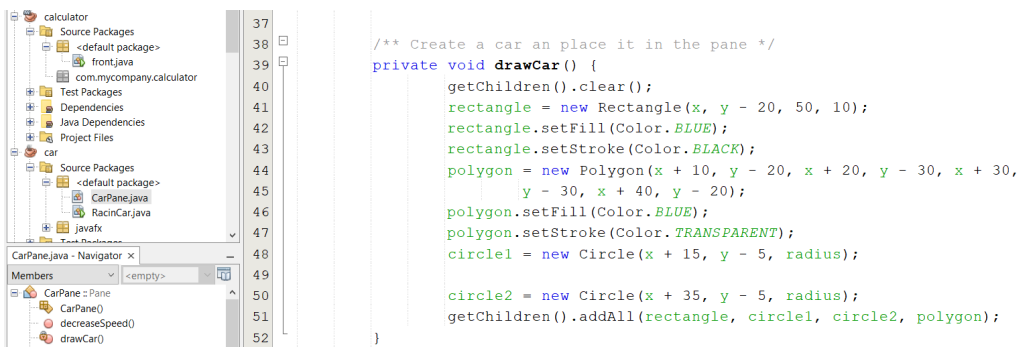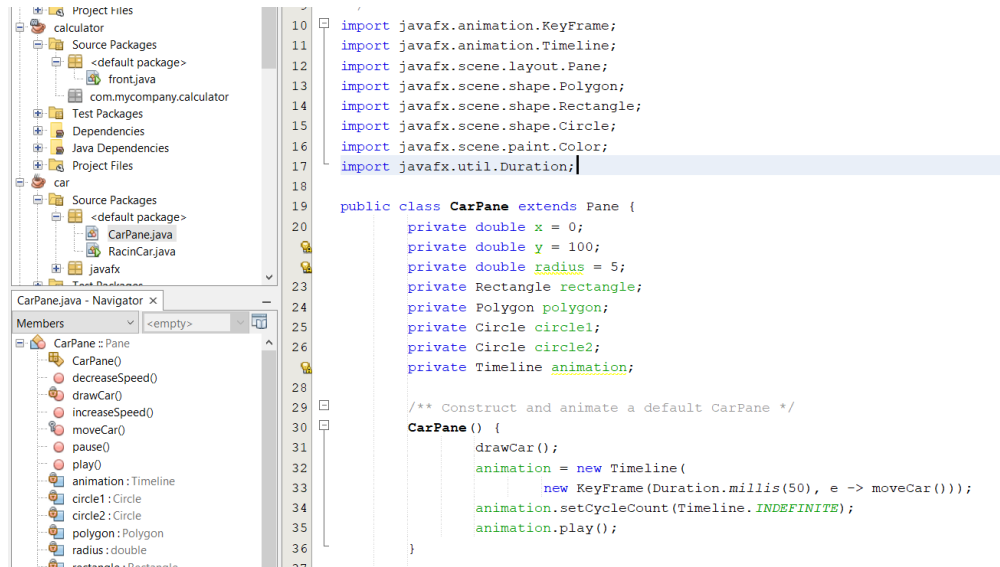# Activity 2: Racing car

Write a program that simulates car racing, as shown in the figure. The car moves from left to right. When it hits the right end, it restarts from the left and continues the same process.
You can use a timer to control animation. Redraw the car with a new base coordinates (x, y).
Also let the user pause/resume the animation with a button press/release and increase/decrease the car speed by pressing the UP and DOWN arrow keys

```java
10    import javafx.animation.KeyFrame;
11    import javafx.animation.Timeline;
12    import javafx.scene.layout.Pane;
13    import javafx.scene.shape.Polygon;
14    import javafx.scene.shape.Rectangle;
15    import javafx.scene.shape.Circle;
16    import javafx.scene.paint.Color;
17    import javafx.util.Duration;
18
19    public class CarPane extends Pane {
20        private double x = 0;
          private double y = 100;
          private double radius = 5;
23        private Rectangle rectangle;
24        private Polygon polygon;
25        private Circle circle1;
26        private Circle circle2;
          private Timeline animation;
28
29        /** Construct and animate a default CarPane */
30        CarPane() {
31            drawCar();
32            animation = new Timeline(
33                    new KeyFrame(Duration.millis(50), e -> moveCar()));
34            animation.setCycleCount(Timeline.INDEFINITE);
35            animation.play();
36        }
```

```java
37
38        /** Create a car an place it in the pane */
39        private void drawCar() {
40            getChildren().clear();
41            rectangle = new Rectangle(x, y - 20, 50, 10);
42            rectangle.setFill(Color.BLUE);
43            rectangle.setStroke(Color.BLACK);
44            polygon = new Polygon(x + 10, y - 20, x + 20, y - 30, x + 30,
45                    y - 30, x + 40, y - 20);
46            polygon.setFill(Color.BLUE);
47            polygon.setStroke(Color.TRANSPARENT);
48            circle1 = new Circle(x + 15, y - 5, radius);
49
50            circle2 = new Circle(x + 35, y - 5, radius);
51            getChildren().addAll(rectangle, circle1, circle2, polygon);
52        }
```

creation of 2 functions for the pause and play action and 2 functions for speed manipulation

```java
54        /** Pause animation */
55        public void pause() {
56            animation.pause();
57        }
58
59        /** Play animation */
60        public void play() {
61            animation.play();
62        }
63
64        /** Increase rate by 1 */
65        public void increaseSpeed() {
66            animation.setRate(animation.getRate() + 1);
67        }
68
69        /** decrease rate by 1 */
70        public void decreaseSpeed() {
71            animation.setRate(animation.getRate() > 0 ? animation.getRate() - 1 : 0);
72        }
73
74        /** Redraw car with new x value */
75        protected void moveCar() {
76            if (x <= getWidth()) {
77                x += 1;
78            }
79            else
80                x = 0;
81            drawCar();
82        }
```

```java
10  import javafx.application.Application;
11  import javafx.scene.Scene;
12  import javafx.stage.Stage;
13  import javafx.scene.input.KeyCode;
14
15  public class RacinCar extends Application {
16      @Override // Override the start method on the Application class
17      public void start(Stage primaryStage) {
18          // Create a car
19          CarPane pane = new CarPane();
20
21          // Create and register handles
22          pane.setOnMousePressed(e -> pane.pause());
23          pane.setOnMouseReleased(e -> pane.play());
24
25          pane.setOnKeyPressed(e -> {
26              if (e.getCode() == KeyCode.UP) {
27                  pane.increaseSpeed();
28              }
29              else if (e.getCode() == KeyCode.DOWN) {
30                  pane.decreaseSpeed();
31              }});
32
33          // Create a scene and place it in the stage
34          Scene scene = new Scene(pane, 600, 100);
35          primaryStage.setTitle("Racing Car"); // Set the stage title
36          primaryStage.setScene(scene); // Place the scene in the stage
37          primaryStage.show(); // Display the stage
38
39          pane.requestFocus(); // Request focus
40      }
41      public static void main(String[] args) {
42          Application.launch(args);
43
44  }
```

Racing Car   — □ ✕