# 实验11： JavaFX UI组件和多媒体

## 实验目的和要求

掌握用各种界面组件创建图形用户界面的方法。

掌握Label、Button、CheckBox、RadioButton、TextField和ComboBox使用方法。

掌握Media、MediaPlayer和MediaView观看和播放音频的方法。

## 实验题目

### P578 16.4

原题：（创建一个英里/公里的转换器）编写一个程序来转换英里和公里。如果在英里文本域Mile中输入一个值之后按下回车键，就会在公里文本域Kilometer中显示对应的公里值。同样的，在公里文本域Kilometer中输入一个值之后按下回车键，就会在英里文本域Mile中显示对应的英里值。

源代码：

**package** project;

**import** javafx.application.Application;

**import** javafx.geometry.Pos;

**import** javafx.scene.Scene;

**import** javafx.scene.control.Label;

**import** javafx.scene.control.TextField;

**import** javafx.scene.input.KeyCode;

**import** javafx.scene.layout.GridPane;

**import** javafx.stage.Stage;

**public** **class** **T1** **extends** **Application** {

**private** **TextField** tf1 = **new** TextField();

**private** **TextField** tf2 = **new** TextField();

**public** **static** **void** **main**(**String**[] args) {

// **TODO** Auto-generated method stub

**Application**.*launch*(args);

}

**private** **void** **change1**()

{

**double** **Mile** = **Double**.*parseDouble*(tf1.getText());

**double** **Kilometer** = Mile\*1.6023;

tf2.setText(""+Kilometer);

}

**private** **void** **change2**()

{

**double** **Kilometer** = **Double**.*parseDouble*(tf2.getText());

**double** **Mile** = Kilometer/1.6023;

tf1.setText(""+Mile);

}

***@Override***

**public** **void** **start**(**Stage** ps) **throws** **Exception** {

// **TODO** Auto-generated method stub

**GridPane** **gridPane** = **new** GridPane();

gridPane.setHgap(5);

gridPane.setVgap(5);

gridPane.add(**new** Label("Mile"),0,0);

gridPane.add(tf1,1,0);

gridPane.add(**new** Label("Kilometer"),0,1);

gridPane.add(tf2,1,1);

gridPane.setAlignment(*Pos*.***CENTER***);

tf1.setAlignment(*Pos*.***BOTTOM\_RIGHT***);

tf2.setAlignment(*Pos*.***BOTTOM\_RIGHT***);

tf1.setOnAction(**e**->{change1();});

tf2.setOnAction(**e**->{change2();});

**Scene** **scene** = **new** Scene(gridPane,250,250);

ps.setTitle("ShowTest0");

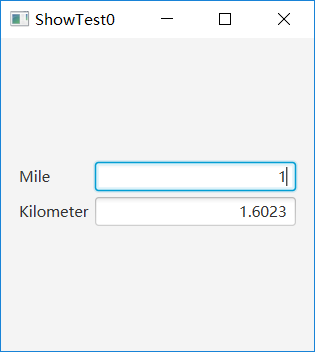
ps.setScene(scene);

ps.show();

}

}

结果及截图：



### P580 16.16

原题：（演示Label的属性）编写一个程序吗，允许用户动态地设置属性contentDisplay和graphicTextGap。

源代码：

**package** project;

**import** javafx.application.Application;

**import** javafx.collections.FXCollections;

**import** javafx.collections.ObservableList;

**import** javafx.geometry.Pos;

**import** javafx.scene.Scene;

**import** javafx.scene.control.ComboBox;

**import** javafx.scene.control.Label;

**import** javafx.scene.control.ListView;

**import** javafx.scene.control.ScrollPane;

**import** javafx.scene.control.SelectionMode;

**import** javafx.scene.layout.BorderPane;

**import** javafx.scene.layout.HBox;

**import** javafx.scene.layout.StackPane;

**import** javafx.stage.Stage;

**public** **class** **T2** **extends** **Application** {

**StackPane** stackPane;

**Label** l1;

**Label** l2;

**private** **String**[] countries = { "Canada", "China", "Danmark", "France", "Germany", "India", "Norway" };

**private** **String**[] items = {"Single","Multiply"};

**ListView**<String> lv;

**ComboBox**<String> cbo;

**public** **static** **void** **main**(**String**[] args) {

// **TODO** Auto-generated method stub

**Application**.*launch*(args);

}

**public** **void** **Single**()

{

lv.getSelectionModel().setSelectionMode(*SelectionMode*.***SINGLE***);

lv.getSelectionModel().selectedItemProperty().addListener(**ov** -> {

**for** (**Integer** **i** : lv.getSelectionModel().getSelectedIndices()) {

l1.setText("Selected items are :" + countries[i]);

