Scala Trait(特征) Scala 正则表达式

Scala 模式匹配

Scala 提供了强大的模式匹配机制，应用也非常广泛。

一个模式匹配包含了一系列备选项，每个都开始于关键字 **case**。每个备选项都包含了一个模式及一到多个表达式。箭头符号 **=>** 隔开了模式和表达式。

以下是一个简单的整型值模式匹配实例：

```
object Test {
   def main(args: Array[String]) {
      println(matchTest(3))

   }
   def matchTest(x: Int): String = x match {
      case 1 => "one"
      case 2 => "two"
      case _ => "many"
   }
}
```

执行以上代码，输出结果为：

```
$ scalac Test.scala
$ scala Test
many
```

match 对应 Java 里的 switch，但是写在选择器表达式之后。即： **选择器 match {备选项}。**

match 表达式通过以代码编写的先后次序尝试每个模式来完成计算，只要发现有一个匹配的case，剩下的case不会继续匹配。

接下来我们来看一个不同数据类型的模式匹配：

```
object Test {
   def main(args: Array[String]) {
      println(matchTest("two"))
      println(matchTest("test"))
      println(matchTest(1))
      println(matchTest(6))

   }
   def matchTest(x: Any): Any = x match {
      case 1 => "one"
      case "two" => 2
      case y: Int => "scala.Int"
      case _ => "many"
```

```
    }
}
```

执行以上代码，输出结果为：

```
$ scalac Test.scala
$ scala Test
2
many
one
scala.Int
```

实例中第一个 case 对应整型数值 1，第二个 case 对应字符串值 two，第三个 case 对应类型模式，用于判断传入的值是否为整型，相比使用isInstanceOf来判断类型，使用模式匹配更好。第四个 case 表示默认的全匹配备选项，即没有找到其他匹配时的匹配项，类似 switch 中的 default。

使用样例类

使用了case关键字的类定义就是就是样例类(case classes)，样例类是种特殊的类，经过优化以用于模式匹配。

以下是样例类的简单实例:

```
object Test {
   def main(args: Array[String]) {
       val alice = new Person("Alice", 25)
    val bob = new Person("Bob", 32)
       val charlie = new Person("Charlie", 32)

    for (person <- List(alice, bob, charlie)) {
        person match {
            case Person("Alice", 25) => println("Hi Alice!")
            case Person("Bob", 32) => println("Hi Bob!")
            case Person(name, age) =>
               println("Age: " + age + " year, name: " + name + "?")
         }
      }
   }
   // 样例类
   case class Person(name: String, age: Int)
}
```

执行以上代码，输出结果为：

```
$ scalac Test.scala
$ scala Test
Hi Alice!
```

```
Hi Bob!
Age: 32 year, name: Charlie?
```

在声明样例类时，下面的过程自动发生了：

- 构造器的每个参数都成为val，除非显式被声明为var，但是并不推荐这么做；
- 在伴生对象中提供了apply方法，所以可以不使用new关键字就可构建对象；
- 提供unapply方法使模式匹配可以工作；
- 生成toString、equals、hashCode和copy方法，除非显示给出这些方法的定义。

← Scala Trait(特征) Scala 正则表达式 →