

Scala 数据类型

Scala 与 Java有着相同的数据类型，下表列出了 Scala 支持的数据类型：

数据类型	描述
Byte	8位有符号补码整数。数值区间为 -128 到 127
Short	16位有符号补码整数。数值区间为 -32768 到 32767
Int	32位有符号补码整数。数值区间为 -2147483648 到 2147483647
Long	64位有符号补码整数。数值区间为 -9223372036854775808 到 9223372036854775807
Float	32 位, IEEE 754 标准的单精度浮点数
Double	64 位 IEEE 754 标准的双精度浮点数
Char	16位无符号Unicode字符, 区间值为 U+0000 到 U+FFFF
String	字符序列
Boolean	true或false
Unit	表示无值，和其他语言中void等同。用作不返回任何结果的方法的结果类型。Unit只有一个实例值，写成()。
Null	null 或空引用
Nothing	Nothing类型在Scala的类层级的最低端；它是任何其他类型的子类型。
Any	Any是所有其他类的超类
AnyRef	AnyRef类是Scala里所有引用类(reference class)的基类

上表中列出的数据类型都是对象，也就是说scala没有java中的原生类型。在scala是可以对数字等基础类型调用方法的。

Scala 基础字面量

Scala 非常简单且直观。接下来我们会详细介绍 Scala 字面量。

整型字面量

整型字面量用于 Int 类型，如果表示 Long，可以在数字后面添加 L 或者小写 l 作为后缀。：

```
0
035
```

```
21
0xFFFFFFFF
0777L
```

浮点型字面量

如果浮点数后面有f或者F后缀时，表示这是一个Float类型，否则就是一个Double类型的。实例如下：

```
0.0
1e30f
3.14159f
1.0e100
.1
```

布尔型字面量

布尔型字面量有 true 和 false。

符号字面量

符号字面量被写成：**<标识符>**，这里 **<标识符>** 可以是任何字母或数字的标识（注意：不能以数字开头）。这种字面量被映射成预定义类scala.Symbol的实例。

如：符号字面量 **'x'** 是表达式 **scala.Symbol("x")** 的简写，符号字面量定义如下：

```
package scala
final case class Symbol private (name: String) {
  override def toString: String = "'" + name
}
```

字符字面量

在 Scala 字符变量使用单引号 **'** 来定义，如下：

```
'a'
'\u0041'
'\n'
'\t'
```

其中 \ 表示转义字符，其后可以跟 **u0041** 数字或者 **\r\n** 等固定的转义字符。

字符串字面量

在 Scala 字符串变量使用双引号 **"** 来定义，如下：

```
"Hello,\nWorld!"
"菜鸟教程官网：www.runoob.com"
```

多行字符串的表示方法

多行字符串用三个双引号来表示分隔符，格式为：`""" ... """`。

实例如下：

```
val foo = """菜鸟教程
www.runoob.com
www.w3cschool.cc
www.runnoob.com
以上三个地址都能访问"""
```

Null 值

空值是 `scala.Null` 类型。

`Scala.Null`和`scala.Nothing`是用统一的方式处理Scala面向对象类型系统的某些"边界情况"的特殊类型。

`Null`类是`null`引用对象的类型，它是每个引用类（继承自`AnyRef`的类）的子类。`Null`不兼容值类型。

Scala 转义字符

下表列出了常见的转义字符：

转义字符	Unicode	描述
<code>\b</code>	<code>\u0008</code>	退格(BS)，将当前位置移到前一系列
<code>\t</code>	<code>\u0009</code>	水平制表(HT)（跳到下一个TAB位置）
<code>\n</code>	<code>\u000a</code>	换行(LF)，将当前位置移到下一行开头
<code>\f</code>	<code>\u000c</code>	换页(FF)，将当前位置移到下页开头
<code>\r</code>	<code>\u000d</code>	回车(CR)，将当前位置移到本行开头
<code>\"</code>	<code>\u0022</code>	代表一个双引号(")字符
<code>\'</code>	<code>\u0027</code>	代表一个单引号(')字符
<code>\\</code>	<code>\u005c</code>	代表一个反斜线字符 '\'

0 到 255 间的 `Unicode` 字符可以用一个八进制转义序列来表示，即反斜线"`\`"后跟 最多三个八进制。

在字符或字符串中，反斜线和后面的字符序列不能构成一个合法的转义序列将会导致 编译错误。

以下实例演示了一些转义字符的使用：

```
object Test {
  def main(args: Array[String]) {
    println("Hello\tWorld\n\n" );
  }
}
```

[运行实例 »](#)

执行以上代码输出结果如下所示：

```
$ scalac Test.scala
$ scala Test
Hello    World
```

[← Scala 基础语法](#)[Scala 变量 →](#)[📝 点我分享笔记](#)