1. **TransitionAnimations**

   There are five transition animations in problem 2 of homework 7. When you press the button ones, these five animations will be played in sequence. Please modify the code of homework, and when you press the button, only one animation will be played. Therefore, you have to press the button five times, all five animations will be played for a round.

   Note:
   You can use what you learned in homework to implement this program or use following method to complete it.
   EX:

```
ScaleTransition scaleTransition = ……;
…… // some setting code
scaleTransition.setNode(rectangle);  // apply the settings to rectangle
scaleTransition.play();
```

2. **Ball Animation**

   Please modify the code in problem 1 or problem 3 of homework 7 to meet the following requirements. When the circle hits the top or bottom edge, change its color to red. When the circle hits the left or right edge, change its color to blue.

   Note:
   To apply the settings object of FillTransition to an object, you can use:
   fillTransition.setShape(c); (Not fillTransition.setNode(c))

3. Simple game

Please write a JavaFX program to meet the following requirements.

a) Show one image (in png format) on the pane.

b) Put a start button on the pane, and after user presses the start button, set the button invisible and start the game.

c) The image keeps moving to the right. When it is going to be outside the screen, move it back to the left edge

d) Display the current time (in minutes : seconds: millisecond format) on the top-left corner of the pane.

e) When you press the keyboard:

> Press 'up' or 'w' to move the image up.

> Press 'down' or 's' to move the image down.

You have to build FXML by SceneBuilder.

Note:

a) To show image on screen by ImageView, you can refer to following code.

```
ImageView im;
Image si = new Image;
si = new Image(getClass().getResource(file_path).toExternalForm());
im.setImage(si);
```

b) To set button invisible, you can use:

```
Button bt;
bt.setVisible(false);
```

c) To obtain the location of image, you can use:

```
ImageView im;
int x = im.getLayoutX();
int y = im.getLayoutY();
```

d) To set the location of image, you can use:

```
ImageView im;
int x = im.setLayoutX();
int y = im.setLayoutY();
```

e) When the event function of key-series is invoked, you can use following code to determine which key is pressed.
if(e.getCode()==KeyCode.UP) ……

4. Simple game (Advance)

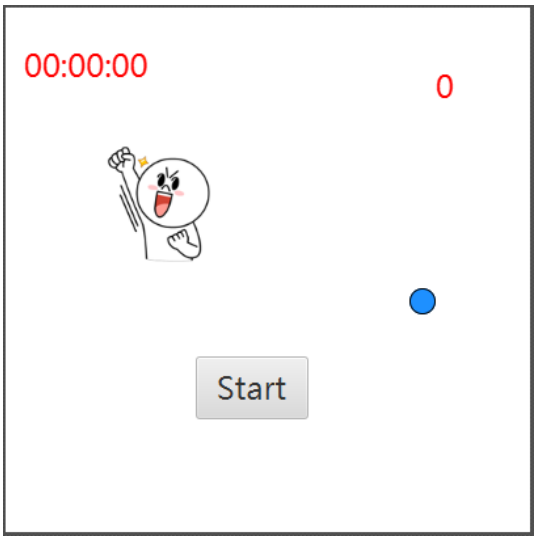
a) Change the static image in problem 3 into an animation. You have to use more than one images to implement the animation, or use transition method to change image.

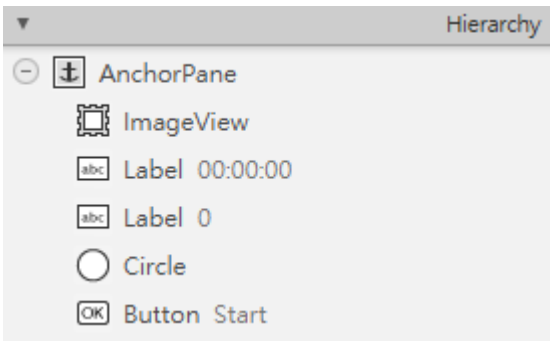
b) Put a circle on the screen. Whenever the animation touches the circle, put the circle at a new random position.

c) Count the number of times the circle is touched, and display it on the top-right corner.

