

Java 实验指导书

学 院 计算机科学与技术学院

班 级

学 生 banban

学 号

授 课 老 师

二〇二二 年 十 一 月 十 一 日

目录

实验一、安装 JDK 并熟悉 java 的运行环境	3
实验二、基本语法练习	5
实验三、面向对象编程实验	9
实验四、异常处理实验	14
实验五、小应用程序实验	16
实验六、图形图像实验	19
实验七、GUI（图形用户接口）实验	21
实验八、多线程实验	32
实验九、输入输出流实验	36
实验十、数据库应用实验	43

实验一、安装 JDK 并熟悉 java 的运行环境

一、实验目的

熟悉 JAVA 的运行环境及学习简单的编程。

二、预习内容

安装工具软件的基本方法。

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

安装 Eclipse 及 JAVA 的核心编译程序 J2SDK。

1. 打开 Eclipse 的安装盘安装 Eclipse。
2. 在相同目录下安装 J2SDK。
3. 打开 Eclipse 软件对 J2SDK 文件进行配置。
4. 编写一应用程序，在屏幕上显示“HELLO WORLD”和爱心标志。

```
public class Lab01 {  
    public static void main(String args[]) throws InterruptedException {  
        System.out.println("HELLO WORLD");//显示“HELLO WORLD”和爱心标志  
        for (double i = 1.5; i > -1.5; i -= 0.1) {  
            for (double j = -2; j < 2; j += 0.05) {  
                double a = i * i + j * j - 1;  
                if ((a * a * a - i * i * i * j * j) <= 0.0) {  
                    Thread.sleep(10);  
                    System.out.print("*");  
                } else System.out.print(" ");  
            }  
            System.out.println();  
        }  
    }  
}
```

5. 编写一小程序实现上述功能：在屏幕上显示“HELLO WORLD”和爱心标志。



五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度，服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

本次实验建议学时数 2 学时

七、实验总结与体会

实验二、基本语法练习

一、实验目的

1. 熟悉 Java 的基本语法
2. 编写应用程序接收命令行参数
3. 编写应用程序接收用户从键盘的输入
4. 掌握字符串与数组的基本方法

二、预习内容

java 编程的基本结构

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

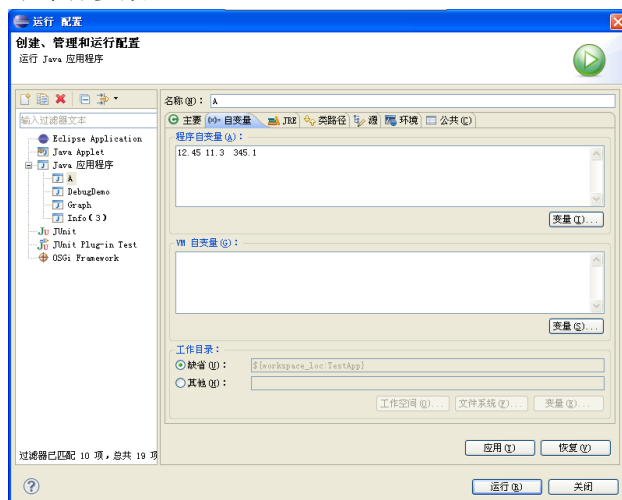
四、实验内容

1. 编写一个应用程序求若干个数的平均数，原始数字要求从命令行输入。应用程序中 main 方法的参数 String 类型的数组 args 能接受用户从命令行键入的参数。

(1) 编辑 Average.java。

```
public class Lab02_1 {  
    public static void main(String[] args) {  
        double n, sum = 0;  
        for (String arg : args) {  
            sum = sum + Double.parseDouble(arg);  
        }  
        n = sum / args.length;  
        System.out.println("average = " + n);  
    }  
}
```

命令行参数: 12.34 34.45 21212121



注意:

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

- 1) 参数的个数可以利用 `args.length` 来取得。
- 2) 参数 “12.34” “34.45” “21212121” 分别赋给了 `main` 方法的参数 `args[0]`、`args[1]` 和 `args[2]`。
- 3) 由于 `args[0]`、`args[1]` 和 `args[2]` 是 `String` 类型的, 所以要利用 `Double.valueOf(String s)` 方法将 `String` 类型转化成 `Double` 类型, 再用 `double Value()` 方法求出 `double` 类型的值。

```

-----
-a----      2022-10-22      10:08      594 Lab01.java
-a----      2022-10-22      14:52     1054 Lab02.class
-a----      2022-10-22      14:47      362 Lab02.java

PS C:\Users\xu\on\Desktop\class_test\src\lab1> java .\Lab02.java 12.34 34.45 21212121
average = 7070722.596666667
PS C:\Users\xu\on\Desktop\class_test\src\lab1>

```

2. 编写一个应用程序, 接受用户从键盘输入的摄氏温度, 然后将其转化成等效的华氏度温度。

方法一: 利用 `System.in.read()` 方法可以接受从键盘输入的 `unicode` 码, 编写一个循环取键值, 循环出口为当输入回车时, 即判断键值是否为 13, 若是, 则退出循环。

(1) 编辑 `Accept.java`, 设保存在 `D:\myjava` 目录下。

```

import java.io.*;
public class Lab02_2 {
    public static void test1(){
        String s="";
        System.out.println("输入摄氏度: (0<数值<100)");
        BufferedReader br = new BufferedReader(new InputStreamReader
(System.in));
        try {
            //读取一行, 返回字符串
            s = br.readLine();
        }catch(Exception e){
            System.out.print("出错了");
        }
        int cel=Integer.parseInt(s.trim());
        double f=cel*1.8+32;
        System.out.println(s+"摄氏度转化成华氏度为: "+f);
    }
    public static void main(String[] args) {
        test1();
    }
}

```

注意:

1) `in` 是 `System` 类的一个属性, `System.in` 的返回类型是 `InputStream` 类型。输入字节

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

流的父类，InputStream 类中的 read（）方法可以读取一个用户从键盘上输入的键值。System.in.read（）的返回值是整型，因此需要先转化成字符型后再连接成字符串。

2) System.in.read（）需要用 try/catch 机制来捕获异常。

3)将字符串转化成响应的整型数值后再利用公式计算出华氏温度。

方法二：由于 InputStream 类中的 read（）方法只能读取一个键值，所以需要利用循环才能读出用户从键盘输入的一串字符，有没有读取一行字符的命令呢？我们可以利用 Java 中类的嵌套来实现。由于 BufferedReader 类有读取一行字符串的方法，所以可以这样去定义对象：

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

(1) 编辑 Accpel.java，设保存在 D:\myjava 目录下。

```
import java.io.*;
public class Lab02_2 {
    public static void test2(){
        int c = 0;
        String s = "";
        System.out.println("输入摄氏度：（0<数值<100）");
        try {
            while (c!=13) {          //当输入的不是回车
                //将int 型转化成char 型，再连接成字符串
                c = System.in.read();
                s = s + (char)c;
            }
        } catch (Exception e){
            System.out.print("出错了");
        }
        int celsius = Integer.parseInt(s.trim()); // trim() 删除字符串头尾空白符
        double fahre = 1.8 * celsius+32;
        System.out.println("转化成华氏度为： " + fahre);
    }
    public static void main(String[] args) {
        Lab02_2.test2();
    }
}
```

3. 编写应用程序，按升序顺序输出一维数组中的元素。

排序的方法有很多，常用的有选择法和冒泡法，在此使用选择法。

(1) 编辑 Sort.java，设保存在 D:\myjava 目录下。

```
public class Lab02_3 {
    public static void main(String[] args) {
        int[] arr = new int[] { 5, 3, 6, 2, 10, 2, 1 };
        selectSort(arr);
        for (int j : arr) {
```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

        System.out.print(j + " ");
    }
}

public static void selectSort(int[] arr) {
    for (int i = 0; i < arr.length - 1; i++) {
        int minIndex = i; // 用来记录最小值的索引位置，默认值为i
        for (int j = i + 1; j < arr.length; j++) {
            if (arr[j] < arr[minIndex]) {
                minIndex = j;
            }
        }
        // 交换当前索引 i 和最小值索引 minIndex 两处的值
        if (i != minIndex) {
            int temp = arr[i];
            arr[i] = arr[minIndex];
            arr[minIndex] = temp;
        }
    }
}
}

```

五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度，服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

1. 建议学时数 2 学时

实验三、面向对象编程实验

一、实验目的

1. 掌握接口的编写及使用
2. 理解继承、多态
3. 掌握包的编写以及如何使用包中的类

二、预习内容

java 的基本语法知识

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

1. 接口的编写

(1) 编辑 Interfaceclass.java, 设保存在 D:\myjava 目录下。

```
interface Interfaceclass {  
    int i=4;  
    int k=5;  
    //填写 2 个方法声明  
    void func1();  
    int func2(int i);  
}
```

(2) 编辑 UseInterface.java, 设保存在 D:\myjava 目录下。

```
class Lab03_1 implements Interfaceclass {  
    int j;  
    public void func1() { //在使用接口的类中一定要实现接口中的所有抽象方法  
        System.out.println("func1="+1);  
    }  
    public int func2(int i) {  
        System.out.println("func2="+1);  
        return i;  
    }  
    public static void main(String [] args) {  
        Lab03_1 x = new Lab03_1();  
        x.func1();  
        x.func2(k);  
    }  
}
```

2. 多态在工资系统中的应用

设计一个根据雇员类型利用 abstract 方法和多态性完成工资单计算的程序。

Employee 是抽象类, Employee 的子类有 Boss (每星期发给他固定工资, 而不计工作时间)、

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

CommissionWorker（除基本工资外还根据销售额发放浮动工资）、PieceWorker（按其生产的产品数发放工资）、HourlyWorker（根据工作时间长短发放工资）。该例的 Employee 的每个子类都声明为 final，因为不需要再继承它们生成子类。

对所有雇员类型都使用 earnings()方法，但每个人挣的工资按他所属的雇员类计算，所有雇员类都是从超类 Earnings()派生出的。所有在超类中声明 earnings()为抽象方法，并且对于每个子类都提供恰当的 earnings()的实现方法。为了计算雇员的工资，程序仅仅使用雇员对象的一个超类引导 并调用 earnings()方法。在一个实际的工资系统中，各种 Employee 对象的引用可以通过一个 Employee 引用数组来实现。程序依次使用数组的每个元素(Employee 引用) 调用每个对象的 employee()方法。

```
public class Lab03_2 {
    public static void main(String[] args) {
        Employee[] em = new Employee[4];
        Boss zt1 = new Boss("aa","bb",7000);
        CommissionWorker zt2 = new CommissionWorker("cc","dd",4000,300);
        PieceWorker zt3 = new PieceWorker("ee","ff",20,400);
        HourlyWorker zt4 = new HourlyWorker("hh","ii",10,500);
        em[0] = zt1;
        em[1] = zt2;
        em[2] = zt3;
        em[3] = zt4;
        for (Employee employee : em) {
            employee.show();
        }
    }
}

abstract class Employee {
    private String firstName;
    private String lastName;
    public Employee() {}
    public Employee(String first,String last) {
        firstName = first;
        lastName = last;
    }
    public String getEmployeeName() { return firstName; }
    public String getLastName() { return lastName; }
    public String toString() { return firstName+lastName; }
    public void show() {}
    public abstract double earnings();
}
```

// Boss（每星期发给他 固定工资，而不计工作时间）

