

数据类型和运算符

一、注释

1.单行注释 将双斜线 (//) 放在需要注释的内容之前就可以了 //这是单行注释 2.多行注释 使用"/*"和"*/"将程序中需要注释的内容包含起来 /*这是多行注释*/ 3.文档注释 略

二、标识符和关键字

1.标识符规则

- 标识符可以由字母、数字、下划线和美元符组成，其中数字不能打头
- 标识符不能是Java关键字和保留字，但可以包含关键字和保留字
- 标识符不能包含空格
- 标识符只能包含美元符，不能包含其他特殊字符

2.Java所有的关键字都是小写的，TRUE、FALSE、NULL都不是Java关键字，Java一共包含了50个关键字

三、数据类型分类

Java是强类型语言，包含两个定义：（1）所有变量必须先声明后使用 （2）指定类型的变量只能接受类型与之匹配的值

Java中整数值有4种表示方式：十进制、二进制、八进制、十六进制，二进制整数以0b或0B开头，八进制以0开头，十六进制以0x或0X开头，10-15用a~f不区分大小写表示 原码、反码、补码：正数的原码、反码和补码完全相同；负数的反码是原码符号位不变，其余位按位取反，补码是其反码加1，负数由补码求原码为补码按位求反再加1 Java支持两种数据类型：基本类型和引用类型

1.基本数据类型

- 整型
 - byte: 8位
 - short: 16位
 - int: 32位
 - long: 64位

默认情况下，一个整数值是int类型，但有两种特殊情况:

- 直接把一个整数值（在byte或short表数范围内）赋给一个byte或short变量，系统自动把这个整数值当作byte或short来处理
- 如果使用一个整数值（超出了int类型的表数范围）时，系统不会自动把该数当作long类型来处理，这时需要在整数值后增加l或L后缀
- 字符型 字符型通常用于表示单个字符，字符型值必须使用单引号括起来。字符型值由三种表示方式：
 - 直接通过单个字符来指定字符型值，如'a'、'9'等
 - 通过转义字符表示特殊字符型值，如'\n'、'\t'等
 - 直接使用Unicode值来表示字符型值，格式是'\uXXXX'，XXXX代表一个十六进制整数

Java中常用转义字符如下表：

转义字符	说明
\b	退格符
\n	换行符
\r	回车符
\t	制表符
\'	单引号
\"	双引号
\\	反斜线

- 字符型值相当于一个16位的无符号整数，表数范围为0-65535，可以进行等数学运算
 - 如果把0-65535范围内的一个int整数赋给char类型变量，系统自动把该整数当成char类型来处理
 - Java没有提供字符串的基本数据类型，而是通过String类来表示字符串，字符串要使用双引号括起来
 - 浮点型 Java的浮点型有两种：float和double，他们都具有固定的表述范围和字段长度 Java使用二进制科学计数法来表示浮点数，因此可能不能精确表示一个浮点数，只是double类型精度更高。如果要精确表示浮点数，可以使用BigDecimal
 - float：单精度浮点数，32位
 - double：双精度浮点数，64位
- Java浮点数表示由两种形式：十进制形式和科学计数法形式，只有浮点类型的值才能使用科学计数法表示，e/E不区分大小写 Java浮点类型值默认是double类型，表示float类型需要后缀f或F 只有浮点数除以0才能得到正无穷大或负无穷大，Java自动将0当作0.0处理，整数除以0会报错 当程序中数值位特别多时，可以使用下画线任意分隔数值而不改变数值大小
- 布尔型 布尔型只有一个boolean类型，表示逻辑上的真假，boolean类型的数值只能是true或false，不能用0或非0来表示，其他数据类型的值也不能转换为boolean类型 Java没有规定boolean类型占用多少存储空间，虽然可以只占用一位，但是通常占用一个最小内存单元：8位

2.基本数据类型转换

- 自动类型转换 当把一个表数范围小的数值或变量直接赋给另一个表数范围大的变量时，系统可以进行自动类型转换 byte->short->int->long->float->double char->int->long ->float->double 当把任何基本类型的值与字符串值进行连接时，基本类型值自动类型转换为字符串类型，如果希望把基本类型值转换为字符串类型，可以将其与空字符串进行连接，注意连接的先后顺序 数据类型转换：
DataTypeChange.java
- 强制类型转换 当把箭头右边的类型转换为箭头左边的类型时，需要进行强制类型转换，强制类型转换直接截断多余的位，容易造成数据丢失。 字符串不能直接转换为基本类型，可以通过基本类型的包装类实现。
- 表达式类型的自动提升 算术表达式中的自动类型提升：
 - 所有的byte、short、char类型自动提升为int类型

- 整个算法表达式的数据类型自动提升到与表达式中最高等级操作数同样的类型
- 表达式类型严格保持与最高等级操作数一样的类型

3.直接量 并不是所有的数据类型都能定义直接量，直接量包括：int、long、float、double、char、String、boolean、null 其中，null是一种特殊类型，它只有一个值：null，这个直接量可以赋给任何引用类型的变量，表示保存的地址为空。当程序第一次使用某个字符串直接量时，Java会使用常量池来缓存该直接量，如果程序后面需要使用该直接量，Java会使用该常量池中的字符串直接量

4.运算符 Java运算符包括：算术运算符、赋值运算符、比较运算符、逻辑运算符、位运算符、类型相关运算符

- 算术运算符

- +：加法运算符、字符串连接运算符
- -：减法运算符
- *：乘法运算符
- /：除法运算符 除法运算符的两个操作数都是整数时，除数不能是0，否则引发除以0异常
- %：求余运算符 求余运算符的两个操作数都是整数时，第二个数不能是0，否则引发除以0异常
- ++：自加运算符
- --：自减运算符

- 赋值运算符 赋值表达式是有值的，其值为右边的值，因此它支持连续赋值 推荐使用扩展过后的赋值运算符，其更加健壮

- +=
- -=
- *=
- /=
- %=
- &=
- |=
- ^=
- <<=
- >>=
- >>>=

- 比较运算符

- >
- <
- >=
- <=
- == 基本类型的值、变量不能与引用类型的变量、值使用==比较；boolean类型的值、变量不能与其他任何类型的值、变量使用==比较；两个引用类型之间没有父子继承关系，他们的变量也不能使用==比较
- !=

- 逻辑运算符

- &&：短路与
- &
- ||：短路或
- |
- !
- ^

- 位运算符
 - &
 - |
 - ~
 - ^
 - <<:左移运算符
 - >>:右移运算符
 - >>>:无符号右移运算符 一般来说，位运算符只能操作整数类型的变量或值
- 三目运算符 三目运算符只有一个：?: 三木运算符不支持多条语句，三木运算符可嵌套