重庆科技学院

上机实验报告(JAVA)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程名称 | 面向对象程序设计方法 | | 实验题目 | 实验6 | | |
| 机房编号 | I305 | | 上机时间 | 2023/11/7 | | |
| 指导老师 | 周召敏 | | 报告序号 | 6 | 上机成绩 |  |
| 学生姓名 | 刘畅 | 学号 | 202442728 | 教学班级 | 物联网一班 | |
| 一、上机目的和要求:  1. 加深对工具包JavaFX的认识；  2. 掌握JavaFX常用布局面板的基本用法；  3. 掌握JavaFX常用控件的基本用法；  4. 理解事件处理机制；  5. 掌握事件监听者的设计和注册方法；  6. 掌握菜单设计的基本方法。 | | | | | | |
| 二、程序开发环境:  eclipse | | | | | | |
| 三、上机内容(老师布置的具体任务):  1. 编写一个GUI程序，窗口标题为“登录”的，能实现用户名和密码的输入及界面跳转。  （1）单击“登录”按钮后验证输入的用户名和密码是否正确，并在控制台输出相关信息：当账号密码为空时，显示“请输入账号密码”、当账号密码正确时，“用户名和密码正确，登录成功”、当账号密码错误时，“用户名或密码错误，请重新输入”；要求正确的账号为自己的姓名全拼，密码为自己的学号；  （2）将登录界面的背景设置为一张图片，同时按钮（Button）插入图片；    （3）分别设置登录按钮、取消按钮、退出按钮，当点击取消按钮时，输入的账号密码重置为空；当点击退出按钮时，退出整个登录界面；当点击登录按钮时，当账号密码正确时跳转至另一个新的界面，同时关闭当前登录界面；  （4）当跳转进入新的界面时，要求在新界面显示当前登录成功的账号和密码信息；  （5）在新界面设置一个退出按钮，当点击退出按钮时，关闭当前新界面，跳转至登录界面。    代码：  import javafx.application.Application;  import javafx.event.ActionEvent;  import javafx.event.EventHandler;  import javafx.geometry.Pos;  import javafx.scene.Group;  import javafx.scene.Scene;  import javafx.scene.control.Button;  import javafx.scene.control.Label;  import javafx.scene.control.TextField;  import javafx.scene.image.Image;  import javafx.scene.input.MouseEvent;  import javafx.scene.layout.\*;  import javafx.stage.Stage;  import java.awt.desktop.AppEvent;  public class javafx01 extends Application {  @Override  public void start(Stage stage) throws Exception {  GridPane gr = new GridPane();  Image image = new Image("pgoto/keli.jpg");  BackgroundImage backgroundImage = new BackgroundImage(image, BackgroundRepeat.NO\_REPEAT, BackgroundRepeat.NO\_REPEAT, BackgroundPosition.CENTER, BackgroundSize.DEFAULT);  Background background = new Background(backgroundImage);  gr.setBackground(background);  Button b1=new Button("登录");  Button b2=new Button("注册");  Button b3=new Button("退出");  Label l1=new Label("账号");  Label l2=new Label("密码");  TextField t1=new TextField();  TextField t2=new TextField();  gr.add(l1,0,0);  gr.add(l2,0,1);  gr.add(t1,1,0);  gr.add(t2,1,1);  gr.add(b1,0,3);  gr.add(b2,1,3);  gr.add(b3,2,3);  b1.setOnMouseClicked(new EventHandler<MouseEvent>() {  @Override  public void handle(MouseEvent mouseEvent) {  if(t1.getText().equals("刘畅")&&t2.getText().equals("2022442728"))  System.out.println("密码正确，登陆成功");  else if(t1.getText().equals(" ")||t2.getText().equals(" "))  System.out.println("账号和密码不能为空");  else  System.out.println("账号或密码错误");  }  });  b3.setOnAction(new EventHandler<ActionEvent>() {  @Override  public void handle(ActionEvent actionEvent) {  stage.close();  }  });  gr.setAlignment(Pos.CENTER);  stage.setTitle("test1");  stage.setScene(new Scene(gr,1000,500));  stage.show();  }  public static void main(String[] args) {  launch();  }  }  运行截图：    2. 参考图5.1，设计一个简单的加法计算器。单击“计算”按钮，把计算结果显示在最右边文本域中。    