可以 class Person(name:String); 这就行了，不用函数体。

## case模板类

**case class Person**(**var** *name*: *String*, **var** *age*: Int);  
  
**object Person** {  
 **def** apply() = **new** Person(**"mike"**, **21**);  
 **def** apply(*name*: *String*) = **new** Person(**"jdon"**, **34**)  
}

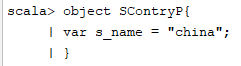
这种是使用了模板模式，这里可以不用关键词 new 而是直接使用模板方法

**val** a = **Person**();  
**val** b = **Person**(**""**);  
**val** c = **Person**(**"william sh"**, **82**);

实际最终会调用 apply() 函数，对于 Person(“will”, 82) 它是使用了默认的模板。

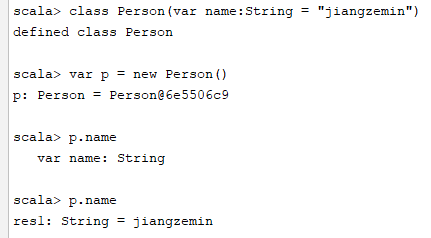
## objec对象

object 关键字声明的对象，它本身不是类定义，这个必须要搞清楚，它仅仅是一个静态数据集。





## 默认参数



## 类层次结构

