# 实验10： 事件驱动编程和动画

## 实验目的和要求

理解事件、事件源及事件类。

理解并掌握处理器类、注册处理器对象和源对象，以及编码处理器事件。

掌握内部类及匿名内部类定义处理器方法。

掌握MouseEvent和KeyEvent事件处理。

掌握基本的动画编程方法。

## 实验题目

### P536 15.3

原题：（移动小球）编写一个程序，在面板上移动小球。应该定义一个面板类来进行显示小球，并提供向左、向右、向上和向下移动小球的方法。请进行边界检查以防止球完全移到视线之外。

源代码：

**package** project;

**import** javafx.application.Application;

**import** javafx.event.ActionEvent;

**import** javafx.event.EventHandler;

**import** javafx.geometry.Pos;

**import** javafx.scene.Scene;

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

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

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

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

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

**import** javafx.scene.paint.Color;

**import** javafx.scene.shape.Circle;

**import** javafx.stage.Stage;

**public** **class** **Test1** **extends** **Application**{

**private** **CirclePane** circlePane = **new** CirclePane();

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

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

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

}

***@Override***

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

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

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

hBox.setSpacing(10);

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

**Button** **btn1** = **new** Button("Up");

**Button** **btn2** = **new** Button("Down");

**Button** **btn3** = **new** Button("Left");

**Button** **btn4** = **new** Button("Right");

hBox.getChildren().addAll(btn1,btn2,btn3,btn4);

btn1.setOnAction(**new** EventHandler<ActionEvent>()

{

***@Override***

**public** **void** **handle**(**ActionEvent** e) {

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

circlePane.Up();

}

});

btn2.setOnAction(**new** EventHandler<ActionEvent>()

{

***@Override***

**public** **void** **handle**(**ActionEvent** e) {

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

circlePane.Down();

}

});

btn3.setOnAction(**new** EventHandler<ActionEvent>()

{

***@Override***

**public** **void** **handle**(**ActionEvent** e) {

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

circlePane.Left();

}

});

btn4.setOnAction(**new** EventHandler<ActionEvent>()

{

***@Override***

**public** **void** **handle**(**ActionEvent** e) {

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

circlePane.Right();

}

});

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

borderPane.setBottom(hBox);

borderPane.setCenter(circlePane);

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

**Scene** **scene** = **new** Scene(borderPane,200,150);

primaryStage.setScene(scene);

primaryStage.setTitle("ShowTest1");

primaryStage.show();

}

}

**class** **CirclePane** **extends** **Pane**

{

**private** **Circle** circle = **new** Circle(100,100,10);

**public** **CirclePane**()

{

getChildren().add(circle);

circle.setStroke(**Color**.***BLACK***);

circle.setFill(**Color**.***WHITE***);

}

**public** **void** **Up**()

{

circle.setCenterY(circle.getCenterY()+10);

}

**public** **void** **Down**()

{

circle.setCenterY(circle.getCenterY()-10);

}

**public** **void** **Left**()

{

circle.setCenterX(circle.getCenterX()-10);

}

**public** **void** **Right**()

{

circle.setCenterX(circle.getCenterX()+10);

}

}

**package** project;

**import** javafx.application.Application;

**import** javafx.event.ActionEvent;

**import** javafx.event.EventHandler;

**import** javafx.geometry.Pos;

**import** javafx.scene.Scene;

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

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

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

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

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

**import** javafx.scene.paint.Color;

**import** javafx.scene.shape.Circle;

**import** javafx.stage.Stage;

**public** **class** **Test2** **extends** **Application**{

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

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

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

}

***@Override***

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

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

**Pane** **pane** = **new** Pane();

**Circle** **circle** = **new** Circle(50,50,20);

circle.setStroke(**Color**.***BLACK***);

circle.setFill(**Color**.***RED***);

pane.getChildren().add(circle);

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

hBox.setSpacing(10);