```
final class Boss extends Employee{
    private final double Fixedsalary;
    public Boss(String first,String last,double fixedsalary) {
```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

        super(first,last);
        this.Fixedsalary = fixedsalary;
    }
    public double earnings() { return Fixedsalary; }
    public void show() {
        System.out.println(super.toString()+" "+earnings());
    }
}
// CommissionWorker (除 基本工资 外还根据 销售额 发放 浮动工资)
final class CommissionWorker extends Employee{
    private final double Basesalary;
    private final double Turnover;

    public CommissionWorker(String first,String last,double basesalary,double
turnover) {
        super(first,last);
        this.Basesalary = basesalary;
        this.Turnover = turnover;
    }
    public double earnings() { return Basesalary + Turnover; }
    public void show() {
        System.out.println(super.toString()+" "+earnings());
    }
}
// PieceWorker (按其生产的 产品数 发放 工资)
final class PieceWorker extends Employee{
    private final double Productsnum;
    private final double Price;
    public PieceWorker(String first,String last,double productsnum,double
price) {
        super(first,last);
        this.Productsnum = productsnum;
        this.Price = price;
    }
    public double earnings() { return Productsnum * Price; }
    public void show() {
        System.out.println(super.toString()+" "+earnings());
    }
}

// HourlyWorker (根据 工作时间 长短发放 工资)
final class HourlyWorker extends Employee{
    private final double Worktime;
    private final double Price;

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

    public HourlyWorker(String first,String last,double hour,double price) {
        super(first,last);
        this.Worktime = hour;
        this.Price = price;
    }
    public double earnings() { return Worktime * Price; }
    public void show() {
        System.out.println(super.toString()+" "+earnings());
    }
}

```

3. 包的建立与使用

(1) 编辑 Calculate.java, 设保存在 D:\myjava 目录下。

```

public class Lab03_3 {
    private int a;
    public Lab03_3(int a) {
        this.a = a;
        System.out.println("from constrartion" + this.a);
    }
    public double volume(double height, double width, double depth) {
        return height * width * depth;
    }
    int add(int x, int y) { return x + y; }
    protected void a() { System.out.println("from constration" + a); }
}

```

编译, 查看 D:\myjava 目录下是否生成了 myoackage 文件夹, 在该文件夹中是否有 Calculate.class 文件。

(2) 编辑 Interfaceclass.java, 设保存在 D:\myjava 目录下。

```

import Lab.Lab03_3;
public class Lab03_4 {
    public static void main(String[] args) {
        Lab03_3 c = new Lab03_3(10);
        double s = c.volume(5,6,7);
        System.out.println(s);
        //int b = c.add(5,6);
        //c.a();
    }
}

```

五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度, 服从实验指导教师的安排
3. 按规定的时间完成实验

实验四、异常处理实验

一、实验目的

1. 异常的生产及捕获
2. 自定义异常及其使用

二、预习内容

面向对象的基本知识

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

1. 异常的捕获

计算两数相除并输出结果。使用两个 catch 子句，分别捕捉除数为 0 的异常和参数输入有异常。编译并运行，当输入除数为 0 时，将有异常出现，当输入的不是整数时，如将 30 输成了 3o，出现的是另一种异常。

2. 定义异常

编写程序包含自定义异常，当输入数值为 13 和 4 时抛出该异常。编辑 Ex2.java，设保存在 D:\myjava 目录下。

```
class Ex2 extends Exception {
    Ex2(String msg) {}
}

public class Lab04 {
    private int x;
    void setX(int x) { this.x = x; }

    void f1() throws Ex2 {
        if (x == 13)
            throw new Ex2("I don't like 13!");
        else if (x == 4)
            throw new Ex2("I don't like 4!");
        else
            System.out.println(100 / x);
    }

    public static void main(String[] args) {
        Lab04 a = new Lab04();
        try {
            //          a.setX(5);    // 20
            //          a.setX(13);   // get message:null
```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```
//          a.setX(4);    // get message:null
//          a.setX(0);    // get message:/ by zero
//          a.f1();        // get message:/ by zero
    } catch (Exception e) {
        System.out.println("get message:" + e.getMessage());
    }
}
}
```

编译并运行，分别取消注释上面程序中被注释的语句。当释放 a.setX（13）语句后，查看运行结果，当释放 a.setX（4）语句后，查看运行结果。

五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度，服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

1. 建议学时数 2 学时

实验五、小应用程序实验

一、实验目的

1. 掌握小应用程序的结构及生命周期
2. 掌握带参数小应用程序的编写

二、预习内容

网页制作的相关知识

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

1. 小应用程序生命周期

小应用程序有 `init()`、`start()`、`stop()` 和 `destroy()` 方法，当小应用程序初次加载时，先执行 `init()`，再执行 `start()` 方法。当小应用程序的窗口从后台恢复到前台时，如当小应用程序的窗口被最小化时，会执行 `stop()` 方法，窗口还原时会执行 `start()` 方法，这时不执行 `init()` 和 `destroy()` 方法。当小应用程序被关闭时，再执行 `destroy()` 方法。

编辑 `AppLife.java`，设保存在 `D:\myjava` 目录下。

```
import java.applet.*;
import java.awt.*;

public class Lab05_1 extends Applet {
    static int a, b, c, d;
    int x1 = 150, y1 = 50, x2 = 300, y2 = 50;

    public void paint(Graphics g) {
        g.drawLine(x1, y1, x1, y2);
        g.drawLine(x1, y1, x1, y1 + a);
        g.drawString("init", x1, y1);
        g.drawLine(x1 + 50, y1, x1 + 50, y1 + b);
        g.drawString("start", x1 + 50, y1);
        g.drawLine(x1 + 100, y1, x1 + 100, y1 + c);
        g.drawString("stop", x1 + 100, y1);
        g.drawLine(x1 + 150, y1, x1 + 150, y1 + d);
        g.drawString("destory", x2, y2);
    }

    public void init() {
        a += 50;
        repaint();
    }
}
```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>


```

public void start() {
    b += 50;
    repaint();
}

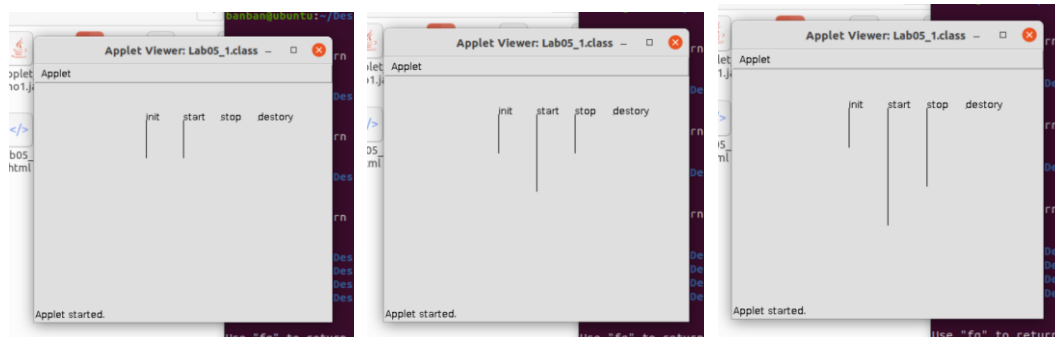
public void stop() {
    c += 50;
    repaint();
}

public void destroy() {
    d += 50;
    repaint();
}
}

```

注意问题：

- (1) 编译 AppLife.java。
- (2) 编辑 AppLife.html，要求存放在与 AppLife.class 相同的目录下。
`<applet code="AppLife.class" height=300 width=400>`
`</applet>`
- (3) 运行 AppLife.html。



2. 带参数的小应用程序

编写一个小应用程序，由 htm 文件向其传递参数。

编写 App.java，设保存在 D:\myjava 目录下。

```

import java.applet.*;
import java.awt.*;
public class Lab05_2 extends Applet{
    String name;
    public void init() { name = getParameter("name"); }
    public void paint(Graphics g) { g.drawString("Welcome" + name, 120, 50); }
}

```

注意不要雷同

banban

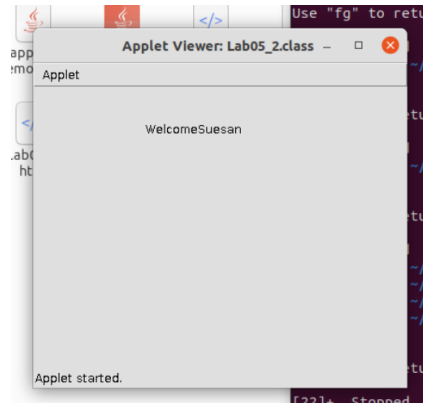
<https://github.com/dream4789/Computer-learning-resources.git>

执行步骤：

- (1) 编译 App.java。
- (2) 编辑 App.html，要求与 App.class 在同一目录下。

```
<applet code="App.class" height=300 width=400>  
<param name="name" value="Suesan">  
</applet>
```

执行 App.html。



五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度，服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

1. 建议学时数 2 学时

实验六、图形图像实验

一、实验目的

1. 了解 Java.awt 包中提供的 Font 类、Color 类、Graphics 类和 Image 类
2. 掌握设置控制 GUI 组件字体和颜色的一般方法
3. 掌握使用 Graphics 绘制线条、矩形、圆角矩形、椭圆、弧等基本图形
4. 掌握使用 Graphics 绘制图象

二、预习内容

绘制图形的基本方法

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

1. 绘制图形

编辑 Gra.java, 保存在 D:\myjava 目录下。

```
import java.applet.*;
import java.awt.*;

public class Lab06 extends Applet {
    Font f1 = new Font("Helvetica", Font.PLAIN, 28);

    public void paint(Graphics g) {
        g.setFont(f1);
        g.setColor(Color.blue);
        g.drawString("我是美丽的大气球!", 120, 50); //在(120,50)处输出字符串“我是美丽的大气球!”
        g.setColor(Color.green);
        g.drawOval(100, 20, 200, 100);
        g.setColor(Color.black);
        g.fillOval(180, 60, 10, 10);
        g.fillOval(180, 60, 10, 10);
        g.setColor(Color.red);
        g.drawArc(130, 50, 50, 180, 180, 123);
        g.setColor(Color.yellow);
        g.drawLine(150, 120, 150, 250); //画直线, 两点为(150,120) (150,250);
    }
}
```

- (1) 编译 Gra.java。
- (2) 编辑 Gra.java, 要求与 Gra.class 在同一目录下。
<applet code="Gra.class" height=300 width=400>

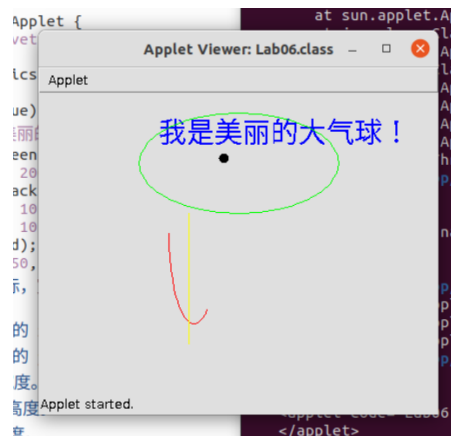
注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

</applet>

(3) 运行 Gra.htm。



五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度，服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

1. 建议学时数 2 学时

实验七、GUI（图形用户接口）实验

一、实验目的

1. 掌握用 `MouseListener` 和 `MouseMotionListener` 接口处处理鼠标事件 `Mouse Event` 的方法。
2. 掌握制作菜单及处理菜单事件的方法
3. 掌握创建对话框及定位、显示、激活和关闭对话框的方法

二、预习内容

图形用户接口编程所需的基本类及使用方法

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

1. 制作一个简单的画板

编辑 `Mou.java`，设保存在 `D:\myjava` 目录下。