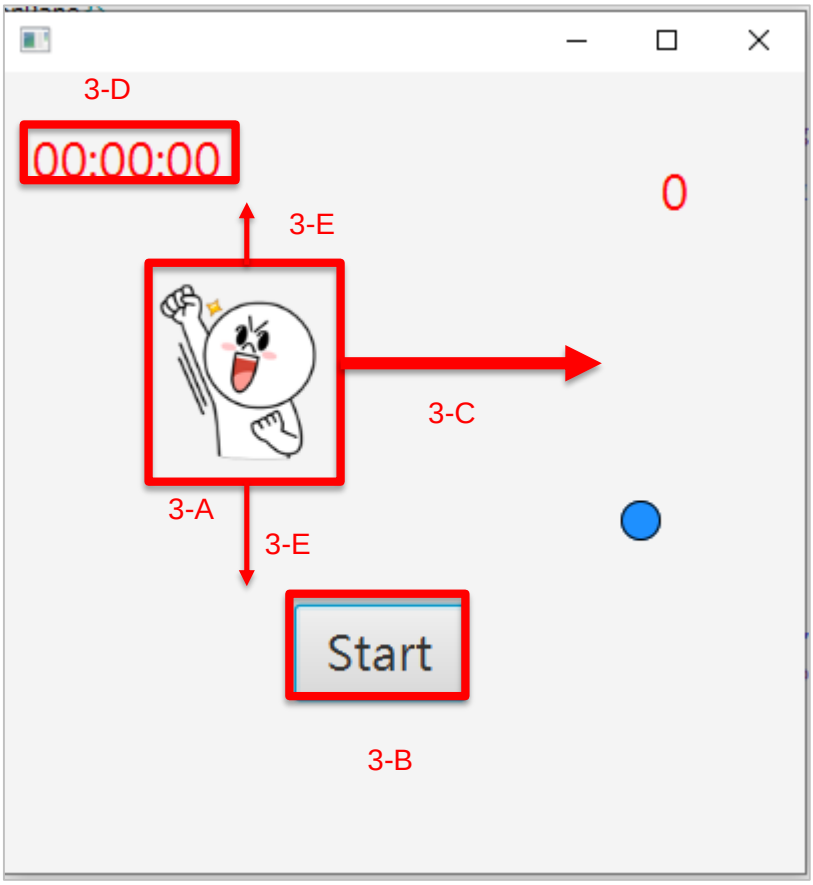
Note:
To play animation with images, you can use Image array to implement it.

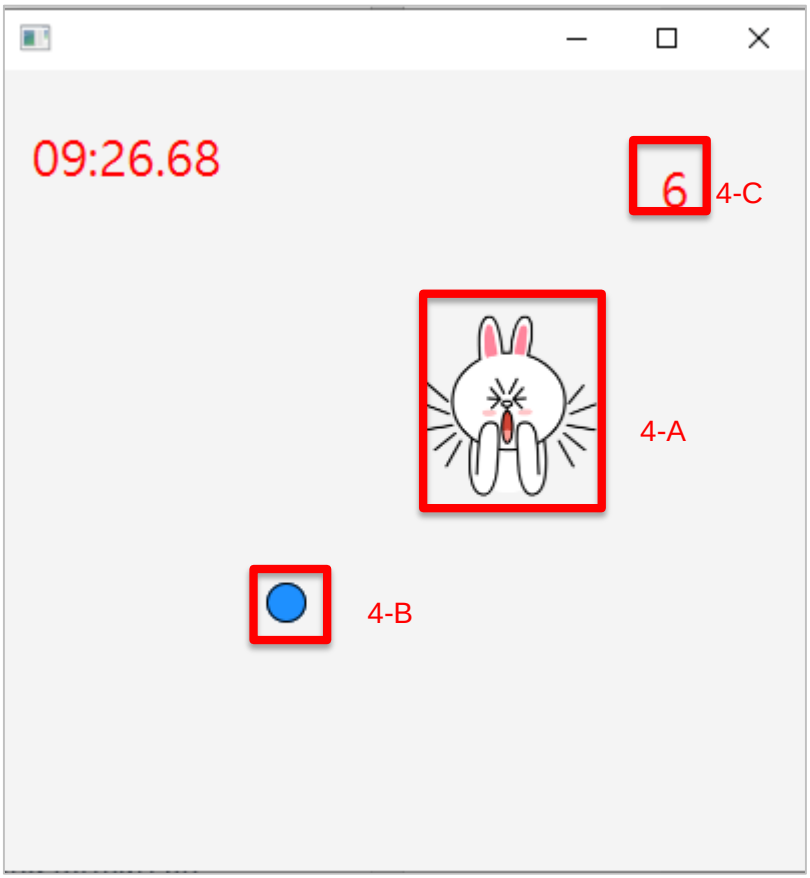
Object Scene and Hierarchy of problem 3 and 4:





The result is as follows.





Basic Sample Code:

```
import java.util.Random;
import javafx.animation.*;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.fxml.FXML;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.scene.input.KeyCode;
import javafx.scene.input.KeyEvent;
import javafx.scene.input.MouseEvent;
import javafx.scene.layout.BorderPane;
import javafx.scene.layout.Pane;
import javafx.scene.paint.Color;
import javafx.scene.shape.Circle;
import javafx.util.Duration;

public class Controller_student {

	/*** fx:id to FXML ***/
	@FXML
	Label clock;
	@FXML
	ImageView image;
	@FXML
	Circle point;	//4-b
	@FXML
	Pane pane;
	@FXML
	Label scorelabel; //4-c
	@FXML
	Button startbutton;
	/************************/



	 //Problem 3-a, 3-c, 3-d, 4-a, 4-b, 4-c
	 public void initialize() {


		//set panel be focusable
		pane.setFocusTraversable(true);
		pane.requestFocus();
	}



	 //Problem3-b
	@FXML
	public void animationStart(ActionEvent e) {


		//set panel be focusable
		pane.setFocusTraversable(true);
		pane.requestFocus();
	}
```

```
        //Problem 3-e
        @FXML
        public void keyPressed(KeyEvent e) {

        }

}
```