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

**Button** **btn1** = **new** Button("Left");

**Button** **btn2** = **new** Button("Right");

**Button** **btn3** = **new** Button("Up");

**Button** **btn4** = **new** Button("Down");

hBox.getChildren().addAll(btn1,btn2,btn3,btn4);

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

borderPane.setBottom(hBox);

borderPane.setCenter(pane);

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

**Scene** **scene** = **new** Scene(borderPane,400,200);

primaryStage.setScene(scene);

primaryStage.setTitle("ShowTest1");

primaryStage.show();

btn1.setOnAction(**e**->{circle.setCenterX(circle.getCenterX() > 0 ? circle.getCenterX() - 10 : 0);});

btn2.setOnAction(**e**->{circle.setCenterX(circle.getCenterX() < scene.getWidth() ? circle.getCenterX() + 10 : 0);});

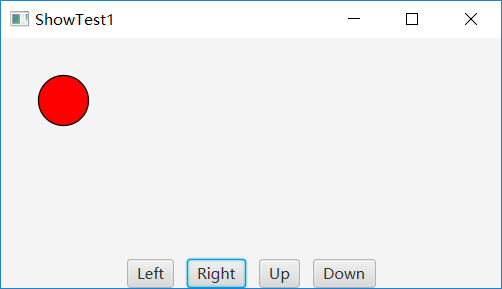
btn3.setOnAction(**e**->{circle.setCenterY(circle.getCenterY() > 0 ? circle.getCenterY() - 10 : 0);});

btn4.setOnAction(**e**->{circle.setCenterY(circle.getCenterY() < scene.getHeight() ? circle.getCenterY() + 10 : 0);});

}

}

结果及截图：



### P536 15.4

原题：（创建一个简单的计算器）编写一个程序完成加法、减法、乘法和除法操作。

源代码：

**package** project;

**import** javafx.application.Application;

**import** javafx.geometry.Insets;

**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.layout.BorderPane;

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

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

**import** javafx.stage.Stage;

**public** **class** **Test3** **extends** **Application** {

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

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

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

}

***@Override***

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

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

**GridPane** **gridpane** = **new** GridPane();

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

gridpane.setPadding(**new** Insets(11.5, 12.5, 13.5, 14.5));

gridpane.setHgap(5.5);

gridpane.setVgap(5.5);

**TextField** **t1** = **new** TextField();

**TextField** **t2** = **new** TextField();

**TextField** **t3** = **new** TextField();

gridpane.add(**new** Label("Number 1:"), 0, 0);

gridpane.add(t1, 1, 0);

gridpane.add(**new** Label("Number 2:"), 0, 1);

gridpane.add(t2, 1, 1);

gridpane.add(**new** Label("Result :"), 0, 2);

gridpane.add(t3, 1, 2);

**Button** **b1** = **new** Button("Add");

**Button** **b2** = **new** Button("Subtract");

**Button** **b3** = **new** Button("Multiply");

**Button** **b4** = **new** Button("Divide");

b1.setOnAction(**e** -> {

**double** **a** = **Double**.*parseDouble*(t1.getText());

**double** **b** = **Double**.*parseDouble*(t2.getText());

t3.setText(**String**.*format*("%.2f", (a+b)));

});

b2.setOnAction(**e** -> {

**double** **a** = **Double**.*parseDouble*(t1.getText());

**double** **b** = **Double**.*parseDouble*(t2.getText());

t3.setText(**String**.*format*("%.2f", (a-b)));

});

b3.setOnAction(**e** -> {

**double** **a** = **Double**.*parseDouble*(t1.getText());

**double** **b** = **Double**.*parseDouble*(t2.getText());

t3.setText(**String**.*format*("%.2f", (a\*b)));

