

面向对象程序设计

实验要求

目录

实验报告模板	4
实验 01 系统安装配置	7
1、实验目的：	7
2、实验内容：	7
实验 02 数据类型分析	8
1、实验目的：	8
2、实验内容：	8
实验 03 综合控制结构程序设计	10
1、实验目的：	10
2、实验内容：	10
实验 04 类的定义与抽象实现	13
1、实验目的	13
2、实验内容	14
实验 05 程序封装实现与测试	15
1、实验目的	15
2、实验内容：	16
实验 06 类的继承实现与测试	17
1、实验目的	17
2、实验内容：	18
实验 07 类的多态实现与测试	错误！未定义书签。
1、实验目的	错误！未定义书签。
2、实验内容：	错误！未定义书签。
实验 08 数组功能与字符串编程	错误！未定义书签。
1、实验目的	错误！未定义书签。
2、实验内容	错误！未定义书签。
实验 09 线程程序设计	错误！未定义书签。
1. 实验目的	错误！未定义书签。
2. 实验内容	错误！未定义书签。
实验 10 IO 基础操作实现	错误！未定义书签。
1. 实验目的	错误！未定义书签。
2. 实验内容	错误！未定义书签。
实验 11 异常处理功能实现	错误！未定义书签。
1. 实验目的	错误！未定义书签。
2. 实验内容	错误！未定义书签。
实验 12 事件监听 GUI 编程实现	错误！未定义书签。
1. 实验目的	错误！未定义书签。
2. 实验内容	错误！未定义书签。

实验报告模板



山西工学院
SHANXI COLLEGE OF TECHNOLOGY

实验报告

学院名称：_____

专业名称：_____

课程名称：_____

班 级：_____ 学号：_____

学生姓名：_____

指导老师：_____

年 月 日

学生姓名		学号		实验成绩	
实验项目名称					
实验地点		实验日期			
<p>一、实验目的和要求</p> <p>二、实验内容和原理</p> <p>三、程序代码（要求有注释说明）</p> <p>四、实验结果截屏图</p>					

五、实验结果与分析

注：实验报告可根据专业实际情况进行调整，封面需使用教务部统一制定的模板

实验 01 系统安装配置

1、实验目的：

熟悉系统的安装与配置方式。了解 Java 语言的基本规则

2、实验内容：

根据自己的计算机配置情况，整理 Java 系统的安装技术文档

- (1)、JDK 安装过程；
- (2)、系统配置方式；
- (3)、其它 IDE 的安装(Notepad++,Eclipse 等内容,选择自己使用的版本)；
- (4)、通过 Hello World 代码验证系统是否安装成功；
- (5)、说明一下 Java 编程的基本规则（文件名，文件结构）
- (6)、要求有插图内容。

实验 02 数据类型分析

1、实验目的：

Java 语言的数据类型比较特殊，分成基础数据类型与引用数据类型，其中基础数据类型四类八种。Java 语言是强类型的编程语言，在编程中出现数据类型不符的时候，系统无法编译程序。本实验的重点是通过练习掌握基础数据类型的使用方式。

2、实验内容：

Java 语言分成基础数据类型与引用数据类型，本实验的重点是分析基础类型中出现的情况与处理方式。参考下面的代码分析错误原因与解决方法。要求说明原因。

通过测试下面的代码，分析数据类型的使用方法

```
public class Test

{

    public static void main(String[]args)

    {

        int i,j;

        float f1=0.1;

        float f2=123;

        double d1=2e20;

        double d2=124;
```



```
byte b1=1,b2=2;

byte b3=129;

j=j+10;

i=i/10;

i=i*0.1;

char c1='a', c2=125;

byte b=b1-b2;

char c=c1+c2-1;

float f3=f1+f2;

float f4=f1+f2*0.1;

double d=d1*i+j;

float f=d1*5+d2;

}

}
```

