

## 实验1 托运行李

## 1. 相关知识点

类型转换运算符是单目运算符,其运算所得数据的类型可能不同于操作元的类型。类型转换运算符不改变操作元本身的类型,操作元经常是数值型数据。例如,(float)12的结果是12.0f,(int)45.98的结果是45,(double)(int)68.89的结果是68.0。

## 2. 实验目的

本实验的目的是让学生掌握类型转换运算符。

## 3. 实验要求

(1) 火车在计算托运行李费用时以 kg 为单位计算费用(12 元/kg),忽略重量中的小数部分,即忽略不足 1kg 的部分。

(2) 汽车在计算托运行李费用时以 kg 为单位计算费用(22 元/kg),对重量中的小数部分进行四舍五入,即将不足 1kg 的部分进行四舍五入。

(3) 飞机在计算托运行李费用时以 g(1kg 等于 1000g)为单位计算费用(0.062 元/g),对重量中的小数部分,即不足 1g 的部分进行四舍五入。

用 double 型变量 weight 存放行李重量,用 charge 存放托运费,程序让用户从键盘输入 weight 的值,该值是行李的重量(以 kg 为单位),然后程序将分别计算出用火车、汽车和飞机托运行李的费用。

## 4. 程序效果示例

程序运行效果如图 3.1 所示。

## 5. 程序模板

按模板要求,将【代码】替换为程序代码。

//BaggageAndMoney.java

import java.util.Scanner;

public class BaggageAndMoney {

public static void main(String args[]) {

int trainCharge = 12;

int carCharge = 22;

double planeCharge = 0.062;

Scanner reader = new Scanner(System.in);

double weight, charge;

System.out.printf("输入行李重量:");

//火车托运费:12 元/kg

//汽车托运费:22 元/kg

//飞机托运费:0.062 元/g

```
输入行李重量:9.62567
行李重量:9.625670 千克(kg)
需要计费的重量:9.0(kg)
用火车托运(12元/kg),费用:108.000000元
需要计费的重量:10.0(kg)
用汽车托运(22元/kg),费用:220.000000元
行李重量:9625.670000克(g)
需要计费的重量:9626(g)
用飞机托运(0.062000元/g),费用:596.812000元
```

图 3.1 托运行李

```
weight = reader.nextDouble();
System.out.printf("行李重量: %f 千克(kg)\n", weight);
System.out.printf("需要计费的重量: %d(kg)\n", (int)weight);
【代码 1】 //将表达式 (int)weight * trainCharge 的值赋值给 charge
System.out.printf("用火车托运( %d 元/kg), 费用: %f 元\n", trainCharge, charge);
System.out.printf("需要计费的重量: %d(kg)\n", (int)(weight + 0.5));
【代码 2】 //将表达式 (int)(weight + 0.5) * carCharge 的值赋值给 charge
System.out.printf("用汽车托运( %d 元/kg), 费用: %f 元\n", carCharge, charge);
System.out.printf("行李重量: %f 克(g)\n", weight * 1000);
System.out.printf("需要计费的重量: %d(g)\n", (int)(weight * 1000 + 0.5));
【代码 3】 //将表达式 (int)(weight * 1000 + 0.5) * planeCharge 的值赋值给 charge
System.out.printf("用飞机托运( %f 元/g), 费用: %f 元\n", planeCharge, charge);
}
}
```

6. 实验指导

为了实现四舍五入,只需将浮点数据加上 0.5,再对结果进行 int 型转换运算即可。类型转换运算符的级别是 2 级,因此,(int)15.9+0.5 的结果是 15.5,即相当于((int)15.9)+0.5,而(int)(15.9+0.5)的结果才是 16。

7. 实验后的练习

在实验的基础上进行改进,让飞机在托运行李时给用户一定的优惠:免收费用中不足一元、一角或一分的金额。

8. 填写实验报告

|           |       |       |       |   |   |
|-----------|-------|-------|-------|---|---|
| 实验编号: 301 | 学生姓名: | 实验时间: | 教师签字: |   |   |
| 实验效果评价    | A     | B     | C     | D | E |
| 模板完成情况    |       |       |       |   |   |
| 实验后练习效果评价 | A     | B     | C     | D | E |
| 练习完成情况    |       |       |       |   |   |
| 总评        |       |       |       |   |   |