图5.1  代码：  package test2;  import com.sun.tools.javac.Main;  import javafx.application.Application;  import javafx.event.EventHandler;  import javafx.geometry.Pos;  import javafx.scene.Scene;  import javafx.scene.control.Button;  import javafx.scene.control.Label;  import javafx.scene.control.TextField;  import javafx.scene.input.MouseEvent;  import javafx.scene.layout.GridPane;  import javafx.stage.Stage;  public class jisuanqi extends Application {  @Override  public void start(Stage st) throws Exception {  GridPane gr = new GridPane();  gr.setAlignment(Pos.BASELINE\_CENTER);  TextField t1 = new TextField();  TextField t2 = new TextField();  TextField t3 = new TextField();  t1.setPrefWidth(60);  t2.setPrefWidth(60);  t3.setPrefWidth(60);  Button b=new Button("计算");  b.setOnMouseClicked(new EventHandler<MouseEvent>() {  @Override  public void handle(MouseEvent mouseEvent) {  int a= Integer.parseInt(t1.getText());  int b= Integer.parseInt(t2.getText());  int sum=a+b;  t3.setText(String.valueOf(sum));  }  });  gr.add(t1,0,0);  gr.add(new Label("+"),1,0);  gr.add(t2,2,0);  gr.add(new Label("="),3,0);  gr.add(t3,4,0);  gr.add(b,2,1);  st.setTitle("计算器");  st.setScene(new Scene(gr,400,400));  st.show();  }  public static void main(String[] args) {  launch();  }  }  运行截图    3.设计如图5.2所示的聊天窗口，实现以下功能：    图5.2  （1）单击“发送”按钮，将下方TextField中的信息发送到聊天信息区（TextArea），并清空其中的内容；  （2）关闭窗口前将“聊天记录”保存到磁盘文件中。  代码：  package test3;  import javafx.application.Application;  import javafx.event.ActionEvent;  import javafx.event.EventHandler;  import javafx.scene.Scene;  import javafx.scene.control.Button;  import javafx.scene.control.TextArea;  import javafx.scene.control.TextField;  import javafx.scene.layout.VBox;  import javafx.stage.Stage;  public class lixotian extends Application {  public static void main(String[] args) {  launch(args);  }  @Override  public void start(Stage primaryStage) throws Exception {  TextArea chatArea = new TextArea();  chatArea.setEditable(false);  chatArea.setPrefRowCount(10);  TextField inputArea = new TextField();  inputArea.setPrefColumnCount(20);  Button sendButton = new Button("发送");  sendButton.setOnAction(new EventHandler<ActionEvent>() {  @Override  public void handle(ActionEvent event) {  String message = inputArea.getText();  if (!message.isEmpty()) {  chatArea.appendText(message + "\n");  }  inputArea.clear();  }  });  VBox root = new VBox();  root.setSpacing(10);  root.getChildren().addAll(chatArea, inputArea, sendButton);  Scene scene = new Scene(root, 300, 300);  primaryStage.setTitle("聊天应用");  primaryStage.setScene(scene);  primaryStage.show();  }  }  运行截图： | | | | | | |
| 四、上机调试中出现的错误信息、错误原因及解决办法:  背景图片插入不熟悉 | | | | | | |
| 五、上机实验中的收获及心得:  认识工具包JavaFX的；  了解JavaFX常用布局面板的基本用法；  了解JavaFX常用控件的基本用法；  了解事件处理机制；  了解事件监听者的设计和注册方法；  了解菜单设计的基本方法。 | | | | | | |

建议：正文采用五宋体，代码用小五号Times new Roman字体