要求修改并说明原因，增加代码验证修改的结果。

实验 03 综合控制结构程序设计

1、实验目的：

任何编程语言，在具体实现功能时均使用三种典型的控制结构完成。但绝对不是单一的使用，绝大多数情况是综合运用这些规则进行设计。

2、实验内容：

- 1、基础的 IO 练习（使用 Scanner 类）
- 2、选择结构的使用（使用不同的控制方式）
- 3、循环结构的使用（使用不同的控制方式）

实验说明：

- 1、Scanner 类位于 java.util 包中，使用前需要先导入。实例化对象后使用，参考 API 文档中关于 Scanner 类的使用方法。



Scanner 使用分隔符模式将其输入分解为标记，默认情况下该分隔符模式与空白匹配。然后可以使用不同的 next 方法将得到的标记转换为不同类型的值。

例如，以下代码使用户能够从 System.in 中读取一个数：

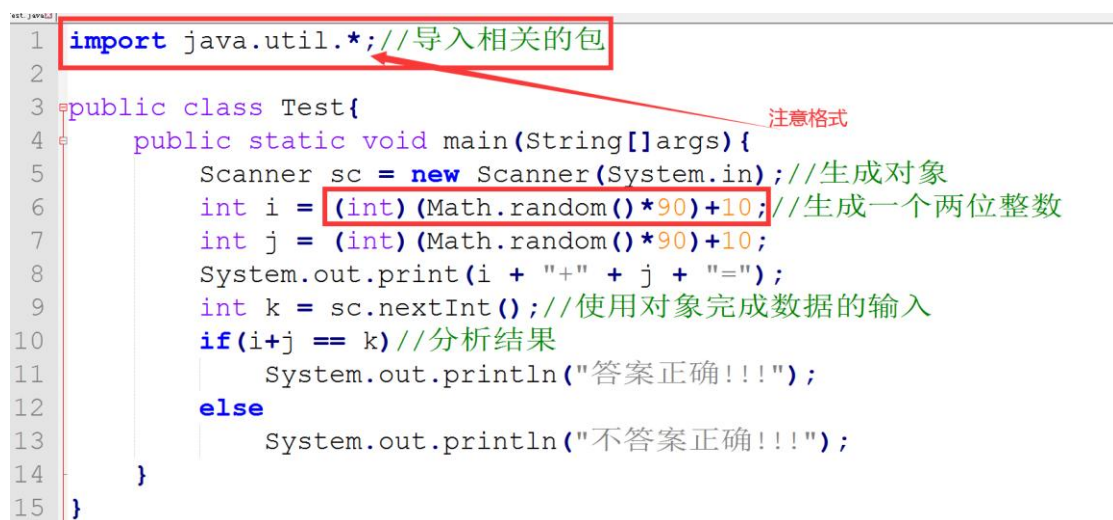
```
Scanner sc = new Scanner(System.in);
int i = sc.nextInt();
```

再看一个例子，以下代码使 long 类型可以通过 myNumbers 文件中的项分配：

```
Scanner sc = new Scanner(new File("myNumbers"));
while (sc.hasNextLong()) {
    long aLong = sc.nextLong();
}
```

注意 Scanner 类有很多方法，使用时针对的对象不同，设计代码的时候注意区分功能。生成数据可以使用 lang 包中的 Math 类的 random() 方法，此方法的返回值是 double 类型，数值范围 0.0~1.0 之间。

2、选择结构的使用



```
1 import java.util.*; //导入相关的包
2
3 public class Test{
4     public static void main(String[] args){
5         Scanner sc = new Scanner(System.in); //生成对象
6         int i = (int)(Math.random()*90)+10; //生成一个两位整数
7         int j = (int)(Math.random()*90)+10;
8         System.out.print(i + "+" + j + "=");
9         int k = sc.nextInt(); //使用对象完成数据的输入
10        if(i+j == k) //分析结果
11            System.out.println("答案正确!!!");
12        else
13            System.out.println("不答案正确!!!");
14    }
15 }
```

```
E:\java2023>java Test
88+91=100
不答案正确!!!
```

```
E:\java2023>java Test
82+24=106
答案正确!!!
```

选择结构有多种方式，考虑如何出现不同的评价方式。使用 switch 或其它结构模式。

3、循环功能代码设计，主要使用的控制结构为 for、while 与

do...while 结构。具体的规则与 C 语言相同。

在设计过程中考虑如何实现综合效果，随机数据的生成方式可以使用 `java.lang.Math` 类，注意它的返回值是 `double` 类型，也可以使用 `java.util.Random` 类中的方法。尝试通过阅读 API 文档学习。

编程实现“整数加法练习”的程序要求如下：

- 1、能够实现连续若干个题（考虑如何实现）；
- 2、计算过程中允许重新录入若干次答案（考虑如何实现）
- 3、能够根据正确与错误统计分数；
- 4、能够实现动态的评价方式；
- 5、要求两个加数与答案均为两位整数（实验难点）；

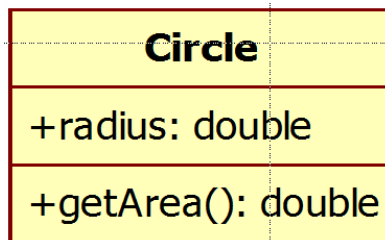
扩展要求：能够实现两位小数的加法运算

实验 04 类的定义与抽象实现

1、实验目的

掌握类的类的方式、对象生成方式，整体结构的处理，如何封装信息，如何验证效果

参考案例：根据图中定义的 Circle 类，实现计算圆面积的方法，通过测试类进行验证。



类的设计代码如下：

```
public class Circle {  
    double radius;//成员变量  
    public Circle(double r){//构造方法  
        radius = r;  
    }  
    public double getArea() {//计算面积  
        final double PI = 3.14;  
        return PI * radius * radius;  
    }  
}
```