}

});

}

**public** **void** **Multiply**()

{

lv.getSelectionModel().setSelectionMode(*SelectionMode*.***MULTIPLE***);

lv.getSelectionModel().selectedItemProperty().addListener(**ov** -> {

**String** **s** = "";

**for** (**Integer** **i** : lv.getSelectionModel().getSelectedIndices()) {

s += countries[i]+" ";

}

l1.setText("Selected items are :" + s);

});

}

***@Override***

**public** **void** **start**(**Stage** ps) **throws** **Exception** {

// **TODO** Auto-generated method stub

l1 = **new** Label();

l1.setAlignment(*Pos*.***CENTER***);

l2 = **new** Label();

l2.setAlignment(*Pos*.***CENTER***);

// l.setText(arg0);

stackPane = **new** StackPane();

cbo = **new** ComboBox<>();

cbo.getItems().addAll(items[0],items[1]);

cbo.setStyle("-fx-color: green");

cbo.setValue("Single");

**HBox** **hBox1** = **new** HBox();

hBox1.getChildren().add(l1);

hBox1.setStyle("-fx-border-color: green");

**HBox** **hBox2** = **new** HBox();

hBox2.getChildren().add(l2);

hBox2.getChildren().add(cbo);

hBox2.setStyle("-fx-border-color: green");

lv = **new** ListView<>(**FXCollections**.*observableArrayList*(countries));

lv.setPrefSize(400, 400);

cbo.setOnAction(**e**->{

**if**(cbo.getValue()==items[0])

{

Single();

}

**else** **if**(cbo.getValue()==items[1])

{

Multiply();

}

});

**BorderPane** **borderPane** = **new** BorderPane();

borderPane.setCenter(stackPane);

borderPane.setBottom(hBox1);

borderPane.setTop(hBox2);

borderPane.setRight(**new** ScrollPane(lv));

**Scene** **scene** = **new** Scene(borderPane, 500, 250);

ps.setTitle("ShowTest1");

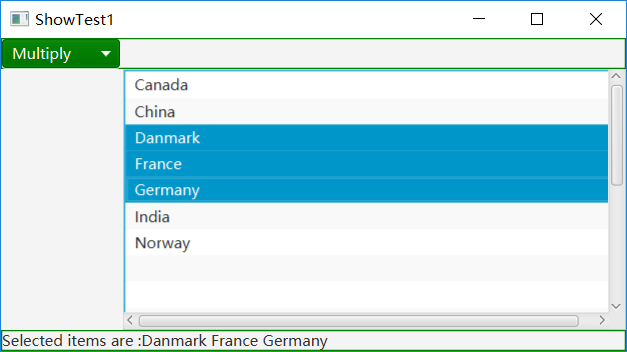
ps.setScene(scene);

ps.show();

}

}

结果及截图：



### P582 16.22

原题：（播放、循环播放和停止播放发一个音频剪辑）编写一个满足下面的程序：

1. 使用AudioClip获取一个音频文件，该文件存放在类目录下。
2. 放置三个标记为Play、Loop和Stop的按钮。
3. 单击Play按钮时，会播放音频文件一次。单击Loop按钮时，会循环播放音频。单击Stop按钮时，停止播放该音频。

