

Kotlin 分支



ChenME 关注

2018.07.26 14:26:10 字数 354 阅读 348

Kotlin 分支

1. if 分支表达式

在 Kotlin 中，由于没有了三目运算符（?:），所以就有了分支表达式，其效果可以替代三目运算符，甚至效果会更优

下面举个例子：

在程序开发中，一般分为开发环境、测试环境、正式环境，在这3个环境中会对应3个不同的接口地址，下面来模拟一下：

```
1 | const val ENVIRONMENT_DEV = 2 // 开发环境
2 | const val ENVIRONMENT_TEST = 1 // 测试环境
3 | const val ENVIRONMENT_ONLINE = 0 // 正式环境
4 |
5 | val projectEnvironment = ENVIRONMENT_DEV
6 |
7 | val PATH_HOST = if (ENVIRONMENT_ONLINE == projectEnvironment) {
8 |     "http://api.online.com"
9 | } else if (ENVIRONMENT_TEST == projectEnvironment) {
10 |     "http://api.test.com"
11 | } else {
12 |     "http://api.dev.com"
13 | }
14 |
15 | // http://api.dev.com
```

- 在 Java 中运用三目运算符，同样也可以实现，但是看起来就没那么直观了

```
1 | final String PATH_HOST = ENVIRONMENT_ONLINE == projectEnvironment ?
2 |     "http://api.online.com" :
3 |     (ENVIRONMENT_TEST == projectEnvironment ?
4 |         "http://api.test.com" :
5 |         "http://api.dev.com");
```

- 又或者是使用 `if else if else` 判断，但是，这样做了就没法将变量用 `final` 修饰了。

2. when 分支语句（表达式）

- 在 Kotlin 中，已经不支持 `switch` 语句了，除了 `switch` 支持的判断类型不丰富外（`char`, `byte`, `int`, `Character`, `Byte`, `Short`, `Integer`, `String`, `enum` (JDK1.7)），还有就是使用起来比较繁琐，例如每个分支结束后都要使用 `break` 做中断等；取而代之的是 `when`，下面一段示例代码：

1. 分支语句

```
1 | val number1 = 10
2 | val number2 = 10
3 | when (number1) {
4 |     is Int -> println("$number1 是一个 Int")
5 |     number2 -> println("$number1 与 number2 的值相等")
6 |     in 1..100 -> println("$number1 在 [1,100]")
7 | }
```

```
7 |         !in 1..100 -> println("$number1 不在 [1,100]")
8 |         else -> println("以上条件都不满足! ")
9 |     } // 10 是一个 Int
```

- 投过运行结果可知，当程序执行到符合条件的位置，会自动结束。

2. `when` 不仅限于语句，它也可以用表达式是上面：

- 还拿开发环境的例子来说，也可以用 `when` 表达式实现

```
1 | val PATH_HOST = when (projectEnvironment) {
2 |     ENVIRONMENT_DEV -> "http://api.online.com"
3 |     ENVIRONMENT_TEST -> "http://api.test.com"
4 |     else -> "http://api.dev.com"
5 | }
```



1人点赞 >



📖 Kotlin

