[GUI] Ćwiczenia XIV:

```
package mvcfx;

import javafx.application.Application; // Application
import javafx.stage.*; // Stage
import javafx.scene.*; // Scene

public class Program extends Application {

//  public static void main(String[] args) {
//      launch(args);
//  }

    @Override
    public void init() {
        System.out.println("init: " + Thread.currentThread().getName());
    }

    @Override
    public void start(Stage primaryStage) throws Exception {

        System.out.println("start begin: " + Thread.currentThread().getName());

        Model model = new Model(2024);
        View view = new View();
        Controller controller = new Controller(model, view);

        controller.initView();
        controller.initController();

        // scene, stage
        Scene scene = new Scene(view.getRootNode());
        primaryStage.setScene(scene);
        primaryStage.setTitle("FX");
        primaryStage.show();

        System.out.println("start end: " + Thread.currentThread().getName());

    }

    @Override
    public void stop() {
        System.out.println("stop: " + Thread.currentThread().getName());
    }
}
```

```
package mvcfx;
```

```
public class Model {

    private int x;

    public Model(int x) {
        this.x = x;
    }

    public int getX() {
        return this.x;
    }

    public void setX(int x) {
        this.x += x;
    }

    public void resetX() {
        setX(-this.x);
    }
}
```

```
package mvcfx;

import javafx.event.*;

public class Controller {

    private View view;
    private Model model;

    Controller(Model model, View view) {
        this.model = model;
        this.view = view;
    }

    public void initView() {
         view.getSumLabel().setText("" + model.getX());
    }

    public void initController() {

        view.getSumButtton().addEventHandler(
            /* event type */
            ActionEvent.ACTION,

            /* event handler/"listener" */
            (event) -> {
                int newX = Integer.valueOf(view.getTextField().getText());
                model.setX(newX);
                view.getSumLabel().setText("" + model.getX());
            }
```

```
        );


        // convenience method
        view.getClearButton().setOnAction(
            new EventHandler<ActionEvent>() {

                @Override
                public void handle(ActionEvent event) {
                    view.getTextField().setText("0");
                    model.resetX();
                    view.getSumLabel().setText("0");
                }
            }
        );
    }
}
```

```
package mvcfx;

import javafx.scene.control.*; // Button, Label
import javafx.scene.layout.*; // Pane, GridPane
import javafx.scene.shape.*; // Circle, Rectangle
import javafx.scene.paint.*; // Color
import javafx.geometry.*; // Pos

public class View {

    private GridPane rootNode;
    private TextField textField;
    private Button sumButtton, clearButton;
    private Label sumLabel;

    public GridPane getRootNode() {
        return rootNode;
    }

    public TextField getTextField() {
        return textField;
    }

    public Button getSumButtton() {
        return sumButtton;
    }

    public Button getClearButton() {
        return clearButton;
    }

    public Label getSumLabel() {
        return sumLabel;
```

```
    }

    public View() {

        // left component
        GridPane leftComponent = new GridPane();

        // controls
        textField = new TextField();
        textField.setAlignment(Pos.CENTER);

        sumButtton = new Button("Sum");
        clearButton = new Button("Clear");

        sumLabel = new Label("Label");

        sumLabel.setMaxSize(Double.MAX_VALUE, Double.MAX_VALUE);

        leftComponent.add(textField, 0, 0, 2, 1);
        leftComponent.add(sumButtton, 0, 1);
        leftComponent.add(clearButton, 1, 1);
        leftComponent.add(sumLabel, 0, 2);

        //leftComponent.setGridLinesVisible(true);

        // right component
        Pane rightComponent = new Pane();

        // shapes
        Circle circle1= new Circle(100, 100, 50, Color.TRANSPARENT);
        circle1.setStroke(Color.RED);

        Rectangle rectangle = new Rectangle(0,  0,  100, 100);
        rectangle.setFill(Color.TRANSPARENT);
        rectangle.setStroke(Color.BLUE);

        Circle circle2 = new Circle(150, 150, 50, Color.TRANSPARENT);
        circle2.setStroke(Color.GREEN);

        rightComponent.getChildren().addAll(circle1, rectangle, circle2);

        // root node (panel)
        rootNode = new GridPane();
        rootNode.add(leftComponent, 0, 0);
        rootNode.add(rightComponent, 1, 0);
    }
}
```