源代码：

**package** project;

**import** java.io.File;

**import** java.net.URI;

**import** javafx.application.Application;

**import** javafx.geometry.Pos;

**import** javafx.scene.Scene;

**import** javafx.scene.control.Button;

**import** javafx.scene.control.Label;

**import** javafx.scene.control.Slider;

**import** javafx.scene.layout.BorderPane;

**import** javafx.scene.layout.HBox;

**import** javafx.scene.layout.Region;

**import** javafx.scene.media.Media;

**import** javafx.scene.media.MediaPlayer;

**import** javafx.scene.media.MediaView;

**import** javafx.stage.Stage;

**import** javafx.util.Duration;

**public** **class** **T3** **extends** **Application** {

**File** file = **new** File("e://广东雨神 - 广东十年爱情故事.mp3");

**URI** url = file.toURI();

**private** **String** MEDIA\_URL = url.toString();

***@Override*** // Override the start method in the Application class

**public** **void** **start**(**Stage** primaryStage) {

**Media** **media** = **new** Media(MEDIA\_URL);

**MediaPlayer** **mediaPlayer** = **new** MediaPlayer(media);

**MediaView** **mediaView** = **new** MediaView(mediaPlayer);

**Button** **playButton** = **new** Button("Play");

**Button** **loopButton** = **new** Button("Loop");

playButton.setOnAction(**e** -> {

**if** (playButton.getText().equals("Play")) {

mediaPlayer.play();

playButton.setText("Stop");

} **else** {

mediaPlayer.pause();

playButton.setText("Play");

}

});

**Button** **rewindButton** = **new** Button("Back");

rewindButton.setOnAction(**e** -> mediaPlayer.seek(**Duration**.***ZERO***));

**Slider** **slVolume** = **new** Slider();

slVolume.setPrefWidth(150);

slVolume.setMaxWidth(**Region**.***USE\_PREF\_SIZE***);

slVolume.setMinWidth(30);

slVolume.setValue(50);

mediaPlayer.volumeProperty().bind(

slVolume.valueProperty().divide(100));

**HBox** **hBox** = **new** HBox(10);

hBox.setAlignment(*Pos*.***CENTER***);

hBox.getChildren().addAll(playButton,loopButton, rewindButton,

**new** Label("Volume"), slVolume);

**BorderPane** **pane** = **new** BorderPane();

pane.setCenter(mediaView);

pane.setBottom(hBox);

// Create a scene and place it in the stage

**Scene** **scene** = **new** Scene(pane, 400, 80);

primaryStage.setTitle("MediaDemo"); // Set the stage title

primaryStage.setScene(scene); // Place the scene in the stage

primaryStage.show(); // Display the stage

}

/\*\*

\* The main method is only needed for the IDE with limited

\* JavaFX support. Not needed for running from the command line.

\*/

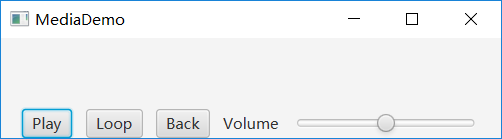
**public** **static** **void** **main**(**String**[] args) {

**Application**.*launch*(args);

}

}

结果及截图：



### P583 16.25

原题：（赛车）编写一个程序，模拟四辆赛车。可以对每辆赛车设置速度吗，用100表示最高速。

源代码：

**package** project;

**import** javafx.animation.KeyFrame;

**import** javafx.animation.Timeline;

**import** javafx.application.Application;

**import** javafx.scene.Scene;

**import** javafx.scene.control.Label;

**import** javafx.scene.control.TextField;

**import** javafx.scene.image.Image;

**import** javafx.scene.image.ImageView;

**import** javafx.scene.layout.BorderPane;

**import** javafx.scene.layout.HBox;

**import** javafx.scene.layout.Pane;

**import** javafx.scene.layout.VBox;

**import** javafx.stage.Stage;

**import** javafx.util.Duration;

**public** **class** **T4** **extends** **Application**{

**Label** l1,l2,l3,l4;

**TextField** t1,t2,t3,t4;

**ImageView** [] cars = **new** **ImageView**[4];

**static** **final** **int** ***maxSpeed*** = 100;

**private** **Timeline** animation;

**public** **static** **void** **main**(**String**[] args) {

// **TODO** Auto-generated method stub

**Application**.*launch*(args);

}

**public** **void** **moveCar**(**int** i)

{

cars[i].setX(cars[i].getX()+10);