```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;

public class Lab07_1 extends Applet implements MouseMotionListener {
    int x = -1, y = -1;

    public void init() {
        setBackground(Color.cyan);
        addMouseMotionListener(this);
    }

    public void paint(Graphics g) {
        if (x != -1 && y != -1) {
            g.setColor(Color.red);
            g.drawLine(x, y, x, y);
        }
    }

    public void mouseDragged(MouseEvent e) {
        x = (int) e.getX();
        y = (int) e.getY();
    }

    public void mouseMoved(MouseEvent e) {}
}
```

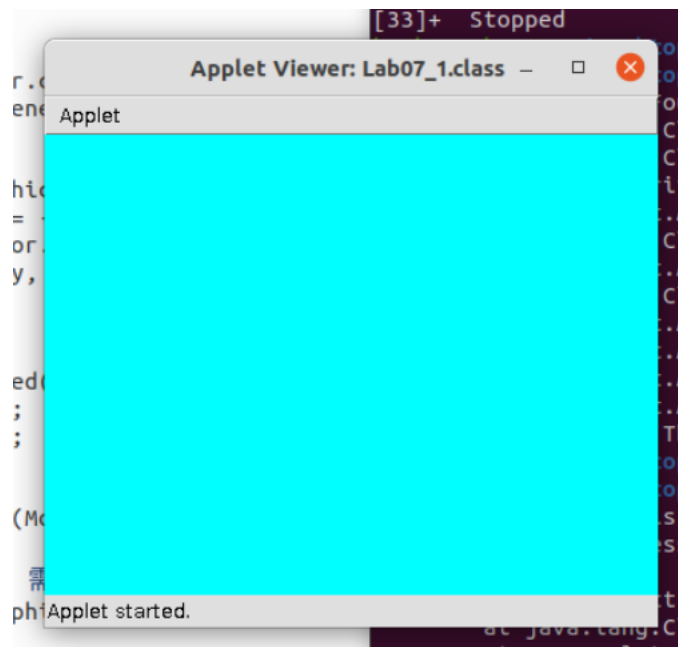
注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```
// 由于使用的是Listener，需要将其他不重载的方法，列举在这里
public void update(Graphics g) { paint(g); }
}
```

编译并运行查看结果。



2. 菜单的编写

编辑 TestMenu.java，设保存在 D:\myjava 目录下。

编译并运行 TestMenu.class 查看结果。

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Lab07_3 {
    public static void main(String[] args) {
        MyDialogFrame df = new MyDialogFrame();
    }
}

class MyDialogFrame extends Frame implements ActionListener, ComponentListener
{
    Dialog MegDlg, InOutDlg;
    Button btn1, btn2, btnY, btnN, btnR;
    JTextField tf = new JTextField("no information", 45);
    JTextField gerMeg = new JTextField("inout information", 10);

    MyDialogFrame() {
```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

super("use dialog");
btn1 = new Button("Hidden");
btn2 = new Button("Asking");
btnY = new Button("YES");
btnN = new Button("NO");
btnR = new Button("Back");

// 页面布局
setLayout(new FlowLayout());

// 添加组件
add(tf);
add(btn1);
add(btn2);

btn1.addComponentListener(this);
this.addWindowStateListener(new WinAdpt()); //Frame 响应窗口关闭事件
btn1.addActionListener(this);
btn2.addActionListener(this);
btnY.addActionListener(this);
btnN.addActionListener(this);
btnR.addActionListener(this);
setSize(1000, 700);
show();
}

public void actionPerformed(ActionEvent e) { //实现 Action Listener 中的方法
    if (e.getActionCommand().equals("隐藏")) {
        MegDlg = new Dialog(this, " true ?", true);

        Panel p1 = new Panel();
        p1.add(new Label(" continue?"));
        MegDlg.add("Center", p1);

        Panel p2 = new Panel();
        p2.add(btnY);
        p2.add(btnN);
        MegDlg.add("Douth", p2);

        MegDlg.setSize(200, 100);
        MegDlg.show();
    } else if (e.getActionCommand().equals("响应")) {
        InOutDlg = new Dialog(this, "input the information");

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

        InOutDlg.add("Center", gerMeg);
        InOutDlg.add("South", btnR);

        InOutDlg.setSize(200, 100);
        InOutDlg.addFocusListener((FocusListener) this);
        InOutDlg.show();
    } else if (e.getActionCommand().equals("是")) {
        MegDlg.dispose();
        btn1.setVisible(false);
    } else if (e.getActionCommand().equals("否")){
        MegDlg.dispose();
    } else if (e.getActionCommand().equals("返回")) {
        tf.setText(gerMeg.getText() + "是对话框的输入");
        InOutDlg.dispose();
    }
}

@Override
public void componentResized(ComponentEvent e) {
    tf.setText(tf.getText()
        + e.getComponent().getClass().getSimpleName() + " resized. ");
}

@Override
public void componentMoved(ComponentEvent e) {
    tf.setText(tf.getText()
        + e.getComponent().getClass().getSimpleName() + " moved. ");
}

@Override
public void componentShown(ComponentEvent e) {
    tf.setText(tf.getText()
        + e.getComponent().getClass().getSimpleName() + " shown. ");
}

@Override
public void componentHidden(ComponentEvent e) {
    tf.setText("按钮"+((Button) e.getComponent()).getLabel()+ " 获得了焦点");
}

// 处理 FocusListener 中的方法
public void focusGained(FocusEvent e){
    // 得到焦点时执行得内容
    gerMeg.setText("对话框"+((Dialog) e.getComponent()).getTitle()+"获得

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

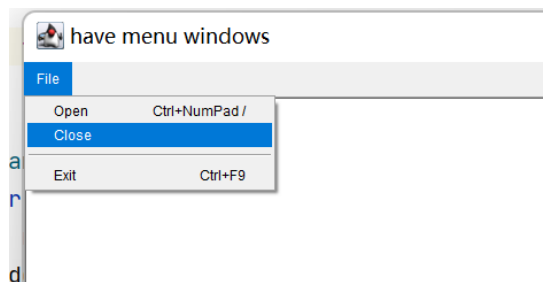

```

了注意的焦点");
    }

    public void focusLost(FocusEvent e){
        // 失去焦点时执行得内容
        gerMeg.setText("对话框"+((Dialog) e.getComponent()).getTitle()+"失去了注意的焦点");
    }
}

class WinAdpt extends WindowAdapter {
    public void windowClosing(WindowEvent e) { //处理关闭窗口事件
        ((Frame) e.getWindow()).dispose();
        System.exit(0);
    }
}

```



3.使用 Dialog 实验消息对话框和一般对话框
编辑 TestDialog.java, 设保存在 D:\myjava 目录下。

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class Lab07_3 {
    public static void main(String[] args) {
        MyDialogFrame df = new MyDialogFrame();
    }
}

class MyDialogFrame extends Frame implements ActionListener, ComponentListener
{
    Dialog MegDlg, InOutDlg;
    Button btn_H, btn_A, btn_Y, btn_N, btn_R;
    JTextField tf = new JTextField("no information", 45);
    JTextField gerMeg = new JTextField("inout information", 10);

    MyDialogFrame() {
        super("use dialog");
    }
}

```

注意不要雷同 banban
<https://github.com/dream4789/Computer-learning-resources.git>

```

    btn_H = new Button("Hidden");
    btn_A = new Button("Asking");
    btn_Y = new Button("YES");
    btn_N = new Button("NO");
    btn_R = new Button("Back");

    // 页面布局
    setLayout(new FlowLayout());

    // 添加组件
    add(tf);
    add(btn_H);
    add(btn_A);

    btn_H.addComponentListener(this);
    this.addWindowStateListener(new WinAdpt()); // Frame 响应窗口关闭事件
    btn_H.addActionListener(this);
    btn_A.addActionListener(this);
    btn_Y.addActionListener(this);
    btn_N.addActionListener(this);
    btn_R.addActionListener(this);
    // 窗口大小
    setSize(1000, 700);
    show();
}

```

```

public void actionPerformed(ActionEvent e) { //Action Listener 中的方法
    if (e.getActionCommand().equals("Hidden")) {
        MegDlg = new Dialog(this, " true ?", true);

        Panel p1 = new Panel();
        p1.add(new Label(" continue?"));
        MegDlg.add("Center", p1);

        Panel p2 = new Panel();
        p2.add(btn_Y);
        p2.add(btn_N);
        MegDlg.add("Douth", p2);

        MegDlg.setSize(200, 100);
        MegDlg.show();
    } else if (e.getActionCommand().equals("Asking")) {
        InOutDlg = new Dialog(this, "input the imformation");
        InOutDlg.add("Center", gerMeg);
    }
}

```

```

        InOutDlg.add("South", btn_R);

        InOutDlg.setSize(200, 100);
        InOutDlg.addFocusListener((FocusListener) this);
        InOutDlg.show();
    } else if (e.getActionCommand().equals("YES")) {
        MegDlg.dispose();
        btn_H.setVisible(false);
    } else if (e.getActionCommand().equals("NO")) {
        MegDlg.dispose();
    } else if (e.getActionCommand().equals("Back")) {
        tf.setText(gerMeg.getText() + "是对话框的输入");
        InOutDlg.dispose();
    }
}

// 列举 Component Listener 中未重载的方法
@Override
public void componentResized(ComponentEvent e) {
    // 当组件的位置发生变化时调用
    tf.setText(tf.getText()
        + e.getComponent().getClass().getSimpleName() + " resized. ");
}

@Override
public void componentMoved(ComponentEvent e) {
    // 当组件的位置发生变化时调用
    tf.setText(tf.getText()
        + e.getComponent().getClass().getSimpleName() + " moved. ");
}

@Override
public void componentShown(ComponentEvent e) {
    // 在组件可见时调用
    tf.setText(tf.getText()
        + e.getComponent().getClass().getSimpleName() + " shown. ");
}

@Override
public void componentHidden(ComponentEvent e) {
    // 组件变得不可见时调用。指定者: 接口 componentlistener 中的
    componenthidden
    tf.setText("按钮" + ((Button) e.getComponent()).getLabel() + " 获得了
    焦点 ");
}

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

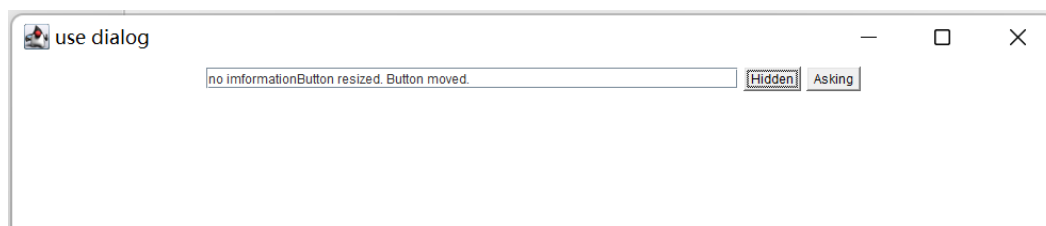
```

    }

    // 处理 FocusListener 中的方法
    public void focusGained(FocusEvent e) {
        // 得到焦点时执行得内容
        gerMeg.setText("对话框" + ((Dialog) e.getComponent()).getTitle() + "
获得了注意的焦点");
    }
    public void focusLost(FocusEvent e) {
        // 失去焦点时执行得内容
        gerMeg.setText("对话框" + ((Dialog) e.getComponent()).getTitle() + "
失去了注意的焦点");
    }
    }
    class WinAdpt extends WindowAdapter {
        public void windowClosing(WindowEvent e) { // 处理关闭窗口事件
            dispose();
            System.exit(0);
        }
    }
}

```

编译并运行查看结果



4、根据上面参考程序， 实现一个登陆系统（在数据库中创建一个用户表（编号，用户名，密码））

1) 创建登陆窗口，实现用户名和密码的数据库连接、判断；如果成功，转动第二步；不成功，重新输入用户名和密码。只能输入三次，如果超过三次，则退出系统。

2) 如果成功，则进入一个新窗口，原来的登陆窗口关闭

