



河南中医药大学信息技术学院（智能医疗行业学院）智能医学工程专业《计算机程序设计》课程

第06章：常量与变量

黄子杰

河南中医药大学信息技术学院（智能医疗行业学院）与河南方和信息科技有限公司 联合建设
河南中医药大学信息技术学院智能医疗教研室

<https://aitcm.hactcm.edu.cn>

2025/2/16

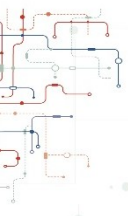
本章概要

概述：

常量：在程序中固定不变的量，一旦被定义，它的值就不能再被改变。

变量：在程序中存在大量的数据来代表程序的状态，其中有些数据的值在程序的运行过程中会发生改变，这种数据被成为变量。在程序中通过改变变量的值来实现程序的逻辑功能。





6.1 Java符号

- 运算符

算术运算符：包括加(+)、减(-)、乘(*)、除(/)、求余(%）、自增(++）、自减(--)。

赋值运算符：包括等号(=)、加等于(+=)、减等于(-=)、乘等于(*=)、除等于(/=)、求余等于(%=)。

==比较运算符：包括等于(=)、不等于(!=)、大于(>)、小于(<)、大于等于(>=)、小于等于(<=)。

逻辑运算符：包括与(&&)、或(||)、非(!)。

位运算符：包括按位与(&)、按位或(|)、异或(^)、左移()、右移()。

三元运算符：条件表达式？：

- 标识符

标识符的命名规则如下：

由字母、数字、美元符号(\$)和下划线(_)组成。

不能以数字开头。

不能使用Java关键字。

区分大小写。

可以包含Unicode字符集中符号大于0xC0的所有符号。



6.1 Java符号

注意事项

- 注意事项

算术运算符：整数除法会舍弃小数部分，如果需要精确结果，可以进行类型转换。

逻辑运算符：逻辑与(&&)和逻辑或(||)被称为“短路”运算符，当第一个操作数已经确定结果时，不会继续计算第二个操作数。

位运算符：基于二进制计算，常用于处理二进制数据。

三元运算符：条件表达式? 表达式1: 表达式2; 如果条件表达式为true，结果为表达式1；否则为表达式2。

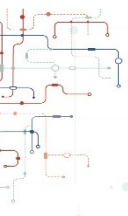
标识符命名：遵循命名规范，避免使用保留字和关键字。

6.1.1 标识符

- 概述：

标识符是Java程序中用于命名变量、方法、类、接口等的名字。标识符可以由字母、数字、下划线 _ 和美元符号 \$ 组成，但不能以数字开头。





6.1.1 标识符

- 详细说明：

合法标识符：由字母、数字、下划线和美元符号组成，且不能以数字开头。

命名规则：Java标识符对大小写敏感，建议使用有意义的名字，如 `studentName` 而非 `sn`。

示例：`int age;` 中 `age` 是一个标识符。

- 生活举例：

标识符：就像给你的书籍或文件取名字，确保名称具有意义且不重复。



6.1.1 标识符

- 面试题示例：

问：Java中的标识符有什么规则？

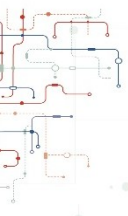
答：标识符可以包含字母、数字、下划线和美元符号，但不能以数字开头，并且对大小写敏感。

6.1.2 关键字

- 概述：

关键字是Java语言中预定义的保留字，具有特殊的语法意义，不能用作标识符。





6.1.2 关键字

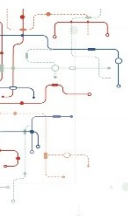
- 详细说明：

示例关键字：int、class、public、static、void 等。

使用：关键字用于定义变量类型、控制程序流程、声明类和方法等。

生活举例：

关键字：就像编程语言中的“规定动作”，就像语法规则在书写文章时必须遵守的规范。



6.1.2 关键字

- 面试题示例：

问：Java中的关键字是什么？举例说明。

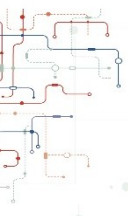
答：关键字是Java语言的保留字，用于定义语言的基本结构。例如，`class` 用于定义类，`public` 用于声明访问权限。

6.1.3 分隔符

- 概述：

分隔符是用于分隔代码中不同部分的符号，如语句结束符、块界限等。





6.1.3 分隔符

- 详细说明：

示例分隔符：

分号 ;：用于结束一条语句。

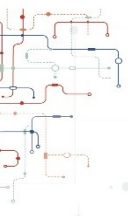
大括号 {}：用于定义代码块。

小括号 ()：用于方法调用和控制语句。

中括号 []：用于数组声明。

- 生活举例：

分隔符：就像书中的标点符号，用于分隔句子、段落，使文本更清晰易读。



6.1.3 分隔符

- 面试题示例：

问：Java中的常见分隔符有哪些？它们的作用是什么？

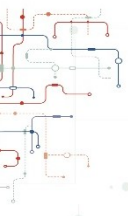
答：常见的分隔符有分号（;）用于结束语句，大括号（{}）用于定义代码块，小括号（()）用于方法调用和控制语句。