}

**public** **void** **setSpeed**(**int** i,**int** speed)

{

moveCar(i);

**if**(speed>0&&speed<=***maxSpeed***)

{

animation = **new** Timeline(**new** KeyFrame(**Duration**.*millis*(50),**e**-> moveCar(i)));

animation.setCycleCount(**Timeline**.***INDEFINITE***);

animation.play();

}

}

***@Override***

**public** **void** **start**(**Stage** ps) **throws** **Exception** {

// **TODO** Auto-generated method stub

l1 = **new** Label("Car 1:");

l2 = **new** Label("Car 2:");

l3 = **new** Label("Car 3:");

l4 = **new** Label("Car 4:");

t1 = **new** TextField();

t1.setOnAction(**e**->{

setSpeed( 0,**Integer**.*parseInt*(t1.getText()));

});

t2 = **new** TextField();

t2.setOnAction(**e**->{

setSpeed( 1,**Integer**.*parseInt*(t2.getText()));

});

t3 = **new** TextField();

t3.setOnAction(**e**->{

setSpeed( 2,**Integer**.*parseInt*(t3.getText()));

});

t4 = **new** TextField();

t4.setOnAction(**e**->{

setSpeed(3, **Integer**.*parseInt*(t4.getText()));

});

**HBox** **hBox** = **new** HBox();

hBox.getChildren().addAll(l1,t1,l2,t2,l3,t3,l4,t4);

**VBox** **vBox** = **new** VBox();

**Pane** **pane1** = **new** Pane();

**Pane** **pane2** = **new** Pane();

**Pane** **pane3** = **new** Pane();

**Pane** **pane4** = **new** Pane();

vBox.getChildren().addAll(pane1,pane2,pane3,pane4);

vBox.setSpacing(20);

cars[0] = **new** ImageView(**new** Image("image/car.gif"));

cars[0].setFitHeight(100);

cars[0].setFitWidth(200);

cars[1] = **new** ImageView(**new** Image("image/car.gif"));

cars[1].setFitHeight(100);

cars[1].setFitWidth(200);

cars[2] = **new** ImageView(**new** Image("image/car.gif"));

cars[2].setFitHeight(100);

cars[2].setFitWidth(200);

cars[3] = **new** ImageView(**new** Image("image/car.gif"));

cars[3].setFitHeight(100);

cars[3].setFitWidth(200);

pane1.getChildren().add(cars[0]);

pane2.getChildren().add(cars[1]);

pane3.getChildren().add(cars[2]);

pane4.getChildren().add(cars[3]);

**BorderPane** **borderPane** = **new** BorderPane();

borderPane.setTop(hBox);

borderPane.setCenter(vBox);

**Scene** **scene** = **new** Scene(borderPane,780,500);

ps.setScene(scene);

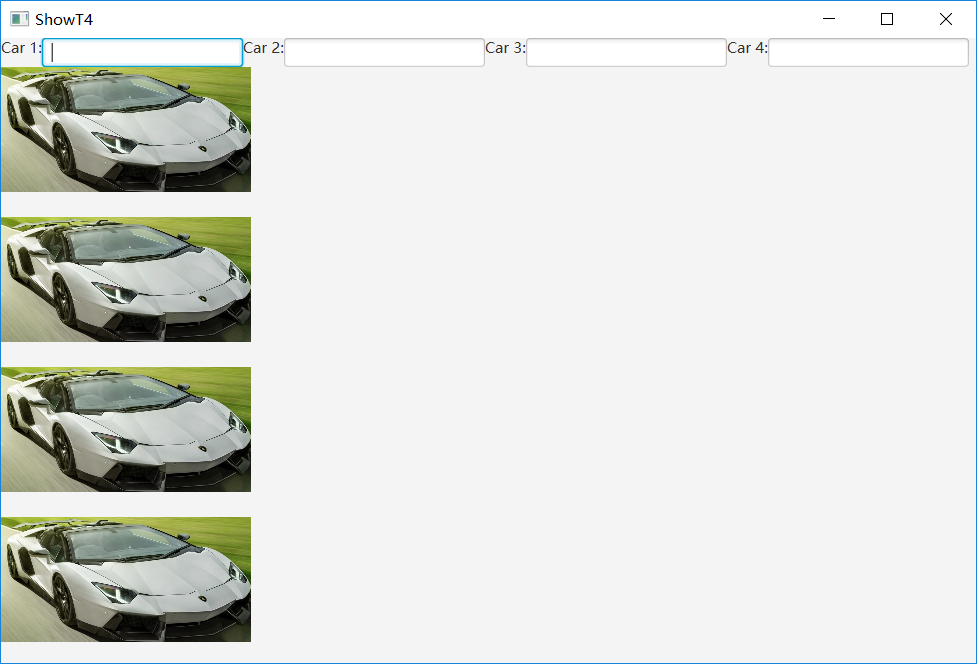
ps.setTitle("ShowT4");