实验 2 自动售货机

1. 相关知识点

复合语句的形式是:

```
{
    若干语句
}
```

复合语句是一条语句。if 语句、if...else 语句中的 if 操作和 else 操作都是复合语句。由于复合语句由若干条语句构成,因此,在复合语句中就可以有各种语句,如可以有 if 语句、if...else 语句、switch 语句等。

2. 实验目的

本实验的目的是让学生掌握在 if...else 分支语句的 if 操作中使用 switch 语句。

### 3. 实验要求

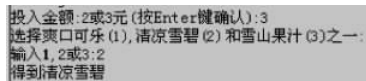
自动售货机为客户提供各种饮料。饮料的价格有两种：2 元和 3 元。用户投入 2 元钱，可以选择“净净矿泉水”“甜甜矿泉水”和“美美矿泉水”三者之一。用户投入 3 元钱，可以选择“爽口可乐”“清凉雪碧”和“雪山果汁”三者之一。编写程序模拟用户向自动售货机投入钱币，得到一种饮料。

### 4. 程序效果示例

程序运行效果如图 3.2 所示。

### 5. 程序模板

请编译、运行模板给出的代码，然后完成实验后的练习。



```
投入金额:2或3元(按Enter键确认):3
选择爽口可乐(1),清凉雪碧(2)和雪山果汁(3)之一:
输入1,2或3:2
得到清凉雪碧
```

图 3.2 自动售货机

```
//MachineSell.java
import java.util.Scanner;
public class MachineSell {
    public static void main(String args[]){
        int money;
        int drinkKind;
        System.out.printf("投入金额:2 或 3 元(按 Enter 键确认):");
        Scanner reader = new Scanner(System.in);
        money = reader.nextInt();
        if(money == 2) {
            System.out.printf("选择净净矿泉水(1), 甜甜矿泉水(2)和美美矿泉水(3)之一:\n");
            System.out.printf("输入 1,2 或 3:");
            drinkKind = reader.nextInt();
            switch(drinkKind) {
                case 1 : System.out.printf("得到净净矿泉水\n");
                    break;
                case 2 : System.out.printf("得到甜甜矿泉水\n");
                    break;
                case 3 : System.out.printf("得到美美矿泉水\n");
                    break;
                default: System.out.printf("选择错误");
            }
        }
        else if(money == 3) {
            System.out.printf("选择爽口可乐(1), 清凉雪碧(2)和雪山果汁(3)之一:\n");
            System.out.printf("输入 1,2 或 3:");
            drinkKind = reader.nextInt();
            switch(drinkKind) {
                case 1 : System.out.printf("得到爽口可乐\n");
                    break;
                case 2 : System.out.printf("得到清凉雪碧\n");
                    break;
                case 3 : System.out.printf("得到雪山果汁\n");
                    break;
                default: System.out.printf("选择错误");
            }
        }
    }
}
```

```

        else {
            System.out.printf("输入的钱币不符合要求");
        }
    }
}

```

## 6. 实验指导

switch 语句中“表达式”的值可以为 byte、short、int、char 型或枚举类型；case 后的“常量值 1”到“常量值 n”也是 byte、short、int、char 型或枚举类型常量，而且互不相同。

## 7. 实验后的练习

改进自动售货机。使得用户也可以投入 5 元钱，选择“草原奶茶”“青青咖啡”和“甜美酸奶”三者之一。

## 8. 填写实验报告

实验编号：302      学生姓名：      实验时间：      教师签字：

| 实验效果评价    | A | B | C | D | E |
|-----------|---|---|---|---|---|
| 模板完成情况    |   |   |   |   |   |
| 实验后练习效果评价 | A | B | C | D | E |
| 练习完成情况    |   |   |   |   |   |
| 总评        |   |   |   |   |   |