6.1.4 注释

- 概述：

注释是代码中的非执行部分，用于解释代码的功能或提供额外信息。注释不会影响程序的执行。





6.1.4 注释

- 详细说明：

单行注释：// 用于注释一行内容。

多行注释：/* ... */ 用于注释多行内容。

文档注释：/** ... */ 用于生成API文档，通常用于类和方法的说明。

6.1.4 注释

- 代码示例：



eg6.1.4.txt

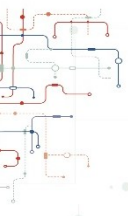




6.1.4 注释

- 生活举例：

注释：就像在文章中添加脚注或边注，提供额外的解释和信息而不影响主要内容。

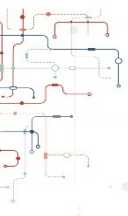


6.1.4 注释

- 面试题示例：

问：Java中注释的类型有哪些？它们的用途是什么？

答：Java中有单行注释、多行注释和文档注释。单行注释用于注释一行代码，多行注释用于注释多行内容，文档注释用于生成API文档。



6.1.5 常量

- 概述：

常量是指在程序运行过程中其值不会改变的量。常量在Java中使用final关键字来定义。

6.1.5 常量

- 详细说明：

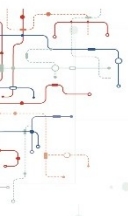
定义常量：使用final关键字定义常量，常量的值在初始化后不能更改。

示例代码：



eg6.1.5.txt

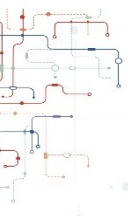




6.1.5 常量

- 生活举例：

常量：就像物理常数，例如地球的重力加速度 9.8 m/s^2 ，它在一定条件下是不变的。



6.1.5 常量

- 面试题示例：

问：如何在Java中定义常量？为什么要使用常量？

答：使用final关键字定义常量，常量的值在初始化后不可更改。常量用于表示固定值，增加代码的可读性和维护性。

6.1.6 变量

- 概述：

变量是程序中可以存储和修改数据的空间。在Java中，变量分为局部变量和实例变量。



6.1.6 变量

- 详细说明：

局部变量：在方法或代码块中定义，作用范围限于其所在的方法或代码块。

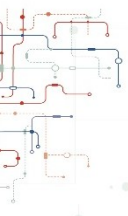
实例变量：在类中定义，作用范围为整个类的实例。

代码示例：



eg6.1.6.txt





6.1.6 变量

- 生活举例：

变量：就像书桌上的容器，可以放置不同的物品，容器的内容可以随时更换。

- 面试题示例：

问：Java中变量的类型有哪些？它们的作用是什么？

答：Java中变量分为局部变量和实例变量。局部变量在方法内定义，仅在方法中有效；实例变量在类中定义，所有实例共享该变量。



6.2 表达式与运算符

- 概述:

在Java中，运算符和表达式是编程的基础，它们用于执行各种计算和操作。Java支持多种类型的运算符，包括算术运算符、赋值运算符、比较运算符和逻辑运算符等。

6.2.1 算术运算符

- 算术运算符：

Java中的算术运算符用于执行基本的数学运算，包括加法(+)、减法(-)、乘法(*)、除法(/)和取模(%)。这些运算符用于将两个数字进行相加、相减、相乘、相除以及计算余数。例如， $a + b$ 表示将变量a和b的值相加， $a - b$ 表示将变量a的值减去变量b的值，以此类推。

代码示例：



eg6.2.1.txt

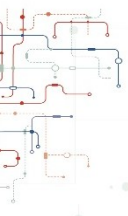




6.2.1 算术运算符

- 生活举例：

算术运算符：就像我们在购物时计算总价、找零等。



6.2.1 算术运算符

- 面试题示例：

问：Java中的算术运算符有哪些？它们的作用是什么？

答：算术运算符包括加法（+）、减法（-）、乘法（*）、除法（/）和取余（%），用于进行基本的数学运算。



6.2.2 关系运算符

- 概述： 关系运算符用于比较两个值的大小关系，结果为布尔值（true 或 false）。

6.2.2 关系运算符

- 详细说明：

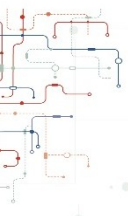
示例运算符：==（等于）、!=（不等于）、>（大于）、<（小于）、>=（大于等于）、<=（小于等于）。