ps.show();

}

}

结果及截图：



### P583 16.27

原题：（显示国旗和国旗描述）程序清单16-8中给出了一个程序，让用户可以从一个组合框中选择国家，从而查看一个国家的国旗以及描述。其中描述是一个写在程序中的字符串。重写这个程序，从文件中读取文本描述，假设这些描述保存在text目录下的文件description0.txt，………，description8.txt中，按照顺序分别表示9个国家：加拿大、中国、丹麦、法国、德国、印度、挪威、英国和美国。

源代码：

**package** project;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.util.Scanner;

**import** javafx.application.Application;

**import** javafx.collections.FXCollections;

**import** javafx.collections.ObservableList;

**import** javafx.scene.Scene;

**import** javafx.scene.control.ComboBox;

**import** javafx.scene.control.Label;

**import** javafx.scene.image.ImageView;

**import** javafx.scene.layout.BorderPane;

**import** javafx.stage.Stage;

**public** **class** **ComboBoxDemo** **extends** **Application** {

// Declare an array of Strings for flag titles

**private** **String**[] flagTitles = {"Canada", "China", "Denmark",

"France", "Germany", "India", "Norway", "United Kingdom",

"United States of America"};

// Declare an ImageView array for the national flags of 9 countries

**private** **ImageView**[] flagImage = {**new** ImageView("image/us.gif"),

**new** ImageView("image/us.gif"),

**new** ImageView("image/us.gif"),

**new** ImageView("image/us.gif"),

**new** ImageView("image/us.gif"),

**new** ImageView("image/us.gif"),

**new** ImageView("image/us.gif"),

**new** ImageView("image/us.gif"), **new** ImageView("image/us.gif")};

// Declare an array of strings for flag descriptions

**private** **String**[] flagDescription = **new** **String**[9];

// Declare and create a description pane

**private** **DescriptionPane** descriptionPane = **new** DescriptionPane();

// Create a combo box for selecting countries

**private** **ComboBox**<String> cbo = **new** ComboBox<>(); // flagTitles);

***@Override*** // Override the start method in the Application class

**public** **void** **start**(**Stage** primaryStage) {

// Set text description

**File** **file0** = **new** File("e://description0.txt");

**String** **s0** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file0);

**while**(sc.hasNext())

{

s0 += sc.next();

s0 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

**File** **file1** = **new** File("e://description1.txt");

**String** **s1** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file1);

**while**(sc.hasNext())

{

s1 += sc.next();

s1 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

**File** **file2** = **new** File("e://description2.txt");

**String** **s2** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file2);

**while**(sc.hasNext())

{

s2 += sc.next();

s2 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

**File** **file3** = **new** File("e://description3.txt");

**String** **s3** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file3);

**while**(sc.hasNext())

{

s3 += sc.next();

s3 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

**File** **file4** = **new** File("e://description4.txt");

**String** **s4** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file4);

**while**(sc.hasNext())

{

s4 += sc.next();

s4 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

**File** **file5** = **new** File("e://description5.txt");

**String** **s5** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file5);

**while**(sc.hasNext())

{

s5 += sc.next();

s5 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

**File** **file6** = **new** File("e://description6.txt");

**String** **s6** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file6);

**while**(sc.hasNext())

{

s6 += sc.next();

s6 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

**File** **file7** = **new** File("e://description7.txt");

**String** **s7** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file7);

**while**(sc.hasNext())

{

s7 += sc.next();

s7 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

**File** **file8** = **new** File("e://description8.txt");

**String** **s8** = "";

**try** {

**Scanner** **sc** = **new** Scanner(file8);

**while**(sc.hasNext())

{

s8 += sc.next();

s8 += " ";