```

import java.awt.event.*;
import java.util.Arrays;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JTextField;

```

```

public class Lab07_4 extends JFrame {

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

static int n = 0;
private static JFrame jf = new JFrame("登陆窗口");
private JPanel jp = new JPanel();
private JLabel l_name = new JLabel("账号");
private JLabel l_password = new JLabel("密码");
static JTextField t_name = new JTextField(20);
static JPasswordField t_password = new JPasswordField(20);
private JButton ok = new JButton("登陆");
private JButton cancel = new JButton("退出");
public Lab07_4() {
    ok.addActionListener(new okListener());
    cancel.addActionListener(new cancelListener());
    jp.add(l_name);
    jp.add(t_name);
    jp.add(l_password);
    jp.add(t_password);
    jp.add(ok);
    jp.add(cancel);
    jf.add(jp);
    jf.setVisible(true);
    jf.setSize(300, 150);
    jf.setDefaultCloseOperation(EXIT_ON_CLOSE);
    jf.setLocation(500, 270);
}
class okListener implements ActionListener {
    public void actionPerformed(ActionEvent arg0) {
        check();
    }
}
class cancelListener implements ActionListener {
    public void actionPerformed(ActionEvent arg0) {
        System.exit(0);
    }
}
public static void check() {
    char[] ch = t_password.getPassword(); // 字节数组
    // String password = String.valueOf(ch);
    String password = new String(ch);

    if (t_name.getText().equals("")) {
        JOptionPane.showMessageDialog(null, "请输入用户名!", "error",
JOptionPane.ERROR_MESSAGE);
    } else if (password.equals("")) {
        JOptionPane.showMessageDialog(null, "请输入密码", "error",

```

注意不要雷同

banban

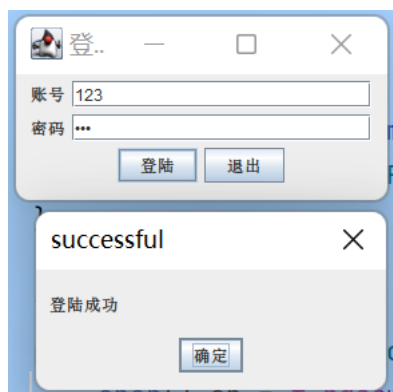
<https://github.com/dream4789/Computer-learning-resources.git>

```

JOptionPane.ERROR_MESSAGE);
    } else if (t_name.getText().equals("123") && password.equals("456")) {
        JOptionPane.showMessageDialog(null, " 登 陆 成 功 ", "successful",
JOptionPane.PLAIN_MESSAGE);
        n = 0;
        jf.setVisible(false);
        JFrame jf2 = new JFrame("登陆成功窗口");
        jf2.setSize(300, 150);
        jf2.setVisible(true);
    } else {
        JOptionPane.showMessageDialog(null, "账号或密码不正确!", "error",
JOptionPane.ERROR_MESSAGE);
        n += 1;
        if (n >= 3)
            System.exit(1);
    }
}

public static void main(String [] args) {
    new Lab07_4();
}
}

```



五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度，服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

1. 建议学时数 6 学时

实验八、多线程实验

一、实验目的

1. 掌握多线程的实现方法
2. 学会利用多线程来显示动画

二、预习内容

线程与进程的基础知识

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

1. 使用 Runnable 接口的方法实现多线程
编辑 TestRunnable.java, 保存在 D:\myjava 目录下。

```
import java.applet.*;
import java.awt.*;
public class Lab08_1 extends Applet implements Runnable {
    Label prompt1 = new Label("the first thread");
    Label prompt2 = new Label("the second thread");
    TextField threadFirst = new TextField(14);
    TextField threadSecond = new TextField(14);
    Thread thread1, thread2;
    int count1 = 0, count2 = 0;
    public void init() {
        add(prompt1);
        add(threadFirst);
        add(prompt2);
        add(threadSecond);
    }
    public void start() {
        thread1 = new Thread(this, "FirstThread");
        thread2 = new Thread(this, "SecondThread");
        thread1.start();
        thread2.start();
    }
    public void run() {
        String currentRunning;
        while (true) {
            try {
                Thread.sleep((int) (Math.random() * 10000));
            } catch (Exception e) {
```

注意不要雷同

banban

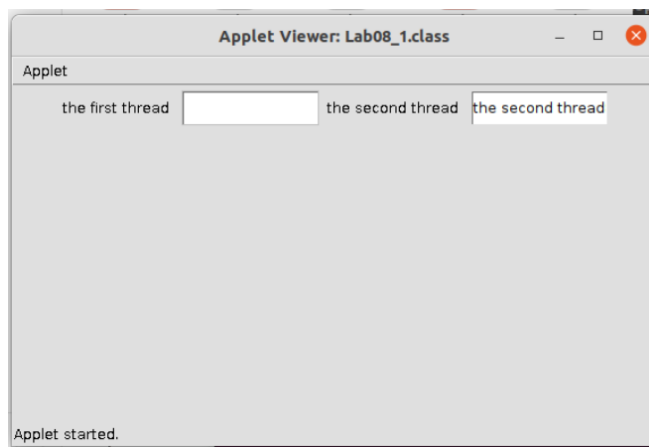
<https://github.com/dream4789/Computer-learning-resources.git>


```

    }
    currentRunning = Thread.currentThread().getName();
    if (currentRunning.equals("FirstThread")) {
        count1++;
        threadFirst.setText("the first thread" + count1 + "use");
    } else if (currentRunning.equals("SecondThread")) {
        count2++;
        threadSecond.setText("the second thread" + count2 + "use");
    }
}
}
}

```

- (1) 编译 TestRunnable.java。
- (2) 编辑 TestRunnable.htm, 要求与 TestRunnable.class 在同一目录下。
`<applet code=TestRunnable.class height=300 width=400>`
`</applet>`
- (3) 运行 TestRunnable.htm。



2. 实现简单动画

实现一个简单动画，效果为一个球由小到大，从屏幕左侧滚动到右侧。

```
import java.awt.*;
```

```

public class Lab08_2 extends Frame {
    public void launchFrame() {
        int width = 800;
        int height = 200;
        setSize(width, height);
        setVisible(true);
        setBackground(Color.blue);
        setTitle("会变大的球");
        new PaintThread().start();
    }
}

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

int x = 0;
int y = 150;
int w = 6;
int h = 6;

public int getX() {
    return x;
}

public void paint(Graphics g) {
    g.setColor(Color.yellow);
    g.fillOval(x, y, w, h);
    if (x < 600) {
        x = x + 10;
        y = y - 2;
        w += 2;
        h += 2;
    }
}

public static void main(String [] args) {
    Lab08_2 mv = new Lab08_2();
    mv.launchFrame();
}

class PaintThread extends Thread {
    public void run() {
        while (true) {
            if (getX() < 600)
                repaint();
            try {
                Thread.sleep(100);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }
}
}

```

3. 新建一个主 Java 文件，命名为 SallTicket java，使用 Runnable 接口定义多线程类 Ticket.java，在 Ticket 中定义成员变量 ticket=100，用同步方式创建两个线程售票。

```

public class Lab08_3 implements Runnable {
    int ticket = 100;

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

public void run() {
    while (true) {
        try {
            Thread.sleep(300);
        } catch (Exception e) {
            System.out.println(e);
        }
        showLeftedTicket();
    }
}

public synchronized void showLeftedTicket() {
    if (Thread.currentThread().getName().equals("1") && (ticket > 0)) {
        System.out.println("1 号窗口售出一张票, 目前还剩余票数: " + --ticket);
    }
    if (Thread.currentThread().getName().equals("2") && (ticket > 0)) {
        System.out.println("2 号窗口售出一张票, 目前还剩余票数: " + --ticket);
    }
    if (ticket <= 0) {
        System.out.println("票已售罄!");
        System.exit(1);
    }
}

public static void main(String[] args) {
    Lab08_3 tt = new Lab08_3();
    Thread t1, t2;
    t1 = new Thread(tt);
    t2 = new Thread(tt);
    t1.setName("1");
    t2.setName("2");
    t1.start();
    t2.start();
}
}

```

五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度, 服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

1. 建议学时数 2 学时

实验九、输入输出流实验

一、实验目的

1. 了解文件的概念和文件对象的创建方法
2. 了解 `FileInputStream` 和 `FileOutoutStream` 的基本概念
3. 学会创建文件输入输出流
4. 掌握使用文件输入输出流读写文件的方法

二、预习内容

输入输出类的使用方法

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

1. 编写程序读取文本文件内容

```
import java.io.*;
import javax.imageio.IOException;

public class Lab09_1 {
    public static void main(String[] args) {
        try {
            File infile = new File("C:/Users/xulon/Desktop/read.txt");
            FileReader reader = new FileReader(infile);
            @SuppressWarnings("resource")
            BufferedReader br = new BufferedReader(reader);
            String s = null;
            while ((s = br.readLine()) != null) {
                System.out.println(s + "");
            }
        } catch (IOException e) {
            System.out.println(e);
        }
    }
}
```

2. 读取图像文件

编辑 `ReadPic.java`。设保存在 `D:myjava` 目录下。设该目录下已经存在一个 `run.gif` 文件（可在此目录下放置一个本机已有的图像文件名字为 `run.gif`。）

```
import java.io.*;

public class Lab09_2 {
```

/**此程序将当前目录下的run.gif 文件复制到copyrun.gif 中

```
Lab09_2() {
    try {
        File f1 = new File("C:/Users/xulon/Desktop/1111.gif");
        File f2 = new File("C:/Users/xulon/Desktop/copy1111.gif");

        FileInputStream inFile = new FileInputStream(f1);
        BufferedInputStream inB = new BufferedInputStream(inFile);
        FileOutputStream outFile = new FileOutputStream(f2);
        BufferedOutputStream outB = new BufferedOutputStream(outFile);

        byte [] b = new byte[(int)f1.length()];
        while (inB.read(b) != -1) {
            outB.write(b);
        }
        outB.flush();
        inB.close();
        outB.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public static void main(String [] args) {
    new Lab09_2();
}
}
```

- (1) 编译并运行 ReadPic.class。
- (2) 打开 copyrun.gif 与 run.gif 进行对比。

3. 将内容写入文件

编辑 UseStream1.java, 设保存在 D:\myjava 目录下。

```
import java.io.*;
public class Lab09_3 {
    Lab09_3(String path) {
        try {
            File f = new File(path, "hello.txt");
            FileWriter outFile = new FileWriter(f);
            BufferedWriter outB = new BufferedWriter(outFile);
            String s = ("你们好, 这是一个测试写入数据的文件。");
            outB.write(s);
            outB.newLine();           //插入一行
            outB.write("这是利用 FileWrite 与 BuffereWrite 的例题。");
            outB.flush();             //需要调用 flush() 方法
        }
    }
}
```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

        outB.close();           // 写入完毕要关闭流
        FileReader inFile = new FileReader(f);    // 从 text.txt 中读取数据
        BufferedReader inB = new BufferedReader(inFile);
        // inB 中含有能够直接读取一行数据的方法 readLine() 供我们使用，当然返回
        // 值 null 时，意味着读取结束
        String fileContent = "", str = "";
        while ((fileContent = inB.readLine()) != null) {
            str = str + fileContent + "\n";
        }
        System.out.println(str);
        inB.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public static void main(String[] args) {
    new Lab09_3("C:/Users/xulon/Desktop/");
}
}