代码示例：



eg6.2.2.txt





6.2.2 关系运算符

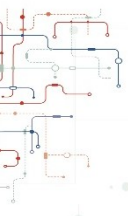
- 生活举例：

关系运算符：就像比较两个物品的价格，确定哪个更便宜。

- 面试题示例：

问：Java中的关系运算符有哪些？它们的作用是什么？

答：关系运算符用于比较两个值的关系，如等于（==）、不等于（!=）、大于（>）、小于（<）等。



6.2.3 逻辑运算符

- 逻辑运算符：

逻辑运算符用于组合和操作布尔值。Java中的逻辑运算符包括逻辑与(&&)、逻辑或(||)。逻辑与运算符当两个操作数都为true时返回true，否则返回false。逻辑或运算符当至少一个操作数为true时返回true，否则返回false。

6.2.3 逻辑运算符

- 详细说明：

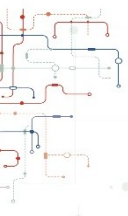
示例运算符：&&（逻辑与）、||（逻辑或）、！（逻辑非）。

代码示例：



eg6.2.3.txt





6.2.3 逻辑运算符

- 生活举例：

逻辑运算符：就像在做决策时考虑多个条件，例如“如果天晴并且气温高，就去郊游”。

- 面试题示例：

问：Java中的逻辑运算符有哪些？它们如何工作？

答：逻辑运算符包括逻辑与（&&）、逻辑或（||）和逻辑非（!）。&&在两个操作数都为真时结果为真，||在至少一个操作数为真时结果为真，!用于取反布尔值。



6.2.4 位运算符

- 概述：

位运算符用于对整数类型的二进制位进行操作。

6.2.4 位运算符

- 详细说明：

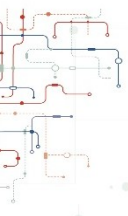
示例运算符：&（按位与）、|（按位或）、^（按位异或）、~（按位取反）、<<（左移）、>>（右移）、>>>（无符号右移）。

代码示例：



eg6.2.4.txt

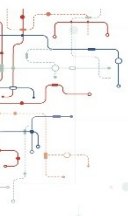




6.2.4 位运算符

- 生活举例：

位运算符：就像在处理机器上的开关，控制每个开关的位置（开或关）。

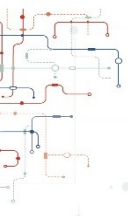


6.2.4 位运算符

- 面试题示例：

问：Java中的位运算符有哪些？它们的作用是什么？

答：位运算符包括按位与（&）、按位或（|）、按位异或（^）、按位取反（~）、左移（<<）、右移（>>）、无符号右移（>>>），用于对整数的二进制位进行操作。



6.2.5 赋值组合运算符

- 概述：

赋值组合运算符用于将一个表达式的结果赋值给一个变量，并同时
进行其他运算。

6.2.5 赋值组合运算符

- 详细说明：

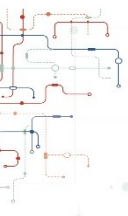
示例运算符：+=（加并赋值）、-=（减并赋值）、*=（乘并赋值）、/=（除并赋值）、%=（取余并赋值）。

代码示例：



eg6.2.5.txt

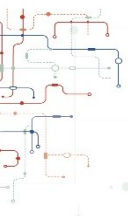




6.2.5 赋值组合运算符

- 生活举例：

赋值组合运算符：就像在账户余额上进行操作，比如“余额增加500元”就相当于“余额 = 余额 + 500元”。

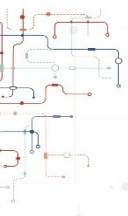


6.2.5 赋值组合运算符

- 面试题示例：

问：Java中的赋值组合运算符有哪些？它们的作用是什么？

答：赋值组合运算符包括+=、-=、*=、/=和%=，用于将运算结果赋值给变量。



6.2.6 其他运算符

- 概述：

其他运算符包括三元运算符和实例判断运算符。

字符串连接运算符 (+)

类型转换运算符

```
double d = 3.14;
```

```
int i = (int)d; // 将double类型的d转换为int类型，结果为3
```

instanceof运算符,这个运算符用于测试一个对象是否是特定类的一个实例。

```
String s = "Hello";
```

```
boolean isString = s instanceof String; // 结果为true
```

