

实验内容	第二周实验 Java 编程基础			成绩	
姓 名	王秋锋	学号	2015111948	班 级	计科三班
专 业	计算机科学与技术			日 期	2017 年 9 月 14 日

【实验 01】——Java 基本数据类型

- ◆ 掌握 java 基本数据类型、变量及其使用。
- ◆ 理解运算符的优先级

【实验内容】

1、编写一个 JAVA 程序，对不同类型的类型进行变量的定义并输出相应的值。

【实验结果与分析】

程序及代码如下

```
//注释（解释）作者：
//功能：在控制台显示变量的值
//日期：2017.09.12
//public：表示这个类是公共的，一个 java 文件中只能有一个 public 类
//class：表示这是一个类
//VariableTest：类名（公共类的类名必须和文件名一致）
```

```
public class VariableTest {
    public static void main(String[] args)
    { int a=10;
    System.out.println("a="+a); //输出语句
    }
}
```

(1) 写出程序运行的结果
a=10

(2) 修改上述程序，定义一个字符型变量，并定义其初值为'a'，然后输出该变量的值，在下面写出修改后的代码片段：

```
public class VariableTest {
    public static void main(String[] args) {
        char a='a';
        System.out.println("a="+a); //输出语句
    }
}
输出 a=a
```

(3) 修改上述程序，定义一个字符型变量，并定义其初值为 100，然后输出该变量的值，在下面写出修改后的代码片段：

```
public class VariableTest {
    public static void main(String[] args) {
```

```
char a=100;
System.out.println("a="+a); //输出语句
}
}
输出 a=d
```

(4) 修改上述程序，定义一个字符型变量，并定义其初值为 **65536**，然后输出该变量的值，在下面写出修改后的代码片段，如果有错误，说明原因。

```
public class VariableTest {
    public static void main(String[] args) {
        char a=65536;
        System.out.println("a="+a); //输出语句
    }
}
```

错误原因：Java 中 char 类型用来表示在 Unicode 编码表中的字符，Unicode 编码被设计用来处理各种语言的所有文字，他占两个字节，可以允许有 65536 个字符即 0~65535 的所有字符。然而 65536 超出了这个范围，所以会产生错误

(5) 修改上述程序，定义一个单精度类型的变量，并定义其初值为 **10.0**，然后输出该变量的值，在下面写出修改后的代码片段。

```
public class VariableTest {
    public static void main(String[] args) {
        float a=10.0f;
        System.out.println("a="+a); //输出语句
    }
}
输出 a=10.0
```

(6) 修改上述程序，定义一个双精度类型的变量，并定义其初值为 **10.0**，然后输出该变量的值，在下面写出修改后的代码片段。

```
public class VariableTest {
    public static void main(String[] args) {
        double a=10.0;
        System.out.println("a="+a); //输出语句
    }
}
```

(8) 替换下面的代码段到上述程序，分析现象

```
float f=10.0f;
double d=100.0d;
f=d;
System.out.println("f="+f);
```

分析：float 在计算机内部一般用 32 位二进制表示，而 double 用 64 位表示，相对精度更高。如果要将 double 转化成 float，需要做个强制类型转换。

(9) 替换下面的代码段到上述程序，分析结果

```
int a=10;
char c='a';
a=c+a;
System.out.printf("c=%c",a);
```

运行结果: c=k

(10) 声明一个布尔类型的变量, 其初值为真, 输出该布尔变量的值。

```
Boolean a=true
```

运行结果: a=true

(11) 替换下面的代码段到上述程序, 分析结果

```
char c1='南';
int c2=c1+1;
char c3=(char)c2;
System.out.println("c1="+c1 + "\t c3="+c3);
```

运行结果: c1=南 c3=单

【实验内容】

2. 编写声明不同数据类型变量的程序文件 TypeDemo.java, 源代码如下。

【实验结果与分析】

```
public class TypeDemo {
    public static void main(String[] args){
        byte b = 0x55;
        short s = 0x55ff;
        int i = 1000000;
        long l = 0xfffffL;
        char c = 'a';
        float f = 0.23F;
        double d = 0.7E-3;
        boolean B = true;
        String S = "这是字符串类数据类型";
        System.out.println("字节型变量 b="+b);
        System.out.println("短整型变量 s="+s);
        System.out.println("整型变量 i="+i);
        System.out.println("长整型变量 l="+l);
        System.out.println("字节型变量 c="+c);
        System.out.println("浮点型变量 f="+f);
        System.out.println("双精度变量 d="+d);
        System.out.println("布尔型变量 B="+B);
        System.out.println("字符串类对象 S="+S);
    }
}
```