```

编译并运行 UseStream1.class。

4. 使用 JFrame 或 Frame 窗体等 GUI 编程，实现一个记事本程序，实现文件的打开、编辑、保存；文本的全选、复制、剪切、粘贴等功能。

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.*;

public class Lab09_f1 {
    public static void main(String[] args) {
        new Solution();
    }
}

class FileMenu extends Menu { // FileMenu 的内容
    public FileMenu(ActionListener action) {
        super("File");
        MenuItem mi1, mi2, mi3, mi4;
        mi1 = (new MenuItem("Save as"));
        mi2 = (new MenuItem("Save"));
        mi3 = (new MenuItem("Open"));
    }
}

```

注意不要雷同 banban
<https://github.com/dream4789/Computer-learning-resources.git>

```

        mi4 = (new MenuItem("Exit"));
        mi1.addActionListener(action); // 添加监听, 点击的时候才能捕捉到
        mi2.addActionListener(action);
        mi3.addActionListener(action);
        mi4.addActionListener(action);
        add(mi1);
        add(mi2);
        add(mi3);
        addSeparator();
        add(mi4);
    }
}

class FontMenu extends Menu { // FontMenu 的内容
    public FontMenu(ActionListener action) {
        super("Font");
        MenuItem mi1, mi2, mi3, mi4, mi5;
        mi1 = new MenuItem("TimesRoman");
        mi2 = new MenuItem("Comic Sans MS");
        mi3 = new MenuItem("consolas");
        mi4 = new MenuItem("Increase font size");
        mi5 = new MenuItem("Decrease font size");
        mi1.addActionListener(action); // 添加监听
        mi2.addActionListener(action);
        mi3.addActionListener(action);
        mi4.addActionListener(action);
        mi5.addActionListener(action);

        add(mi1);
        add(mi2);
        add(mi3);
        addSeparator();
        add(mi4);
        add(mi5);
    }
}

class Solution extends Frame implements ActionListener {
    private String strFont = "consolas"; // 记录字体, 以便换字体
    private int Fontsize = 30; // 字体大小
    private JTextArea jt2; // 文本域
    private JFrame jf; // 窗口
    private String path = ""; // 记录路径, 以便保存
    private String filename = ""; // 记录文件名, 以便保存

    public Solution() { // 搭建基础框架

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

    jt2 = new JTextArea();//new 一个文本区
    String title = "记事本";
    jf = new JFrame(title);
    MenuBar mb = new MenuBar();//新建一个菜单栏
    FileMenu fileMenu = new FileMenu(this);//搭建两个Menu
    mb.add(fileMenu);
    FontMenu fontMenu = new FontMenu(this);
    mb.add(fontMenu);
    jf.setMenuBar(mb);//菜单栏放上去
    jf.setVisible(true);//窗体可见
    jf.setSize(600, 400);//窗体大小
    jf.setLayout(new BorderLayout());//边界布局
    jf.add(BorderLayout.CENTER, jt2);//文本框边界中间放置
    jt2.setBackground(Color.white);
    jt2.setFont(new Font(strFont, Font.PLAIN, Fontsize));
}

public void SaveAsAction() { //Save 动作的处理
    FileDialog fd = new FileDialog(jf, "Save as", FileDialog.SAVE);
    fd.setVisible(true);
    path = fd.getDirectory();//获取路径
    filename = fd.getFile();//获取文件名
    try {
        FileOutputStream out = new FileOutputStream(path + filename, false);
        String s = jt2.getText();//输出到文件
        byte[] b = s.getBytes();
        out.write(b);
        out.close();
    } catch (IOException ex) {
        System.out.println(ex);
    }
}

public void SaveAction() { //保存的处理
    if (path.length() != 0) {
        try {
            FileOutputStream out = new FileOutputStream(path + filename,
false);//这里 false 是覆盖, 否则是加到原文件末尾
            String s = jt2.getText();
            byte[] b = s.getBytes();
            out.write(b);
            out.close();
        } catch (IOException ex) {
            System.out.println(ex);
        }
    }
}

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>


```

    }
} else {
    JOptionPane jo = new JOptionPane();
    jo.showMessageDialog(null, "还没有将其保存到文件, 请选择 Save as 将内容保存到文件");
}
}

public void OpenAction() { // 打开的处理
    FileDialog fd = new FileDialog(jf, "Open", FileDialog.LOAD);
    fd.setVisible(true);
    path = fd.getDirectory();
    filename = fd.getFile();
    if (path == null || filename == null) return;
    jt2.setText(""); // 清空当前的内容
    try {
        BufferedReader re = new BufferedReader(new FileReader(path + filename)); // 读取文件
        String s = null;
        while ((s = re.readLine()) != null) { // 写入文本域
            jt2.append(s + "\r\n"); // 对于不同操作系统的回车处理
        }
        re.close();
    } catch (IOException e1) {
        e1.printStackTrace();
    }
}

public void actionPerformed(ActionEvent e) {
    String arg = e.getActionCommand();
    switch (arg) {
        case "Exit": // 退出
            System.exit(0);
        case "Comic Sans MS": // 三个 if 换字体
            strFont = "Comic Sans MS";
            break;
        case "TimesRoman":
            strFont = "TimesRoman";
            break;
        case "consolas":
            strFont = "consolas";
            break;
        case "Increase font size": // 改变字体大小
            Fontsize = Fontsize + 3;
            break;
        case "Decrease font size":

```

注意不要雷同

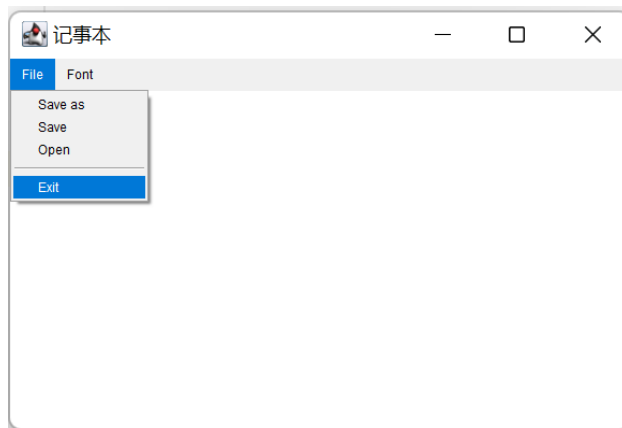
banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

        Fontsize = Fontsize - 3;
        break;
    case "Save as": {
        SaveAsAction();
        break;
    } // 另存为, 把文件内容导出存放
    case "Save": {
        SaveAction();
        break;
    } // 保存文件内容, 和另存为相似
    case "Open": {
        OpenAction();
        break;
    } // 打开文件
    }
    jt2.setFont(new Font(strFont, Font.PLAIN, Fontsize));
    System.out.println(arg);
}
public void paint(Graphics g) {
    g.setFont(new Font(strFont, Font.BOLD, Fontsize));
    g.drawString("", 50, 150);
}
}
}

```



五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度, 服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

1. 建议学时数 4 学时

实验十、数据库应用实验

一、实验目的

1. 掌握 Java 应用程序连接数据库
2. 掌握 Java 小应用程序连接数据库

二、预习内容

数据库表的应用

三、实验设备与环境

装有 JAVA 语言工具软件 (Eclipse、IDEA 等)的微机若干

四、实验内容

1. 完成一个系统：实现 GUI 的学生成绩管理
- 1) 在数据库创建表：学生表（学号、姓名、性别、出生年月）、课程表（课程号、课程名）、成绩表（学号、课程号，成绩）
- 2) 实现用户登陆功能
- 3) 在主窗口实现学生信息管理、课程信息管理、成绩录入修改查询和提交功能、学生成绩的分析（统计各个分数段的人数，并画出统计图）

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;

import java.sql.*;

public class Lab10 {
    // 启动登录界面
    public static void main(String[] args) {
        new Login_win();
    }
}

// Login.java
class Login implements ActionListener {
    // 定义主窗口
```

注意不要雷同
<https://github.com/dream4789/Computer-learning-resources.git>

banban

```

private final JFrame jf;
// 定义输入用户名和密码的标签提示
private final JLabel InputUserName;
private final JLabel InputPassWord;
// 定义输入用户名文本框
private final JTextField UserName;
// 定义输入密码框
private final JPasswordField PassWord;
// 定义登录和取消按钮
private final JButton Login;
private final JButton Cancel;

Login() {
    // 各组件实例化过程
    jf = new JFrame("Login");
    InputUserName = new JLabel("      ID:      ");
    InputPassWord = new JLabel("password:");
    UserName = new JTextField();
    PassWord = new JPasswordField();
    Login = new JButton("登录");
    Cancel = new JButton("退出");

    // 设置主窗口大小、位置和布局
    jf.setSize(400, 450);
    jf.setLocation(600, 400);

    // 设置窗口流式布局
    jf.setLayout(new FlowLayout());

    // 设置用户名和密码框大小
    UserName.setPreferredSize(new Dimension(300, 30));
    PassWord.setPreferredSize(new Dimension(300, 30));

    // 依次向主窗口添加各组件
    jf.getContentPane().add(InputUserName);
    jf.getContentPane().add(UserName);
    jf.getContentPane().add(InputPassWord);
    jf.getContentPane().add(PassWord);
    jf.getContentPane().add(Login);
    jf.getContentPane().add(Cancel);

    // 设置主窗口不可调节大小
    //      jf.setResizable(false);
    jf.setResizable(true);
}

```

```

// 设置主窗口默认关闭操作
jf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

// 给登录和取消按钮添加 Action 监听器
Login.addActionListener(this);
Cancel.addActionListener(this);

// 设置主窗口可见
jf.setVisible(true);
}

@Override
public void actionPerformed(ActionEvent e) {
    // 如果单击【退出】按钮则程序退出
    if (e.getSource().equals(Cancel)) {
        System.exit(0);
    }
    // 如果单击【登录】按钮则检查用户名和密码是否匹配
    else if (e.getSource().equals(Login)) {
        // 如果用户名和密码匹配，则打开具体操作面板
        if (UserName.getText().equals("admin") &&
String.valueOf(Password.getPassword()).equals("1234")) {
            // MySQLGUI myS = new MySQLGUI();
            // myS.initial();
            new MySQLGUI();
            jf.setVisible(false);
            jf.dispose();
        }
        // 如果用户名和密码不匹配，则给出提示对话框
        else {
            JOptionPane.showMessageDialog(jf, "用户名或密码错误", "登陆失
败",
                                JOptionPane.CLOSED_OPTION,
                                JOptionPane.ERROR_MESSAGE, null, null, null);
        }
    }
}
}
}
}

```

// OperationMySQL.java

```

class OperationMySQL {
    // 定义数据库连接 url
    private String dburl = null;