测试类代码如下：

```
public class Test {  
    public static void main(String[] args){  
        Circle c1,c2;           //声明c1和c2为Circle对象  
        c1 = new Circle(10);    //创建c1  
        c2 = new Circle(10);    //创建c2  
        c2.radius=40;           //调用c2的修改半径方法，并不影响c1的成员变量状态  
        System.out.println("c1半径="+c1.radius+"\tc2半径="+c2.radius);  
        System.out.println("c1面积="+c1.getArea()+"\tc2面积="+c2.getArea());  
    }  
}
```

运行结果如下：

```
c1半径=10.0      c2半径=40.0  
c1面积=314.0     c2面积=5024.0
```

2、实验内容

参考上面的案例代码效果设计如下的类，注意通过测试类验证类的设计效果。

1、设计一个空间“点”类，计算任意两点之间的距离，计算任意一点距离原点的距离。

2、创建一个复数类 complex，以附属进行数学运算、复数具有如下格式：RealPart+ImaginaryRart*I,其中，I 为-1 的平方根。

要求如下：

- ① 利用浮点变量表示此类的私有数据。提供两个构造方法，一个用于此类声明的对象的初始化；一个为默认的空参构造方法。
- ② 提供两复数加、减、乘的运算方法。
- ③ 按格式(a,b)打印复数。其中 a 为实部，b 为虚部。

实验 05 程序封装实现与测试

1、实验目的

在面向对象程序设计过程中,为了保证数据安全,通常不允许从外面直接访问类的成员变量,系统通过类中设计的特定的方法处理与类相关的数据。因此在设计中对于全体成员变量或需要保护的成员变量通过 private 修饰。需要设计一组 get/set 方法来处理相关业务逻辑。此方式使用 Eclipse 中的生成方式可以快速解决。

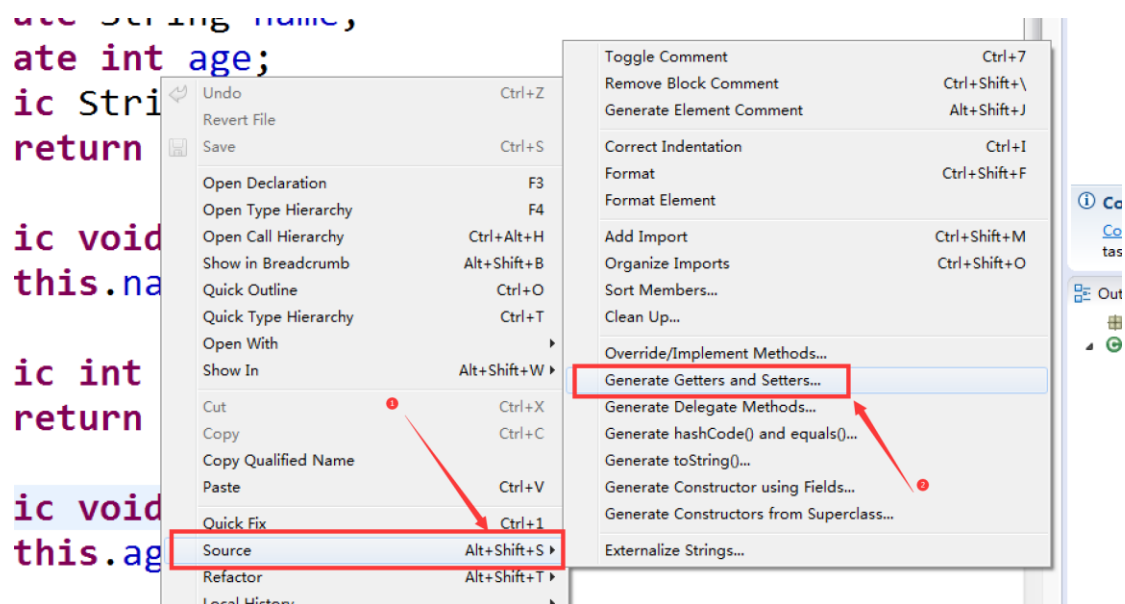
```
public class Circle { //定义Circle类
    private double radius;
    public Circle(double r){           //构造方法
        radius = r;
    }
    public void setRadius(double r){    //设定半径
        radius = r;
    }
    public double getRadius(){          //修改半径
        return radius;
    }
    public double getArea() {           //计算面积
        final double PI = 3.14;
        double area;
        area = PI * radius * radius;
        return area;
    }
}

public class Test {
    public static void main(String[] args){
        Circle c1,c2;                  //声明c1和c2为Circle对象
        c1 = new Circle(10);           //创建c1
        c2 = new Circle(10);           //创建c2
        c2.setRadius(40);              //调用c2的修改半径方法,并不影响c1的成员变量状态
        System.out.println("c1半径="+c1.getRadius()+"\tc2半径="+c2.getRadius());
        System.out.println("c1面积="+c1.getArea()+"\tc2面积="+c2.getArea());
    }
}
```