});

b4.setOnAction(**e** -> {

**double** **a** = **Double**.*parseDouble*(t1.getText());

**double** **b** = **Double**.*parseDouble*(t2.getText());

**if**(b==0)

{

**System**.***out***.println("除数不为0！");

**System**.*exit*(1);

}

t3.setText(**String**.*format*("%.2f", (a/b)));

});

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

hBox.getChildren().add(b1);

hBox.getChildren().add(b2);

hBox.getChildren().add(b3);

hBox.getChildren().add(b4);

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

hBox.setSpacing(5);

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

borderPane.setCenter(gridpane);

borderPane.setBottom(hBox);

**Scene** **scene** = **new** Scene(borderPane);

primaryStage.setTitle("ShowTest3");

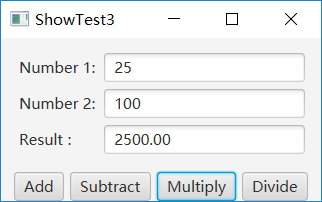
primaryStage.setScene(scene);

primaryStage.show();

}

}

结果及截图：



### P537 15.8

原题：（显示鼠标的位置）编写两个程序，当单击鼠标时显示鼠标的位置，而另一个当按下鼠标时显示鼠标的位置，当释放鼠标时停止显示。

源代码：

一：

**package** project;

**import** javafx.application.Application;

**import** javafx.geometry.Pos;

**import** javafx.scene.Scene;

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

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

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

**import** javafx.stage.Stage;

**public** **class** **Test4** **extends** **Application**{

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

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

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

}

***@Override***

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

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

**StackPane** **stackPane** = **new** StackPane();

**Label** **l** = **new** Label("Label");

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

stackPane.getChildren().add(l);

**StackPane**.*setAlignment*(l, *Pos*.***CENTER***);

stackPane.setOnMouseClicked(**e**->{

l.setText(e.getX()+","+e.getY());

});

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

borderPane.setCenter(stackPane);

**BorderPane**.*setAlignment*(stackPane, *Pos*.***CENTER***);

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

primaryStage.setScene(scene);

primaryStage.setTitle("ShowTest4");

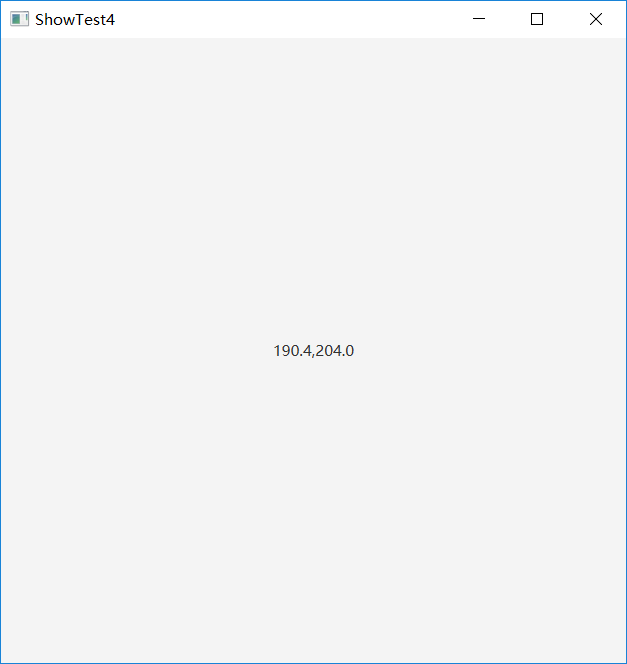
primaryStage.show();

}

}

二：

结果及截图：



### P537 15.10

原题：（输入并显示字符串）请编写一个程序，使用键盘接收一个字符串并把它显示在面板上。回车键表示字符串结束。任何时候输入一个新字符串时都会显示他在面板上。

源代码：

**package** project;

**import** javafx.application.Application;

**import** javafx.scene.Scene;

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

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

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

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

**import** javafx.stage.Stage;

**public** **class** **Test7** **extends** **Application** {

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

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

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

}

***@Override***

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

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

**StackPane** **stackPane** = **new** StackPane();

**TextArea** **ta** = **new** TextArea();

stackPane.getChildren().add(ta);

ta.setOnKeyPressed(**e** -> {

**if** (e.getCode() == *KeyCode*.***ENTER***) {

**System**.*exit*(0);

}

});

