**自动类型转换**

自动类型转换

整型、实型（常量）、字符型数据可以混合运算。运算中，不同类型的数据先转化为同一类型，然后进行运算。

转换从低级到高级。

低 ------------------------------------> 高

byte,short,char—> int —> long—> float —> double

数据类型转换必须满足如下规则：

1. 不能对boolean类型进行类型转换。

2. 不能把对象类型转换成不相关类的对象。

3. 在把容量大的类型转换为容量小的类型时必须使用强制类型转换。

4. 转换过程中可能导致溢出或损失精度，例如：

int i =128;

byte b = (byte)i;

因为 byte 类型是 8 位，最大值为127，所以当 int 强制转换为 byte 类型时，值 128 时候就会导致溢出。

5. 浮点数到整数的转换是通过舍弃小数得到，而不是四舍五入，例如：

(int)23.7 == 23;

(int)-45.89f == -45

自动类型转换

必须满足转换前的数据类型的位数要低于转换后的数据类型，例如: short数据类型的位数为16位，就可以自动转换位数为32的int类型，同样float数据类型的位数为32，可以自动转换为64位的double类型。

实例

public class ZiDongLeiZhuan{

public static void main(String[] args){

char c1='a';//定义一个char类型

int i1 = c1;//char自动类型转换为int

System.out.println("char自动类型转换为int后的值等于"+i1);

char c2 = 'A';//定义一个char类型

int i2 = c2+1;//char 类型和 int 类型计算

System.out.println("char类型和int计算后的值等于"+i2);

}

}

运行结果为:

char自动类型转换为int后的值等于97

char类型和int计算后的值等于66

解析：c1 的值为字符 a ,查 ASCII 码表可知对应的 int 类型值为 97， A 对应值为 65，所以 i2=65+1=66。

强制类型转换

1. 条件是转换的数据类型必须是兼容的。

2. 格式：(type)value type是要强制类型转换后的数据类型 实例：

实例

public class QiangZhiZhuanHuan{

public static void main(String[] args){

int i1 = 123;

byte b = (byte)i1;//强制类型转换为byte

System.out.println("int强制类型转换为byte后的值等于"+b);

}

}

运行结果：

int强制类型转换为byte后的值等于123

隐含强制类型转换

1、 整数的默认类型是 int。

2. 小数默认是 double 类型浮点型，在定义 float 类型时必须在数字后面跟上 F 或者 f。

这一节讲解了 Java 的基本数据类型。下一节将探讨不同的变量类型以及它们的用法。