Scala 函数柯里化(Currying) Scala 字符串

Scala 闭包

闭包是一个函数，返回值依赖于声明在函数外部的一个或多个变量。

闭包通常来讲可以简单的认为是可以访问一个函数里面局部变量的另外一个函数。

如下面这段匿名的函数：

```
val multiplier = (i:Int) => i * 10
```

函数体内有一个变量 i，它作为函数的一个参数。如下面的另一段代码：

```
val multiplier = (i:Int) => i * factor
```

在 multiplier 中有两个变量：i 和 factor。其中的一个 i 是函数的形式参数，在 multiplier 函数被调用时，i 被赋予一个新的值。

然而，factor不是形式参数，而是自由变量，考虑下面代码：

```
var factor = 3
val multiplier = (i:Int) => i * factor
```

这里我们引入一个自由变量 factor，这个变量定义在函数外面。

这样定义的函数变量 multiplier 成为一个"闭包"，因为它引用到函数外面定义的变量，定义这个函数的过程是将这个自由变量捕获而构成一个封闭的函数。

完整实例

```
object Test {
   def main(args: Array[String]) {
      println( "muliplier(1) value = " +  multiplier(1) )
      println( "muliplier(2) value = " +  multiplier(2) )
   }
   var factor = 3
   val multiplier = (i:Int) => i * factor
}
```

运行实例 »

执行以上代码，输出结果为：

```
$ scalac Test.scala
$  scala Test
muliplier(1) value = 3
muliplier(2) value = 6
```

Scala 函数柯里化(Currying) Scala 字符串

点我分享笔记