```

```

// 定义数据库连接
private Connection conn = null;
// 定义数据库状态
private PreparedStatement stmt = null;
// 定义数据库返回结果集
private ResultSet rs = null;
// 定义数据库用户名
private String username = null;
// 定义数据库连接密码
private String password = null;
// 定义数据库驱动方式
private String dbdriver = null;

// 设置数据库连接url 的方法
public void setDburl(String dburl) {
    this.dburl = dburl;
}

// 返回当前实例数据库连接url
public String getDburl() {
    return dburl;
}

// 返回当前实例结果集的方法
public ResultSet getRs() {
    return rs;
}

// 设置当前实例结果集的方法
public void setRs(ResultSet rs) {
    this.rs = rs;
}

// 设置数据库用户名的方法
public void setUsername(String username) {
    this.username = username;
}

// 返回当前实例化数据库用户名
public String getUsername() {
    return username;
}

// 设置数据库连接的方法

```

```

public void setPassword(String password) {
    this.password = password;
}

// 返回当前实例数据库连接密码
public String getPassword() {
    return password;
}

// 设置数据库驱动方式的方法
public void setDbdriver(String dbdriver) {
    this.dbdriver = dbdriver;
}

// 返回当前实例数据库驱动方式的方法
public String getDbdriver() {
    return dbdriver;
}

// 创建数据库连接的方法
Connection CreateConnection(String dburl, String username, String
password) throws Exception {
    setDburl(dburl);
    setUsername(username);
    setPassword(password);
    Class.forName(getDbdriver());
    // 根据数据库路径、用户名和密码创建连接并返回该连接
    return DriverManager.getConnection(dburl, username, password);
}

// 关闭结果集的方法
public void CloseRS() {
    try {
        rs.close();
    } catch (SQLException e) {
        System.out.println("关闭结果集时发生错误！");
    }
}

// 关闭状态的方法
public void CloseStmt() {
    try {
        stmt.close();
    } catch (SQLException e) {

```

```

        System.out.println("关闭状态时发生错误！");
    }
}

// 关闭连接的方法
public void CloseConnection() {
    try {
        conn.close();
    } catch (SQLException e) {
        System.out.println("关闭连接时发生错误！");
    }
}

// 增
void executeInsert(String InsertID, String InsertName, String Chinese,
String Math, String English) throws Exception {
    try {
        conn = CreateConnection(getDburl(), getUsername(),
getPassword());
        stmt = conn.prepareStatement("insert into grade
values(?,?,?,?,?)");
        stmt.setString(1, InsertID);
        stmt.setString(2, InsertName);
        stmt.setString(3, Chinese);
        stmt.setString(4, Math);
        stmt.setString(5, English);
        stmt.executeUpdate();
    } catch (SQLException ex) {
        System.err.println(ex.getMessage());
    }
}

// 删
void executeDelete(String DeleteID) throws Exception {
    try {
        conn = CreateConnection(getDburl(), getUsername(),
getPassword());
        stmt = conn.prepareStatement("delete from grade where ID = ?");
        stmt.setString(1, DeleteID);
        stmt.executeUpdate();
        CloseStmt();
        CloseConnection();
    } catch (SQLException ex) {
        System.err.println(ex.getMessage());
    }
}

```



```

    }
}

// 查 主键 是否在表中
ResultSet executeQuery(String StuID) throws Exception {
    try {
        String sql = "select * from grade where ID = ?";
        conn = CreateConnection(getDburl(), getUsername(),
getPassword());
        stmt = conn.prepareStatement(sql);
        stmt.setString(1, StuID);
        rs = stmt.executeQuery();
    } catch (SQLException e) {
        System.err.println(e.getMessage());
    }
    return rs;
}

// 改
void executeUpdate(String UpdateID, String UpdateItem, String
UpdateContent) throws Exception {
    try {
        conn = CreateConnection(getDburl(), getUsername(),
getPassword());
        String sql = "update grade set " + UpdateItem + " = ? where ID
= ?";

        stmt = conn.prepareStatement(sql);
        stmt.setString(1, UpdateContent);
        stmt.setString(2, UpdateID);
        stmt.executeUpdate();
    } catch (SQLException ex) {
        System.err.println(ex.getMessage());
    }
}

// 按条件查询
ResultSet executeQueryByCondition(String stuid, String stuname, String
chinese, String math, String english) throws Exception {
    try {
        String sql = "select * from grade where ID like ? and Name like ?
and Chinese like ? " +
        "and Math like ? and English like ? order by ID asc";
        conn = CreateConnection(getDburl(), getUsername(),
getPassword());

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

        stmt = conn.prepareStatement(sql);
        if (stuid.equals("")) {
            stmt.setString(1, "");
        } else {
            stmt.setString(1, "" + stuid + "");
        }
        if (stuname.equals("")) {
            stmt.setString(2, "");
        } else {
            stmt.setString(2, "" + stuname + "");
        }
        if (chinese.equals("")) {
            stmt.setString(3, "");
        } else {
            stmt.setString(3, "" + chinese + "");
        }
        if (math.equals("")) {
            stmt.setString(4, "");
        } else {
            stmt.setString(4, "" + math + "");
        }
        if (english.equals("")) {
            stmt.setString(5, "");
        } else {
            stmt.setString(5, "" + english + "");
        }
        rs = stmt.executeQuery();
    } catch (SQLException ex) {
        System.err.println(ex.getMessage());
    }
    return rs;
}

```

// 选课表查询

```

ResultSet executeQueryByCourse(String course) throws Exception {
    try {
        conn = CreateConnection(getDburl(), getUsername(),
getPassword());
        String sql = "select * from course where Course = ?";
        stmt = conn.prepareStatement(sql);
        stmt.setString(1, course);
        rs = stmt.executeQuery();
    } catch (SQLException ex) {
        System.err.println(ex.getMessage());
    }
}

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

    }
    return rs;
}

// 课程总计查询
ResultSet executeQueryByGrade(String grade) throws Exception {
    try {
        conn = CreateConnection(getDburl(), getUsername(),
getPassword());
        String sql = "select * from summary where Course = ?";
        stmt = conn.prepareStatement(sql);
        stmt.setString(1, grade);
        rs = stmt.executeQuery();
    } catch (SQLException ex) {
        System.err.println(ex.getMessage());
    }
    return rs;
}
}

```

```

// MySQLGUI.java
class MySQLGUI extends JFrame implements MouseListener, ItemListener {
    // 定义选项卡
    private JTabbedPane Base;
    // 定义选项卡上的嵌板
    /*
    * jp1, 添加记录
    * jp2, 删除记录
    * jp3, 更新记录
    * jp4, 查找记录
    * jp5, 选课记录
    * jp6 课程平均分
    * */
    private JPanel jp1, jp2, jp3, jp4, jp5, jp6;
    // 定义各按钮
    /*
    * InsertRecord, 添加记录按钮
    * InsertReset, 添加取消按钮
    * DeleteRecord, 删除记录按钮
    * DeleteReset, 删除取消按钮
    * QueryRecord, 查询记录按钮
    * UpdateRecord, 更改记录按钮
    * UpdateReset, 重置更新框
    */
}

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

    * CourseQuery, 选课表查询按钮
    * GradeQuery, 成绩查询按钮
    * */
private JButton InsertRecord, InsertReset, DeleteRecord, DeleteReset,
    QueryRecord, UpdateRecord, UpdateReset, CourseQuery, GradeQuery;
// 定义各标签
/*
    * InsertID1,      插入学号提示标签
    * InsertName1,    插入姓名提示标签
    * InsertChinese1, 插入语文成绩提示标签
    * InsertMath1,    插入数学提示标签
    * InsertEnglish1, 插入英语提示标签
    * DeleteID1,      删除学号提示标签
    * UpdateID1,      更新学号提示标签
    * */
private JLabel InsertID1, InsertName1, InsertChinese1, InsertMath1,
    InsertEnglish1, DeleteID1, UpdateID1;
// 定义各文本框
/*
    * InsertID2,      插入学号文本框
    * InsertName2,    插入姓名文本框
    * InsertChinese2, 插入语文文本框
    * InsertMath2,    插入数学文本框
    * InsertEnglish2, 插入英语文本框
    * DeleteID2,      所要删除学号的文本框
    * UpdateID2,      所要更新学号的文本框
    * UpdateContent,  更新内容填写文本框
    * IDCondition,    查询ID 文本框
    * NameCondition,  查询姓名文本框
    * ChineseCondition, 查询语文文本框
    * MathCondition,  查询数学文本框
    * EnglishCondition, 查询英语文本框
    * */
private JTextField InsertID2, InsertName2, InsertChinese2, InsertMath2,
    InsertEnglish2,
    DeleteID2, UpdateID2, UpdateContent, IDCondition, NameCondition,
    ChineseCondition, MathCondition,
    EnglishCondition;
// 定义显示结果文本域 显示 jp4 jp5 jp6 的查询结果
/*
    * QueryRecordResult, 查询学生信息结果文本域
    * CourseQueryResult, 查询课程信息文本域
    * GradeQueryResult,  查询课程成绩平均分文本域
    * */

```

```

private JTextArea QueryRecordResult, CourseQueryResult,
GradeQueryResult;
// 定义查询选项
/*
 * ID,          选择学号查询
 * Name,        选择姓名查询
 * Chinese,     选择语文查询
 * Math,        选择数学查询
 * English,     选择英语查询
 * */
private JRadioButton ID, Name, Chinese, Math, English;
// 定义一个数据库操作的实例
private OperationMySQL db = null;
// 定义滚动条
private JScrollPane scroll = null;
private JScrollPane CourseScroll = null;
private JScrollPane GradeScroll = null;
// 定义一个复选框用于选择更新的项目
private JComboBox<String> UpdateItem = null;
// 定义复选框用于选择查询的项目
private JComboBox<String> CourseItem = null;    // 课程信息复选框
private JComboBox<String> GradeItem = null;    // 课程成绩复选框

MySQLGUI() {
    // 设置各按钮信息
    setButton();
    // 设置各标签信息
    setLabel();
    // 设置各文本框信息
    setTextField();
    // 设置各面板信息
    setPanel();
    // 设置布局信息
    setLayout();
    // 设置选项卡信息
    setBase();
    // 设置主窗口信息
    setThis();
    // 设置数据库信息
    setDB();
}

// 设置各按钮信息的方法
private void setButton() {

```

```

// jp1 上的按钮
InsertRecord = new JButton("添加");
InsertRecord.setFont(new Font("宋体", 1, 20)); // 1 代表加粗,
20 代表字体大小
InsertRecord.setBackground(Color.CYAN);
InsertRecord.setBounds(150, 400, 100, 45);
InsertRecord.setMargin(new Insets(0, 0, 0, 0)); // 设置按钮的边
缘空白为四个方向全为0, 也即让按钮中的文本与按钮边缘贴齐
InsertReset = new JButton("重置");
InsertReset.setFont(new Font("宋体", 1, 20));
InsertReset.setBackground(Color.CYAN);
InsertReset.setBounds(300, 400, 100, 45);
InsertReset.setMargin(new Insets(0, 0, 0, 0));
// jp2 上的按钮
DeleteRecord = new JButton("删除信息");
DeleteRecord.setFont(new Font("宋体", 1, 20));
DeleteRecord.setBackground(Color.CYAN);
DeleteRecord.setBounds(150, 350, 100, 45);
DeleteRecord.setMargin(new Insets(0, 0, 0, 0));
DeleteReset = new JButton("重置");
DeleteReset.setFont(new Font("宋体", 1, 20));
DeleteReset.setBackground(Color.CYAN);
DeleteReset.setBounds(300, 350, 100, 45);
DeleteReset.setMargin(new Insets(0, 0, 0, 0));
// jp3 上的按钮
UpdateRecord = new JButton("更新");
UpdateRecord.setFont(new Font("宋体", 1, 20));
UpdateRecord.setBackground(Color.CYAN);
UpdateRecord.setBounds(250, 400, 100, 45);
UpdateReset = new JButton("重置");
UpdateReset.setFont(new Font("宋体", 1, 20));
UpdateReset.setBackground(Color.CYAN);
UpdateReset.setBounds(400, 400, 100, 45);
// jp4 上的按钮
ID = new JRadioButton("学号");
ID.setFont(new Font("宋体", 1, 15));
ID.setMargin(new Insets(0, 0, 0, 0));
ID.setBounds(30, 300, 55, 20);
Name = new JRadioButton("姓名");
Name.setFont(new Font("宋体", 1, 15));
Name.setMargin(new Insets(0, 0, 0, 0));
Name.setBounds(30, 330, 55, 20);
Chinese = new JRadioButton("语文");
Chinese.setFont(new Font("宋体", 1, 15));