运行结果:

字节型变量 b=85

短整型变量 s=22015

整型变量 i=1000000

长整型变量 l=65535

字节型变量 c=a

浮点型变量 f=0.23

双精度变量 d=7.0E-4

布尔型变量 B=true

字符串类对象 S=这是字符串类数据类型

分析:

字节型变量 b,用 16 进制表示,赋值是一个字节 0x55,表示成十进制为 85

短整型变量 s,用 16 进制表示,赋值是两个字节 0x55ff,表示成十进制为 22015

整型变量用十进制 1000000 赋值,输出结果也是 1000000

短整型变量 s,用 16 进制表示,赋值两个字节 0xffffL,表示成十进制为 65535

浮点型变量 f,双精度变量 d,字符串类对象 s 输出结果就是赋值结果。

【实验 02】——Java 基础编程练习

【实验内容】

3. 基于控制台的输入:学习掌握利用 Scanner,实现从键盘上接收数据(字符串,整数,实数)

要求:学习并测试 Scanner 类其他方法,如, nextLine, nextFloat, nextDouble 等方法,并给出改写后的源程序代码与程序运行结果。

【源码与运行结果】

nextInt 的源码:

```
import java.util.Scanner;

public class Demo {
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
```

```
System.out.println("请输入一个整数: ");  
int num = sc.nextInt(); //从键盘接收一个整数  
int a, b, c, m;  
a = num / 100;  
b = num % 100 / 10;  
c = num % 100 % 10;  
m = a + b + c;  
System.out.println("a=" + a);  
System.out.println("b=" + b);  
System.out.println("c=" + c);  
System.out.println("m=" + m);  
}
```

```
}
```

请输入一个整数:

374

a=3

b=7

c=4

m=14

nextLine的源码:

```
import java.util.Scanner;
```

```
public class Demo {  
    public static void main(String[] args){  
        Scanner sc = new Scanner(System.in);  
        System.out.println("请输入: ");  
        String num = sc.nextLine(); //从键盘接收一个整数  
        System.out.println("num="+num);  
    }  
}
```

请输入:

abc adf edg

num=abc adf edg

nextFloat 的源码:

```
import java.util.Scanner;
```

```
public class Demo {  
    public static void main(String[] args){  
        Scanner sc = new Scanner(System.in);  
        System.out.println("请输入: ");  
        float num = sc.nextFloat(); //从键盘接收一个整数  
        System.out.println("num="+num);  
    }  
}
```

请输入:

10.11

num=10.11

nextDouble 的源码:

```
import java.util.Scanner;
```

```
public class Demo {  
    public static void main(String[] args){  
        Scanner sc = new Scanner(System.in);  
        System.out.println("请输入: ");  
        double num = sc.nextDouble(); //从键盘接收一个整数  
        System.out.println("num="+num);  
    }  
}
```

请输入:

324.342525

num=324.342525

【实验内容】

4. 随机数的产生: 学习掌握利用 **Random** 类, 产生所需要的随机数。

要求: 学习并测试 **Random** 类, 掌握各种随机数, 以及指定范围内的随机数的产生。利用 **Random** 类, 编写一个程序模拟掷筛子并猜点数的应用程序 (程序每次运行, 产生 1~6 的数字, 然后, 利用 **Scanner** 类从键盘上输入一个用户猜的数字, 给出猜对或猜错的信息)。 **给出掷筛子猜点数程序源码与程序运行结果。**

【源码与运行结果】

源码:

```
import java.util.Arrays; //导入数组类
import java.util.Random; //随机数生成类Random

public class Demo {
    public static void main(String[] args){
        //先建一个对象
        //可以提供一个随机数种子，默认是系统时间
        Random rand = new Random();

        //布尔型随机数
        System.out.println("rand.nextBoolean: "+rand.nextBoolean());

        byte[] buffer = new byte[16];
        //为一个byte数组赋随机数
        rand.nextBytes(buffer);
        System.out.println(Arrays.toString(buffer)); //数组转换为字符串

        //生成0.0~1.0之间的伪随机数
        System.out.println("rand.nextDouble: "+rand.nextDouble());

        //生成0.0~1.0之间的伪随机数
        System.out.println("rand.nextFloat: "+rand.nextFloat());

        //生成一个整数取值范围内的伪随机数
        System.out.println("rand.nextInt: "+rand.nextInt());

        //生成0~26之间的伪随机数
        System.out.println("rand.nextInt(26): "+rand.nextInt(26));

        //生成0.0~1.0之间的伪随机数
        System.out.println("rand.nextLong: "+rand.nextLong());
    }
}
```