}

sc.close();

} **catch** (**FileNotFoundException** **e1**) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

flagDescription[0] = ""+s0;

flagDescription[1] = ""+s1;

flagDescription[2] = ""+s2;

flagDescription[3] = ""+s3;

flagDescription[4] = ""+s4;

flagDescription[5] = ""+s5;

flagDescription[6] = ""+s6;

flagDescription[7] = ""+s7;

flagDescription[8] = ""+s8;

// Set the first country (Canada) for display

setDisplay(0);

// Add combo box and description pane to the border pane

**BorderPane** **pane** = **new** BorderPane();

**BorderPane** **paneForComboBox** = **new** BorderPane();

paneForComboBox.setLeft(**new** Label("Select a country: "));

paneForComboBox.setCenter(cbo);

pane.setTop(paneForComboBox);

cbo.setPrefWidth(400);

cbo.setValue("Canada");

ObservableList<String> **items** =

**FXCollections**.*observableArrayList*(flagTitles);

cbo.getItems().addAll(items);

pane.setCenter(descriptionPane);

// Display the selected country

cbo.setOnAction(**e** -> setDisplay(items.indexOf(cbo.getValue())));

// Create a scene and place it in the stage

**Scene** **scene** = **new** Scene(pane, 450, 170);

primaryStage.setTitle("ComboBoxDemo"); // Set the stage title

primaryStage.setScene(scene); // Place the scene in the stage

primaryStage.show(); // Display the stage

}

/\*\* Set display information on the description pane \*/

**public** **void** **setDisplay**(**int** index) {

descriptionPane.setTitle(flagTitles[index]);

descriptionPane.setImageView(flagImage[index]);

descriptionPane.setDescription(flagDescription[index]);

}

/\*\*

\* The main method is only needed for the IDE with limited

\* JavaFX support. Not needed for running from the command line.

\*/

**public** **static** **void** **main**(**String**[] args) {

*launch*(args);

}

}

**package** project;

**import** javafx.geometry.Insets;

**import** javafx.scene.control.Label;

**import** javafx.scene.control.ContentDisplay;

**import** javafx.scene.control.ScrollPane;

**import** javafx.scene.control.TextArea;

**import** javafx.scene.image.ImageView;

**import** javafx.scene.layout.BorderPane;

**import** javafx.scene.text.Font;

**public** **class** **DescriptionPane** **extends** **BorderPane** {

/\*\* Label for displaying an image and a title \*/

**private** **Label** lblImageTitle = **new** Label();

/\*\* Text area for displaying text \*/

**private** **TextArea** taDescription = **new** TextArea();

**public** **DescriptionPane**() {

// Center the icon and text and place the text under the icon

lblImageTitle.setContentDisplay(*ContentDisplay*.***TOP***);

lblImageTitle.setPrefSize(200, 100);

// Set the font in the label and the text field

lblImageTitle.setFont(**new** Font("SansSerif", 16));

taDescription.setFont(**new** Font("Serif", 14));

taDescription.setWrapText(**true**);

// taDescription.setEditable(false);

// Create a scroll pane to hold the text area

**ScrollPane** **scrollPane** = **new** ScrollPane(taDescription);

// Place label and scroll pane in the border pane

setLeft(lblImageTitle);

setCenter(scrollPane);

setPadding(**new** Insets(5, 5, 5, 5));

}

/\*\* Set the title \*/

**public** **void** **setTitle**(**String** title) {

lblImageTitle.setText(title);

}

/\*\* Set the image view \*/

**public** **void** **setImageView**(**ImageView** icon) {

lblImageTitle.setGraphic(icon);

}

/\*\* Set the text description \*/

**public** **void** **setDescription**(**String** text) {

taDescription.setText(text);

}

}

结果及截图：