2、实验内容：

参考上面的演示代码，创建银行账号 Account 类，成员变量为 ID (帐号) number(金额), 实现存款类(balance)的不同业务 :存(deposit)、取(withdraw)和查询(getbalance)等功能。要求对成员变量进行封装处理。考虑如何设计菜单选项。

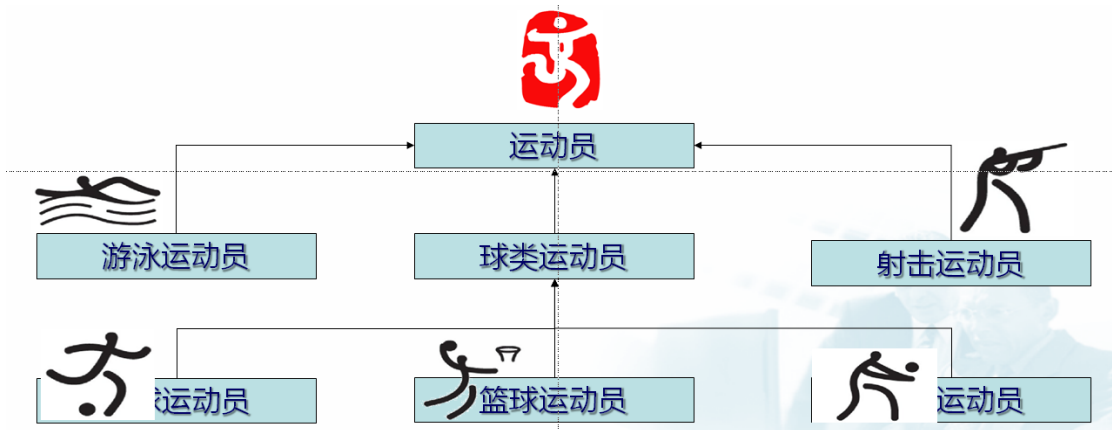
提示：在 Eclipse 中，在代码区按 Alt_Shift_S 快捷键，可以直接使用生成工具完成，也可以在代码点击鼠标右键，选择 Source，然后在级联菜单中选择 Generate Getters/Setters…功能。



实验 06 类的继承实现与测试

1、实验目的

参考课堂教学内容，设计一个能够体现继承概念的故事。

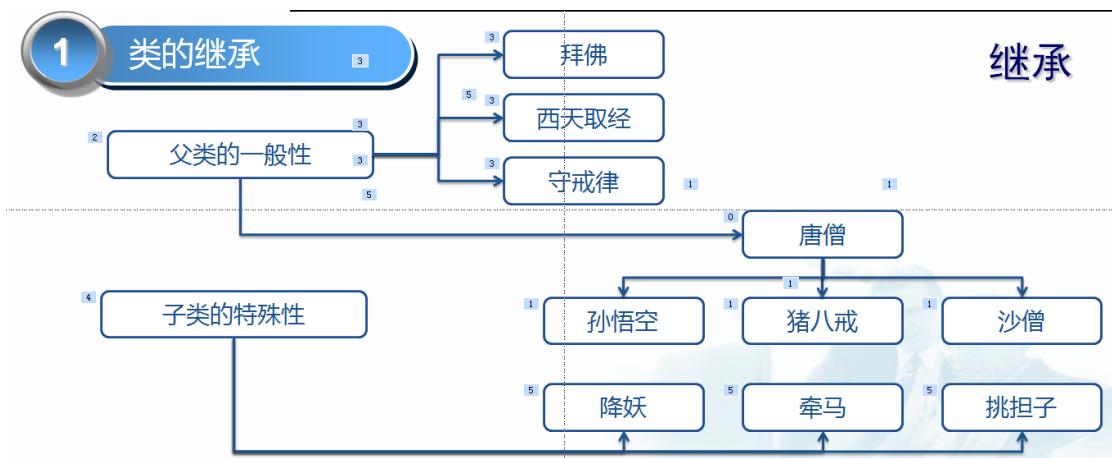


现实世界中存在很多一般与特殊的现象。子类继承父类，父类派生子类，子类还可以派生子类，这样就形成了类的层次结构。

继承机制是实现软件可重用的根基，也是提高软件系统的可扩展性与可维护性的主要途径。

10

子类具有**父类的一般特性**（包括属性和行为），以及自身特殊的特性



2、实验内容：

设计一个动物 (Animal 类) 设计子类 (Dog 与 Cat 实现继续 Animal 模式)。编写一个测试类验证继承方式与结果，特别注意继承父类构造方法的方式，使用 Eclipse 编写此功能，注意代码中的区别是什么？