```

```

Chinese.setMargin(new Insets(0, 0, 0, 0));
Chinese.setBounds(30, 360, 55, 20);
Math = new JRadioButton("数学");
Math.setFont(new Font("宋体", 1, 15));
Math.setMargin(new Insets(0, 0, 0, 0));
Math.setBounds(30, 390, 55, 20);
English = new JRadioButton("英语");
English.setFont(new Font("宋体", 1, 15));
English.setMargin(new Insets(0, 0, 0, 0));
English.setBounds(30, 420, 55, 20);
QueryRecord = new JButton("查询");
QueryRecord.setFont(new Font("宋体", 1, 20));
QueryRecord.setBackground(Color.CYAN);
QueryRecord.setBounds(600, 400, 80, 45);
// jp5 上的按钮
CourseQuery = new JButton("查询");
CourseQuery.setFont(new Font("宋体", 1, 20));
CourseQuery.setBackground(Color.CYAN);
CourseQuery.setBounds(600, 400, 80, 45);
// jp6 上的按钮
GradeQuery = new JButton("查询");
GradeQuery.setFont(new Font("宋体", 1, 20));
GradeQuery.setBackground(Color.PINK);
GradeQuery.setBounds(600, 400, 80, 45);
// 按键监听初始化
initial();
}

// 设置各标签信息的方法
private void setLabel() {
    // jp1 上的标签
    InsertID1 = new JLabel("学 号: ");
    InsertID1.setFont(new Font("楷体", 1, 22));
    InsertID1.setBackground(Color.GREEN);
    InsertID1.setBounds(100, 40, 120, 50);
    InsertName1 = new JLabel("姓 名: ");
    InsertName1.setFont(new Font("楷体", 1, 22));
    InsertName1.setBackground(Color.GREEN);
    InsertName1.setBounds(100, 100, 120, 50);
    InsertChinese1 = new JLabel("语文成绩: ");
    InsertChinese1.setFont(new Font("楷体", 1, 22));
    InsertChinese1.setBackground(Color.GREEN);
    InsertChinese1.setBounds(100, 160, 120, 50);
    InsertMath1 = new JLabel("数学成绩: ");

```

```

InsertMath1.setFont(new Font("楷体", 1, 22));
InsertMath1.setBackground(Color.GREEN);
InsertMath1.setBounds(100, 220, 120, 50);
InsertEnglish1 = new JLabel("英语成绩: ");
InsertEnglish1.setFont(new Font("楷体", 1, 22));
InsertEnglish1.setBackground(Color.GREEN);
InsertEnglish1.setBounds(100, 280, 120, 50);
// jp2 上的标签
DeleteID1 = new JLabel("学 号: ");
DeleteID1.setBounds(100, 100, 100, 50);
DeleteID1.setFont(new Font("楷体", 1, 22));
// jp3 上的标签
UpdateID1 = new JLabel("学 号: ");
UpdateID1.setFont(new Font("楷体", 1, 22));
UpdateID1.setBounds(200, 60, 120, 50);
UpdateItem = new JComboBox<>();
UpdateItem.setFont(new Font("楷体", 1, 22));
UpdateItem.setBounds(200, 200, 100, 45);
UpdateItem.addItem("姓名");
UpdateItem.addItem("语文");
UpdateItem.addItem("数学");
UpdateItem.addItem("英语");
// jp4 上的标签
//...
// jp5 上的标签
CourseItem = new JComboBox<>();
CourseItem.setFont(new Font("楷体", 1, 22));
CourseItem.setBounds(100, 40, 140, 45);
CourseItem.addItem("语文");
CourseItem.addItem("数学");
CourseItem.addItem("英语");
// jp6 上的标签
GradeItem = new JComboBox<>();
GradeItem.setFont(new Font("楷体", 1, 22));
GradeItem.setBounds(100, 40, 140, 45);
GradeItem.addItem("语文");
GradeItem.addItem("数学");
GradeItem.addItem("英语");
}

```

// 设置各文本框信息的方法

```

private void setTextField() {
    // jp1 上的文本框
    InsertID2 = new JTextField();
}

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>


```

InsertID2.setFont(new Font("宋体", 1, 23));
InsertID2.setBounds(210, 40, 200, 35);
InsertName2 = new JTextField();
InsertName2.setFont(new Font("宋体", 1, 23));
InsertName2.setBounds(210, 100, 200, 35);
InsertChinese2 = new JTextField();
InsertChinese2.setFont(new Font("宋体", 1, 23));
InsertChinese2.setBounds(210, 160, 200, 35);
InsertMath2 = new JTextField();
InsertMath2.setFont(new Font("宋体", 1, 23));
InsertMath2.setBounds(210, 220, 200, 35);
InsertEnglish2 = new JTextField();
InsertEnglish2.setFont(new Font("宋体", 1, 23));
InsertEnglish2.setBounds(210, 280, 200, 35);
// jp2 上的文本框
DeleteID2 = new JTextField("输入要删除信息的学号");
DeleteID2.setFont(new Font("楷体", 1, 25));
DeleteID2.setBounds(210, 100, 350, 50);
// jp3 上的文本框
UpdateID2 = new JTextField();
UpdateID2.setFont(new Font("楷体", 1, 20));
UpdateID2.setBounds(310, 60, 200, 45);
UpdateContent = new JTextField("更新内容");
UpdateContent.setFont(new Font("楷体", 0, 22));
UpdateContent.setBounds(310, 200, 200, 45);
// jp4 上的文本框
QueryRecordResult = new JTextArea("查询结果: ");
QueryRecordResult.setFont(new Font("楷体", 1, 20));
//QueryRecordResult.setBounds(30, 30, 560, 260);
QueryRecordResult.setEditable(false);
QueryRecordResult.setLineWrap(true); // 当一行文字过多时自动
换行
scroll = new JScrollPane(QueryRecordResult); // 添加滚动条
scroll.setBounds(30, 30, 560, 260);

```

```

scroll.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR
_AS_NEEDED); // 当需要垂直滚动条时显示

```

```

scroll.setHorizontalScrollBarPolicy(ScrollPaneConstants.HORIZONTAL_SCROLL
BAR_AS_NEEDED); // 当需要水平滚动条时显示
IDCondition = new JTextField();
IDCondition.setFont(new Font("宋体", 1, 18));
IDCondition.setBounds(90, 300, 100, 21);
NameCondition = new JTextField();

```

```

NameCondition.setFont(new Font("宋体", 1, 18));
NameCondition.setBounds(90, 330, 100, 21);
ChineseCondition = new JTextField();
ChineseCondition.setFont(new Font("宋体", 1, 18));
ChineseCondition.setBounds(90, 360, 100, 21);
MathCondition = new JTextField();
MathCondition.setFont(new Font("宋体", 1, 18));
MathCondition.setBounds(90, 390, 100, 21);
EnglishCondition = new JTextField();
EnglishCondition.setFont(new Font("宋体", 1, 18));
EnglishCondition.setBounds(90, 420, 100, 21);
IDCondition.setEditable(false);
NameCondition.setEditable(false);
ChineseCondition.setEditable(false);
MathCondition.setEditable(false);
EnglishCondition.setEditable(false);
// jp5 上的文本框
CourseQueryResult = new JTextArea("查询结果: ");
CourseQueryResult.setFont(new Font("楷体", 1, 20));
CourseQueryResult.setEditable(false);
CourseQueryResult.setLineWrap(true);
CourseScroll = new JScrollPane(CourseQueryResult);
CourseScroll.setBounds(20, 100, 560, 340);

CourseScroll.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_AS_NEEDED);

CourseScroll.setHorizontalScrollBarPolicy(ScrollPaneConstants.HORIZONTAL_SCROLLBAR_AS_NEEDED);
// jp6 上的文本框
GradeQueryResult = new JTextArea("查询结果: ");
GradeQueryResult.setFont(new Font("楷体", 1, 20));
GradeQueryResult.setEditable(false);
GradeQueryResult.setLineWrap(true);
GradeScroll = new JScrollPane(GradeQueryResult);
GradeScroll.setBounds(20, 100, 560, 340);

GradeScroll.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_AS_NEEDED);

GradeScroll.setHorizontalScrollBarPolicy(ScrollPaneConstants.HORIZONTAL_SCROLLBAR_AS_NEEDED);
}

```

```

// 设置各面板信息的方法
private void setPanel() {
    jp1 = new JPanel();
    jp2 = new JPanel();
    jp3 = new JPanel();
    jp4 = new JPanel();
    jp5 = new JPanel();
    jp6 = new JPanel();
}

// 设置布局信息的方法
private void setLayout() {
    // 添加 jp1 的组件
    jp1.setLayout(null);
    jp1.add(InsertRecord);
    jp1.add(InsertReset);
    jp1.add(InsertID1);
    jp1.add(InsertName1);
    jp1.add(InsertChinese1);
    jp1.add(InsertMath1);
    jp1.add(InsertEnglish1);
    jp1.add(InsertID2);
    jp1.add(InsertName2);
    jp1.add(InsertChinese2);
    jp1.add(InsertMath2);
    jp1.add(InsertEnglish2);
    // 添加 jp2 上的组件
    jp2.setLayout(null);
    jp2.add>DeleteID1);
    jp2.add>DeleteID2);
    jp2.add>DeleteRecord);
    jp2.add>DeleteReset);
    // 添加 jp3 上的组件
    jp3.setLayout(null);
    jp3.add(UpdateID1);
    jp3.add(UpdateID2);
    jp3.add(UpdateItem);
    jp3.add(UpdateContent);
    jp3.add(UpdateRecord);
    jp3.add(UpdateReset);
    // 添加 jp4 上的组件
    jp4.setLayout(null);
    // jp4.add(QueryRecordResult);
    jp4.add(scroll);

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

        jp4.add(QueryRecord);
        jp4.add(ID);
        jp4.add(Name);
        jp4.add(Chinese);
        jp4.add(Math);
        jp4.add(English);
        jp4.add(IDCondition);
        jp4.add(NameCondition);
        jp4.add(ChineseCondition);
        jp4.add(MathCondition);
        jp4.add(EnglishCondition);
        // 添加 jp5 上的组件
        jp5.setLayout(null);
        jp5.add(CourseItem);
        jp5.add(CourseQuery);
        jp5.add(CourseScroll);
        // 添加 jp6 上的组件
        jp6.setLayout(null);
        jp6.add(GradeQuery);
        jp6.add(GradeItem);
        jp6.add(GradeScroll);
    }

    // 设置选项卡信息的方法
    private void setBase() {
        Base = new JTabbedPane(JTabbedPane.TOP);
        Base.addTab("添加记录", jp1);
        Base.addTab("删除记录", jp2);
        Base.addTab("更新记录", jp3);
        Base.addTab("查找记录", jp4);
        Base.addTab("选课记录", jp5);
        Base.addTab("课程平均分", jp6);
    }

