

第二章 数据类型与运算符（一）

常量

在程序运行中不能被修改的量
Java 语言中使用关键字 final 来修饰变量使其成为常量

标识符

标识符就是程序员自己规定的具有特定含义的词，比如类名称，属性名称，变量名等
指系统规定的具有特殊意义的一些标识符，又称作保留字

变量

变量一次来源于数学
程序运行过程中其值可以发生变化或者能被修改
计算机当中，变量相当于一定存储空间中的表示

计算机内存

RAM空间【随机访问存储空间】
寄存器，暂存CPU需要的数据
连续的地址表示，eg. 0xFFED A082—0xFFED FFFF

变量如何定义【声明】

根据不同数据类型的需求在计算机内存中开辟相应的存储空间

变量的三要素

- 1.变量的类型【数据类型】
- 2.变量名【有意义的标识符】
- 3.变量的值【存储到相应内存空间的可修改的值】

变量的使用

内存【寄存器】的地址是不方便记忆和使用的
直接使用变量名访问变量的值

变量名的命名规则

只能由字母、数字、下划线和美元符号\$组成
变量名的首字母不能是数字
变量名不能与系统的关键字重复

有符号数&无符号数表示

正负数表示
1个bit位表示符号其余位表示数值大小
注意防止数据溢出和截断

数值型

- 1. 整型 byte [1个字节] short [2个字节] int [4个字节] long [8个字节]
- 2.浮点型 float [4个字节] double [8个字节] 默认情况下的浮点型常量是double型的

非数值类型

- 1. char [2个字节 兼容东亚字符集/宽字符]
- 2.String 字符串类型/字符串类【它是引用数据类型，指向对象的引用】

变量在使用之前要赋值或者要有明确的指向

Java语言中的数据类型

借助temp中间值实现数据的交换

算术运算符

加减乘除、取余数【模数】、自增、自减
+ - * / % ++ --
++与++i -与--j的区别
复合赋值运算符：算术运算符与赋值运算符的组合
+= -= *= /= %= 等效使用

赋值运算符

使用符号 =
变量名=表达式
将数据赋给可变的左值

表达式的概念

操作符和操作数的组合

数据类型的转换

自动类型转换

- 1.相互转换的两种数据类型要兼容【整型和浮点型兼容】
- 2.目标类型大于源类型/小数据类型往大的数据类型转换
- 3.注意char型在实际存储中是以字符的ASCII码值存储的，可以自动向整型转换
计算机自动完成，无须程序做特殊处理

强制类型转换

(类型名) 表达式
为了使得计算的表达式兼容，需要手动进行特殊处理和转换

Scanner类的使用

- 1.使用import指令导入java.util.Scanner 类
- 2.实例化Scanner类对象 eg. Scanner input=new Scanner(System.in)
- 3.使用Scanner可以实现从控制台的数据输入
常用方法 hasNext() next() nextInt() nextDouble()