7.1.2024第五天 书：java核心技术卷一

今日目标：学完变量后 今天学习一下变量之间的关系 算数运算符与数学函数之间的关系 还有类型转化

第三章

3.5运算符

加减乘除 Math.round()------四舍五入

当用一个二元运算符连接两个值时(例如 n+f,n是整数，f是浮点数)，先要将两个操作数转换为同一种类型，然后再进行计算。

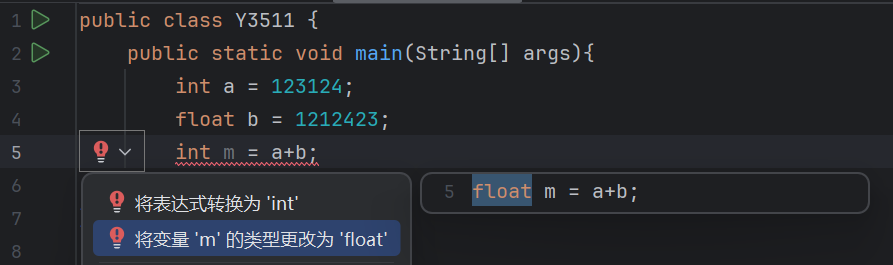
如果两个操作数中有一个是double 类型，另一个操作数就会转换为double类型。

否则，如果其中一个操作数是float类型，另一个操作数就转换为float类型。

否则，如果其中一个操作数是long 类型，另一个操作数就会转换为long 类型。

否则，两个操作数都被转为int类型。

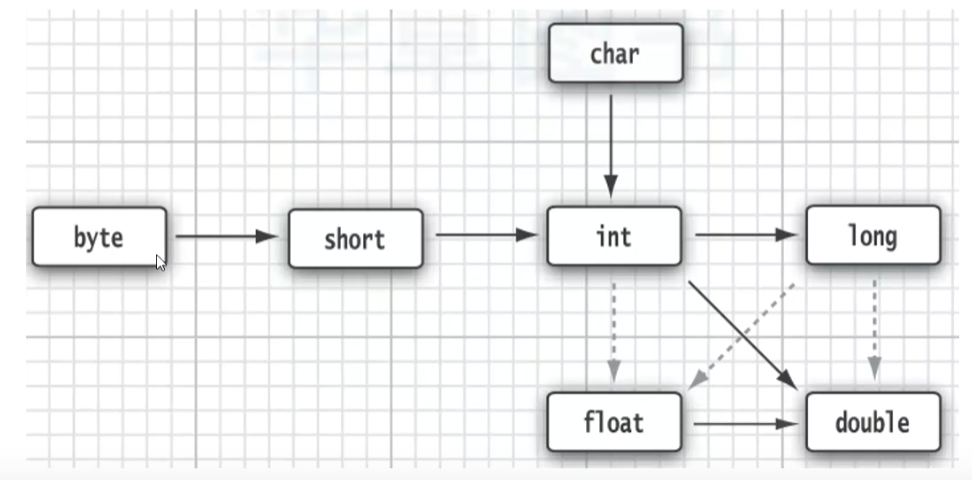
错误的代码如下



正确的代码如下：

public class Y3511 {  
 public static void main(String[] args){  
 int a = 123124;  
 float b = 1212423;  
 float m = a+b;  
 }  
}

3.5.1算术运算与数学函数



实线表示无信息丢失的转化；3个虚线箭头表示可能有精度丢失的转换。

public class Y351 {  
 public static void main(String[] args){  
 int a = 15;  
 int b = 10;  
 System.*out*.println(a+b);  
 System.*out*.println(a-b);  
 System.*out*.println(a\*b);  
 System.*out*.println(a/b);  
 System.*out*.println(a%b);  
 }  
}

运算结果：25，5，150，1，5

例如15/10=1；15.0/10=1.5；

3.5.2 类型转化

在必要的时候，int类型的值将会自动转换为double 类型。

但另一方面，有时也需要将double类型转换成int 类型。

在Java中，允许进行这种数值转换，不过当然可能会丢失一些信息。这种可能损失信息的转换要通过强制类型转换(cast)来完成。强制类型转换的语法格式实在圆括号中指定想要转换的目标类型，后面紧跟待转换的变量名。例如:

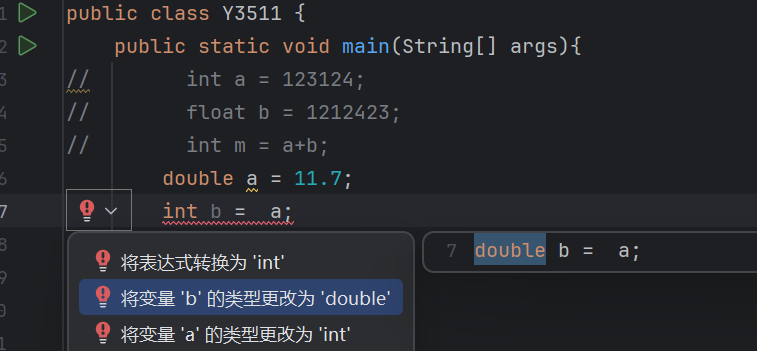
double x=9.997

int nx =(int)x;

这样，变量nx 的值为9，因为强制类型转换通过截断小数部分将浮点值转换为整型如果想舍入(round)一个浮点数来得到最接近的整数(大多数情况下，这种操作更有用)，可以使用Math.round

double x= 9.997:

int nx =(int) Math.round(x);

比如看下面：这个是错误的

public class Y3511 {  
 public static void main(String[] args){  
// int a = 123124;  
// float b = 1212423;  
// int m = a+b;  
 double a = 11.7;  
 int b = (int) a;  
 System.*out*.println(a);  
 System.*out*.println(b);  
 }  
}

这个上面是正确的代码 输出的a等于11 .7 b等于11

Byte1字节 -128~127 127 -128 -127…-1 0 1 2

byte a =126;  
byte a1=(byte) (a+1);  
byte a2=(byte) (a1+1);  
byte a3=(byte) (a2+1);  
System.*out*.println(a);  
System.*out*.println(a1);  
System.*out*.println(a2);  
System.*out*.println(a3);

输出分别为126 127 -128 -127