# 实验 3 猜数字游戏

## 1. 相关知识点

循环是控制结构语句中的最重要的语句之一，循环语句是根据条件反复执行同一代码块。循环语句有下列两种。

### 1) while 循环

while 语句的一般格式：

```

while(表达式){
    若干语句                      //该复合语句称为循环体
}

```

while 语句的执行规则如下：

- (1) 计算表达式的值，如果该值是 true 时，就执行(2)，否则执行(3)。
- (2) 执行循环体，再执行(1)。
- (3) 结束 while 语句的执行。

### 2) for 循环

for 语句的一般格式：

```

for (表达式 1; 表达式 2; 表达式 3) {
    若干语句                      //该复合语句称为循环体
}

```

for 语句的执行规则如下：

- (1) 计算“表达式 1”，完成必要的初始化工作。
- (2) 判断“表达式 2”的值，若“表达式 2”的值为 true，则执行(3)，否则执行(4)。
- (3) 执行循环体，然后计算“表达式 3”，以便改变循环条件，执行(2)。
- (4) 结束 for 语句的执行。

## 2. 实验目的

本实验的目的是让学生使用 if...else 分支和 while 循环语句解决问题。

## 3. 实验要求

编写一个 Java 应用程序，在主类的 main 方法中实现下列功能：

- 程序随机分配给客户一个 1~100 的整数。
- 用户输入自己的猜测。
- 程序返回提示信息，提示信息分别是“猜大了”“猜小了”或“猜对了”。
- 用户可根据提示信息再次输入猜测，直到提示信息是“猜对了”。

## 4. 程序效果示例

程序运行效果如图 3.3 所示。

## 5. 程序模板

按模板要求，将【代码】部分替换为 Java 程序代码。

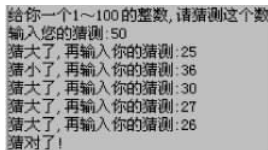


图 3.3 猜数字

```
//GuessNumber.java
import java.util.Scanner;
import java.util.Random;
public class GuessNumber {
    public static void main (String args[]) {
        Scanner reader = new Scanner(System.in);
        Random random = new Random();
        System.out.println("给你一个 1~100 的整数, 请猜测这个数");
        int realNumber = random.nextInt(100) + 1; //random.nextInt(100)是[0,100)中的随机整数
        int yourGuess = 0;
        System.out.print("输入您的猜测:");
        yourGuess = reader.nextInt();
        while(【代码 1】) //循环条件
        {
            if(【代码 2】) //猜大了的条件代码
            {
                System.out.print("猜大了, 再输入你的猜测:");
                yourGuess = reader.nextInt();
            }
            else if(【代码 3】) //猜小了的条件代码
            {
                System.out.print("猜小了, 再输入你的猜测:");
                yourGuess = reader.nextInt();
            }
        }
        System.out.println("猜对了!");
    }
}
```

6. 实验指导

经常使用 while 循环“强迫”程序重复执行一段代码,【代码 1】必须是值为 boolean 型数据的表达式,只要【代码 1】的值为 true,就让用户继续输入猜测。只要用户的输入能使得循环语句结束,就说明用户已经猜对了。

7. 实验后的练习

用 yourGuess > realNumber 替换【代码 1】可以吗? 语句“System. out. println(“猜对了!”);”为何要放在 while 循环语句之后? 放在 while 语句的循环体中合理吗?

8. 填写实验报告

实验编号: 303      学生姓名:                      实验时间:                      教师签字:

| 实验效果评价    | A | B | C | D | E |
|-----------|---|---|---|---|---|
| 模板完成情况    |   |   |   |   |   |
| 实验后练习效果评价 | A | B | C | D | E |
| 练习完成情况    |   |   |   |   |   |
| 总评        |   |   |   |   |   |

实 验 答 案

实验 1

```
【代码 1】 charge = (int)weight * trainCharge;
【代码 2】 charge = (int)(weight+0.5) * carCharge;
【代码 3】 charge = (int)(weight * 1000+0.5) * planeCharge;
```

实验 2

```
【代码 1】 yourGuess!=realNumber;
【代码 2】 yourGuess > realNumber;
【代码 3】 yourGuess < realNumber;
```