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

borderPane.setCenter(stackPane);

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

primaryStage.setScene(scene);

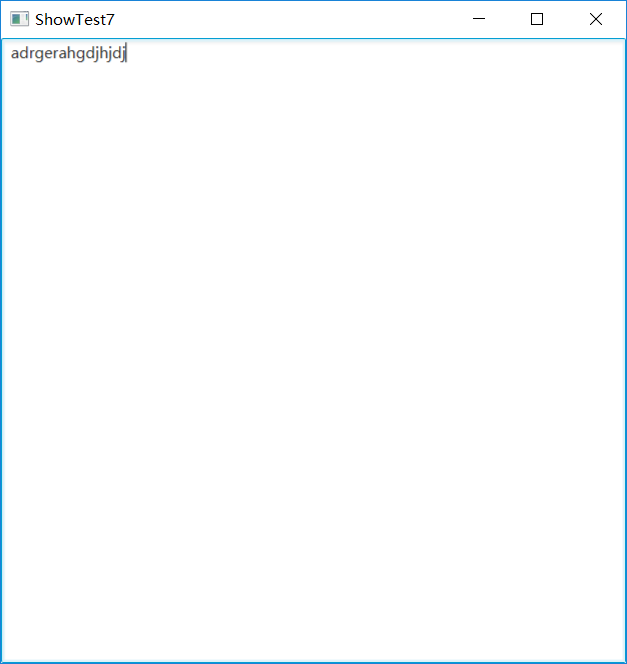
primaryStage.setTitle("ShowTest7");

primaryStage.show();

}

}

结果及截图：



### P538 15.15

原题：（几何问题：添加或删除点）请编写一个程序，让用户在面板上单击以自动创建或移去点。当用户左击鼠标时（主按钮），就创建一个点并且显示在鼠标的位置，用户还可以将鼠标移到一个点上，然后鼠标右击鼠标（次按钮）以移去这个点。

源代码：

**package** project;

**import** java.util.ArrayList;

**import** javafx.application.Application;

**import** javafx.scene.Scene;

**import** javafx.scene.input.MouseButton;

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

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

**import** javafx.scene.paint.Color;

**import** javafx.scene.shape.Circle;

**import** javafx.stage.Stage;

**public** **class** **Test8** **extends** **Application** {

**private** **ArrayList**<Circle> circles = **new** ArrayList();

// private Circle[] circles = new Circle[100];

**private** **static** **int** *i* = 0;

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

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

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

}

***@Override***

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

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

**Pane** **pane** = **new** Pane();

pane.setOnMouseClicked(**e**->{

circles.add(*i*,**new** Circle(e.getX(),e.getY(),5));

circles.get(*i*).setStroke(**Color**.***BLACK***);

circles.get(*i*).setFill(**null**);

**if**(e.getButton()==*MouseButton*.***PRIMARY***)

{

pane.getChildren().add(circles.get(*i*));

*i*++;

}

**else** **if**(e.getButton()==*MouseButton*.***SECONDARY***)

{

**for**(**int** **j**=0;j<*i*;j++)

{

**if**(**Math**.*abs*(e.getX()-circles.get(j).getCenterX())<=5&&**Math**.*abs*(e.getY()-circles.get(j).getCenterY())<=5)

{

pane.getChildren().remove(circles.get(j));

circles.remove(j);

*i*--;

}

}

}

});

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

borderPane.setCenter(pane);

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

primaryStage.setScene(scene);

primaryStage.setTitle("ShowTest8");

primaryStage.show();

}

}

结果及截图：