6.2.6 其他运算符

- 详细说明：

三元运算符：`?:`，用于简化if-else语句。

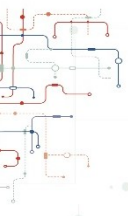
实例判断运算符：`instanceof`，用于判断对象是否是某个类的实例。

代码示例：



eg6.2.6.txt

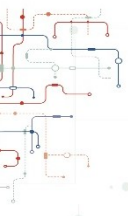




6.2.6 其他运算符

- 生活举例：

其他运算符：就像在购物时选择商品的折扣（? :），或者判断某个物品是否属于某个类别（instanceof）。



6.2.6 其他运算符

- 面试题示例：

问：Java中的三元运算符是什么？如何使用？

答：三元运算符`?:`用于根据条件选择两个值中的一个。格式为
条件 `?` 值1 `:` 值2。

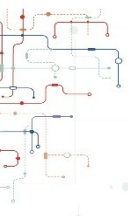
问：`instanceof`运算符的作用是什么？

答：`instanceof`运算符用于判断一个对象是否是某个类的实例或某个接口的实现。

6.2.7 运算符优先级

- 在Java中，运算符的优先级决定了表达式中各个部分被评估的顺序。以下是一个简要的优先级概述，从高到低排列：

- 后缀运算符**：如方法调用、数组访问、后缀自增自减（`x++`, `x--`）。
- 一元运算符**：如前缀自增自减（`++x`, `--x`）、正号（`+`）、负号（`-`）、位非（`~`）、逻辑非（`!`）、类型转换。
- 乘法、除法、取模运算符**：`*`、`/`、`%`，这些运算符按从左到右的顺序执行。
- 加法、减法运算符**：`+`、`-`，也在乘法、除法和取模之后，从左到右执行。
- 位移运算符**：`<<`（左移）、`>>`（有符号右移）、`>>>`（无符号右移），它们在加法和减法之后执行。
- 关系运算符**：`<`、`<=`、`>`、`>=`、`instanceof`，用于比较两个值。
- 相等运算符**：`==`、`!=`，检查两个值是否相等或不等。
- 按位与运算符**：`&`，对两个整数的二进制表示进行按位与操作。
- 按位异或运算符**：`^`，对两个整数的二进制表示进行按位异或操作。
- 按位或运算符**：`|`，对两个整数的二进制表示进行按位或操作。



6.2.7 运算符优先级

11 逻辑与运算符：&&，如果第一个操作数为true，则计算第二个操作数；如果第一个操作数为false，则结果为false，不再计算第二个操作数。

12 逻辑或运算符：||，如果第一个操作数为true，则结果为true，不再计算第二个操作数；如果第一个操作数为false，则计算第二个操作数。

13 条件运算符：? :，也称为三元运算符，根据条件选择两个值中的一个。

14 赋值运算符：=、+=、-=、*=、/=、%=、&=、^=、|=、<<=、>>=、>>>=，用于给变量赋值。



本章重点

- Java符号

标识符,关键字,分隔符,注释

- 表达式与运算符

算术运算符,关系运算符,逻辑运算符,位运算符,赋值组合运算符

- 运算符优先级



本章作业

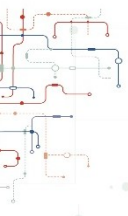
作业一：基础概念与符号应用

请编写一个Java程序，该程序需要包含以下内容：

- 1 使用有效的标识符（变量名、类名等）声明至少两个变量，其中一个为整型（int），另一个为字符串型（String）。
- 2 使用至少两个Java关键字（如public、static、void等）来声明一个方法。
- 3 在该方法中，使用至少两种分隔符（如逗号、分号;）来编写一条包含算术运算符（如+、-）的表达式，并计算两个整型变量的和。
- 4 在程序中添加单行注释和多行注释，分别解释变量声明的目的和方法的功能。
- 5 最后，在main方法中调用你声明的方法，并打印出计算的结果。

要求：

- 1 确保程序能够编译并正确运行。
- 2 注释内容清晰、准确，有助于理解代码。



本章作业

作业二：运算符综合应用

设计一个Java程序，该程序要求用户输入两个整数，并完成以下任务：

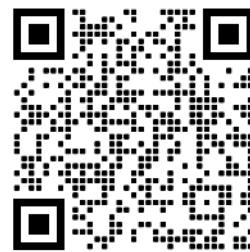
- 1 使用算术运算符计算这两个数的和、差、积、商（注意处理除数为0的情况）。
- 2 使用关系运算符判断这两个数的大小关系，并打印出相应的比较结果（如“第一个数大于第二个数”）。
- 3 结合逻辑运算符，编写一个条件判断，如果第一个数大于第二个数并且它们的和是偶数，则打印“满足条件”。
- 4 在整个程序中合理使用注释，说明各个部分的功能和可能遇到的特殊情况处理。

要求：

- 1 用户输入应使用Scanner类来实现。
- 2 程序中应包含对除数为0的异常处理。
- 3 逻辑清晰，代码结构良好，易于理解。

信创智能医疗系统研发课程体系

河南中医药大学信息技术学院（智能医疗行业学院）



河南中医药大学信息技术学院（智能医疗行业学院）智能医疗教研室

河南中医药大学医疗健康信息工程技术研究所