    // 设置主窗口信息的方法
    private void setThis() {
        this.add(Base);
        this.setTitle("学生信息管理系统");
        this.setLocation(300, 200);
        this.setSize(800, 550);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setResizable(false);
        this.setVisible(true);
    }
}

```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

// 设置数据库信息的方法
private void setDB() {
    db = new OperationMySQL();
    // 连接 mysql

    db.setDburl("jdbc:mysql://localhost:3306/StudentInfo?useSSL=false&allowPublicKeyRetrieval=true&serverTimezone=UTC");
    // 加载驱动
    db.setDbdriver("com.mysql.cj.jdbc.Driver");
    // 这里的用户名和密码是要和你的 mysql 对应的, 也是唯一需要更改的地方
    db.setUsername("root");
    db.setPassword("6666");
}

// 初始化
void initial() {
    // 给各按钮添加监听器
    // InsertRecord, InsertReset, DeleteRecord, DeleteReset,
    QueryRecord, UpdateRecord, CourseQuery, GradeQuery;
    InsertRecord.addMouseListener(this);
    InsertReset.addMouseListener(this);
    DeleteRecord.addMouseListener(this);
    DeleteReset.addMouseListener(this);
    QueryRecord.addMouseListener(this);
    UpdateRecord.addMouseListener(this);
    UpdateReset.addMouseListener(this);
    CourseQuery.addMouseListener(this);
    GradeQuery.addMouseListener(this);
    // 给各复选按钮添加监听器
    // ID, Name, Chinese, Math, English
    ID.addItemListener(this);
    Name.addItemListener(this);
    Chinese.addItemListener(this);
    Math.addItemListener(this);
    English.addItemListener(this);
}

@Override
public void mouseClicked(MouseEvent e) {
    // 添加按钮功能
    // 点击重置键则清空文本框
    if (e.getSource().equals(InsertReset)) {
        InsertID2.setText("");
    }
}

```

```

InsertID2.setFont(new Font("宋体", 1, 23));
InsertName2.setText("");
InsertName2.setFont(new Font("宋体", 1, 23));
InsertChinese2.setText("");
InsertChinese2.setFont(new Font("宋体", 1, 23));
InsertMath2.setText("");
InsertMath2.setFont(new Font("宋体", 1, 23));
InsertEnglish2.setText("");
InsertEnglish2.setFont(new Font("宋体", 1, 23));
} else if (e.getSource().equals(InsertRecord)) {
    // 添加记录功能
    String InsertStuID = InsertID2.getText();
    String InsertStuName = InsertName2.getText();
    String InsertStuChinese = InsertChinese2.getText();
    String InsertStuMath = InsertMath2.getText();
    String InsertStuEnglish = InsertEnglish2.getText();
    try {
        db.setRs(db.executeQuery(InsertStuID));
        if (!db.getRs().next()) {
            db.executeInsert(InsertStuID, InsertStuName,
InsertStuChinese, InsertStuMath, InsertStuEnglish);
            JOptionPane.showMessageDialog(this, "添加信息成功!", "数
数据库操作提示",
                JOptionPane.CLOSED_OPTION,
JOptionPane.INFORMATION_MESSAGE, null, null, null);
        } else JOptionPane.showMessageDialog(this, "添加失败", "温馨
提示",
            -1, 1, null, null, null);
    } catch (Exception exception) {
        exception.printStackTrace();
    } finally {
        db.CloseRS();
        db.CloseStmt();
        db.CloseConnection();
    }
} else if (e.getSource().equals>DeleteReset)) {
    // 删除重置功能
    DeleteID2.setText("");
    DeleteID2.setFont(new Font("楷体", 1, 25));
} else if (e.getSource().equals>DeleteRecord)) {
    // 删除功能
    String DeleteStuID = DeleteID2.getText();
    try {
        db.setRs(db.executeQuery>DeleteStuID));

```

```

        if (db.getRs().next()) {
            db.executeDelete(DeleteStuID);
            JOptionPane.showMessageDialog(this, "删除成功!", "数据库
操作提示",
                -1, 1, null, null, null);
        } else JOptionPane.showMessageDialog(this, "删除失败", "温馨
提示",
            -1, 1, null, null, null);
    } catch (Exception exception) {
        exception.printStackTrace();
    }
} else if (e.getSource().equals(UpdateReset)) {
    // 重置更新框功能
    UpdateID2.setText("");
    UpdateID2.setFont(new Font("宋体", 1, 20));
    UpdateContent.setText("");
    UpdateContent.setFont(new Font("宋体", 1, 20));
} else if (e.getSource().equals(UpdateRecord)) {
    // 完成更新功能
    String UpdateStuID = UpdateID2.getText();
    try {
        db.setRs(db.executeQuery(UpdateStuID));
        if (!db.getRs().next()) {
            JOptionPane.showMessageDialog(this, "没有记录无法更新",
                "温馨提示", JOptionPane.CLOSED_OPTION,
JOptionPane.INFORMATION_MESSAGE,
                null, null, null);
        } else {
            String updateItem = null;
            // 更新选项是姓名
            if (UpdateItem.getSelectedItem().toString().equals("姓
名")) {
                updateItem = "Name";
            }
            // 更新的是语文成绩
            else if
(UpdateItem.getSelectedItem().toString().equals("语文")) {
                updateItem = "Chinese";
            }
            // 更新的是数学成绩
            else if
(UpdateItem.getSelectedItem().toString().equals("数学")) {
                updateItem = "Math";
            }
        }
    }
}

```

```

        // 更新的是英语成绩
        else if
        (UpdateItem.getSelectedItem().toString().equals("英语")) {
            updateItem = "English";
        }
        db.executeUpdate(UpdateStuID, updateItem,
UpdateContent.getText());
        JOptionPane.showOptionDialog(this, "更新成功!", "数据库
操作提示",

            -1, 1, null, null, null);
    }
} catch (Exception exception) {
    exception.printStackTrace();
} finally {
    db.CloseRS();
    db.CloseStmt();
    db.CloseConnection();
}
} else if (e.getSource().equals(QueryRecord)) {
    // 完成查询功能
    try {
        // 默认设置各检索条件均为通配符
        String a = "%", b = "%", c = "%", d = "%", f = "%";
        // 如果 ID 选项被选中, 则获得该选项的输入内容
        if (ID.isSelected())
        && !IDCondition.getText().trim().isEmpty()) {
            a = IDCondition.getText();
        }
        // 如果 Name 选项被选中, 则获得该选项的输入内容
        if (Name.isSelected())
        && !NameCondition.getText().trim().isEmpty()) {
            b = NameCondition.getText();
        }
        // 如果 Math 选项被选中, 则获得该选项的输入内容
        if (Math.isSelected())
        && !MathCondition.getText().trim().isEmpty()) {
            d = MathCondition.getText();
        }
        // 如果 English 选项被选中, 则获得该选项的输入内容
        if (English.isSelected())
        && !EnglishCondition.getText().trim().isEmpty()) {
            f = EnglishCondition.getText();
        }
        // 如果 Chinese 选项被选中, 则获得该选项的输入内容

```



```

        if (Chinese.isSelected()
        && !ChineseCondition.getText().trim().isEmpty()) {
            c = ChineseCondition.getText();
        }
        // 根据各选项检索关键字进行查询，并返回结果集
        db.setRs(db.executeQueryByCondition(a, b, c, d, f));
        // 定义结果集中记录条数
        int i = 0;
        QueryRecordResult.setText("查询结果: ");
        // 输出结果集记录
        while (db.getRs().next()) {
            ++i;
            QueryRecordResult.append("\r\n" + "第" + i + "条记录: "
+ "\r\n"
+ "学号: " + db.getRs().getString(1) + "\r\n"
+ "姓名: " + db.getRs().getString(2) + "\r\n"
+ "语文: " + db.getRs().getString(3) + "\r\n"
+ "数学: " + db.getRs().getString(4) + "\r\n"
+ "英语: " + db.getRs().getString(5) +
("\r\n-----"));
        }
        QueryRecordResult.setText(QueryRecordResult.getText() +
            "\r\n" + "共有" + i + "条学生记录");
    } catch (Exception e1) {
        e1.printStackTrace();
    } finally {
        db.CloseRS();
        db.CloseStmt();
        db.CloseConnection();
    }
} else if (e.getSource().equals(CourseQuery)) {
    // 完成选课查询
    String Course = CourseItem.getSelectedItem().toString();
    try {
        db.setRs(db.executeQueryByCourse(Course));
        int count = 0;
        CourseQueryResult.setText("查询结果: ");
        // 输出结果集记录
        while (db.getRs().next()) {
            ++count;
            CourseQueryResult.append("\r\n" + "第" + count + "条记
录" + "\r\n"
+ "学号: " + db.getRs().getString(1) + "\r\n"

```

```

        + "姓名: " + db.getRs().getString(2) + "\r\n"
        + "课程: " + db.getRs().getString(3) + "\r\n"
        + "学分: " + db.getRs().getString(4) +

("\r\n-----");
    }
    CourseQueryResult.setText(CourseQueryResult.getText() +
        "\r\n" + "共有" + count + "条选课记录");
} catch (Exception exception) {
    exception.printStackTrace();
} finally {
    db.CloseRS();
    db.CloseStmt();
    db.CloseConnection();
}
} else if (e.getSource().equals(GradeQuery)) {
    // 完成课程平均分统计查询
    String Course = GradeItem.getSelectedItem().toString();
    try {
        db.setRs(db.executeQueryByGrade(Course));
        int j = 0;
        GradeQueryResult.setText("查询结果: ");
        // 输出查询结果集
        while (db.getRs().next()) {
            ++j;
            GradeQueryResult.append("\r\n" + "第" + j + "条记录" +
"\r\n"
                + "课程: " + db.getRs().getString(1) + "\t" +
                "平均分: " + db.getRs().getString(2) +

("\r\n-----"));
        }
        GradeQueryResult.setText(GradeQueryResult.getText() +
            "\r\n" + "共有" + j + "条记录");
    } catch (Exception exception) {
        exception.printStackTrace();
    } finally {
        db.CloseRS();
        db.CloseStmt();
        db.CloseConnection();
    }
}
}
}

```

```

@Override
public void mousePressed(MouseEvent e) {}

@Override
public void mouseReleased(MouseEvent e) {}

@Override
public void mouseEntered(MouseEvent e) {}

@Override
public void mouseExited(MouseEvent e) {}

@Override
public void itemStateChanged(ItemEvent e) {
    // 如果查询选项 ID 被选中，则可以输入 ID 进行查询
    if (e.getSource().equals(ID)) {
        IDCondition.setEditable(ID.isSelected());
    }
    // 如果选项姓名被选中，则可以输入姓名进行查询
    else if (e.getSource().equals(Name)) {
        NameCondition.setEditable(Name.isSelected());
    }
    // 如果语文被选中，则可以输入语文成绩进行查询
    else if (e.getSource().equals(Chinese)) {
        ChineseCondition.setEditable(Chinese.isSelected());
    }
    // 如果数学选项被选中，则可以输入数学成绩查询
    else if (e.getSource().equals(Math)) {
        MathCondition.setEditable(Math.isSelected());
    }
    // 如果英语选项被选中，则可以输入英语成绩来查询
    else if (e.getSource().equals(English)) {
        EnglishCondition.setEditable(English.isSelected());
    }
}
}

```

五、注意事项

1. 认真填写实验报告
2. 遵守实验室各项制度，服从实验指导教师的安排
3. 按规定的时间完成实验

六、说明

1. 建议学时数 4 学时