运行结果:

rand.nextBoolean: false

[102, 75, -54, 4, 44, 122, 9, 88, 79, 117, 108, 17, 71, -21, -96, -82]

rand.nextDouble: 0.382638848815136

rand.nextFloat: 0.16601992

rand.nextInt: 1098235519

```
rand.nextInt(26): 6
```

```
rand.nextLong: -4439329319585601582
```

模拟掷筛子并猜点数的应用程序:

```
import java.util.Scanner;
import java.util.Random; //随机数生成类Random

public class Demo {
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("请输入你猜的数: ");
        int guess = sc.nextInt();
        Random rand = new Random();
        int ans = rand.nextInt(5)+1;
        System.out.println("筛子点数为: "+ans);
        if(guess == ans)System.out.println("恭喜你, 猜对了!");
        else System.out.println("很遗憾, 猜错了!");
    }
}
```

请输入你猜的数:

3

筛子点数为: 4

很遗憾, 猜错了!

【实验内容】

5 按照项目要求完成.

(1) 题目描述

编写程序, 实现功能 A、B 要求:

A、输出如下程序界面。

***** 银行信息管理系统 *****

中国银行欢迎您

请选择 (请输入 1~4 选择您的操作)

1. 查询余额
2. 存款
3. 取款
4. 退出

其他数值: 输入错误, 请重新输入

B、用户如果输入 1, 输出“查询余额”; 输入 2, 输出“存款”; 输入 3, 输出取款; 输入 4, 程序退出

输入 1~4 以外的数字，输出“输入错误，请重新输入”，并重新输出界面，并让用户程序选择输入。

【源码与运行结果】

源码：

```
import java.util.Scanner;

public class bank {

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        float account = 10000.0f;
        boolean flag = true;

        while (flag) {
            System.out.println("*****银行信息管理系统*****");
            System.out.println("中国银行欢迎您");
            System.out.println("请选择(请输入1~4选择您的操作)");
            System.out.println("1.查询余额");
            System.out.println("2.存款");
            System.out.println("3.取款");
            System.out.println("4.退出");
            System.out.println("其他数值：输入错误，请重新输入");
            System.out.println("*****");

            int num = scan.nextInt();
            int qk;
            int ck;
            switch (num) {
                case 1:
                    System.out.println("您卡上余额为： " + account);
                    break;
                case 2:
                    System.out.println("请选择您要存款的数额：");
                    ck = scan.nextInt();
                    if (ck < 0) {
                        System.out.println("不能取负值的存款");
                    } else if (ck % 100 == 0) {
                        account += ck;
                        System.out.println("存入 " + ck + " 成功");
                    } else {
                        System.out.println("存款失败，请存入100的整数");
                    }
                    break;
                case 3:
```

```

        System.out.println("请选择您要取款的数额: ");
        qk = scan.nextInt();
        if (qk < 0) {
            System.out.println("不能取负值的存款");
        } else if (qk >= account) {
            System.out.println("余额不足");
        } else if (qk % 100 == 0) {
            account -= qk;
            System.out.println("取出" + qk + "成功");
        } else {
            System.out.println("取款失败, 请取出100的整数");
        }
        break;
    case 4:
        System.out.println("退出");
        System.exit(0);
    }
}
}
}
}

```

*****银行信息管理系统*****

中国银行欢迎您

请选择(请输入1~4选择您的操作)

1. 查询余额

2. 存款

3. 取款

4. 退出

其他数值: 输入错误, 请重新输入

1

您卡上余额为: 10000.0

*****银行信息管理系统*****

中国银行欢迎您

请选择(请输入1~4选择您的操作)

1. 查询余额

2. 存款

3. 取款

4. 退出

其他数值: 输入错误, 请重新输入

2

请选择您要存款的数额:

1500

存入1500成功

*****银行信息管理系统*****

中国银行欢迎您

请选择(请输入1~4选择您的操作)

1. 查询余额

2. 存款

3. 取款

4. 退出

其他数值: 输入错误, 请重新输入

3

请选择您要取款的数额:

4500

取出4500成功

*****银行信息管理系统*****

中国银行欢迎您

请选择(请输入1~4选择您的操作)

1. 查询余额

2. 存款

3. 取款

4. 退出

其他数值: 输入错误, 请重新输入

1

您卡上余额为: 7000.0

*****银行信息管理系统*****

中国银行欢迎您

请选择(请输入1~4选择您的操作)

1. 查询余额

2. 存款

3. 取款

4. 退出

其他数值: 输入错误, 请重新输入

4

退出