# PROGRAMOWANIE W JĘZYKU JAVA

Prowadzący: dr hab. inż. **Jan Prokop**, prof. PRz, e-mail: *jprokop@prz.edu.pl*,
Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki

## LABORATORIUM 4

## Temat: Budowa aplikacji Java FX

JavaFX API - `https://docs.oracle.com/javase/8/javafx/api/toc.htm`

### Zadania

Zbudować aplikacje z poniższych zadań 1 – 3, o wyglądzie sformatowanym za pomocą CSS w wybranym przez siebie wariancie:

- **Kod Java** - w oparciu o przykłady zamieszczone w punkcie 1
- **Kod Java + FXML** - w oparciu o przykłady zamieszczone w punkcie 2

Budowa aplikacji może być realizowana za pomocą narzędzi podstawowych lub w dowolnie wybranym środowisku IDE, np. NetBeans, IntelliJ IDEA, Eclipse, JDeveloper, …. , Scene Builder

**Zadanie 1. Budowa aplikacji o nazwie Java Quiz** (aplikacja stanowiąca test z przedmiotu Java Programming)

Na bazie przykładowego kodu z punktu 1.1. zrealizować aplikację testów z języka Java o następujących funkcjonalnościach (funkcjonalności wybiera prowadzący dla każdej grupy laboratoryjnej):

- Menu, ToolBar, Footer, Login (password field)
- Timer, Tooltip, SplitPane, Set Scene
- Question + Right panel - Image, Java code, Audio, Video, Browser, PieChart, …
- inne podaje prowadzący

**Zadanie 2. Budowa aplikacji o nazwie Java Slide Viewer** (aplikacja będąca przeglądarką slajdów z wykładów z przedmiotu Java Programming)

Na bazie kodu z punktu 1.2. zrealizować aplikację przeglądarki slajdów o następujących funkcjonalnościach (funkcjonalności wybiera prowadzący dla każdej grupy laboratoryjnej) :

- FirstImage, LastImage, wybór slajdu z ComboBox, wybór przedmiotu i numeru wykładu
- Menu, ToolBar, Footer
- 5 slajdów bez logowania i login dla następnych, SlideShow z ustawieniem czasu
- inne podaje prowadzący

**Zadanie 3. Budowa aplikacji o nazwie Java Code Viewer** (aplikacja będąca przeglądarką kodów źródłowych przykładów (*.java) z ćwiczeń laboratoryjnych (right panel) wybieranych ze spisu treści (left panel), jak w książce)

Na bazie kodu z punktu 1.3. zrealizować aplikację o następujących funkcjonalnościach (funkcjonalności wybiera prowadzący dla każdej grupy laboratoryjnej) :

- Menu, ToolBar, Footer, Login
- Left panel - Tree View, Right Panel – kod źródłowy Java + obrazek aplikacji
- Wybór przedmiotu i numeru laboratorium (L01, L02, …)
- inne podaje prowadzący

# 1. Przykłady aplikacji JavaFX – kod Java

## 1.1. Przykład - Kod źródłowy aplikacji JavaFX - `JavaFXQuiz.java`

```
import javafx.application.Application;
import static javafx.application.Application.launch;
import javafx.geometry.Insets;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.CheckBox;
import javafx.scene.layout.VBox;
import javafx.stage.Stage;

public class JavaFXQuiz extends Application {
    @Override
    public void start(Stage primaryStage) {

        Label question = new Label("Java GUI library ?");
        Label response = new Label();
        Button button = new Button("Submit");
        button.setDisable(true);
        CheckBox cb1, cb2, cb3, cb4;
        cb1 = new CheckBox("Swing");
        cb2 = new CheckBox("AWT");
        cb3 = new CheckBox("JavaFX");
        cb4 = new CheckBox("Servlet");
        cb1.setOnAction(e -> button.setDisable(false));
        cb2.setOnAction(e -> button.setDisable(false));
        cb3.setOnAction(e -> button.setDisable(false));
        cb4.setOnAction(e -> button.setDisable(false));
        button.setOnAction(e -> {
            if (cb4.isSelected()) {
                response.setText("Wrong answer");
                button.setDisable(true);
            } else if ((cb1.isSelected()) && (cb2.isSelected()) &&
                                                    (cb3.isSelected())) {
                response.setText("Correct answer");
                button.setDisable(true);
            } else {
                response.setText("Wrong answer");
                button.setDisable(true);
            }
        }
        );
        VBox root = new VBox();
        root.setPadding(new Insets(20, 20, 20, 20));
        root.setSpacing(10);
        root.getChildren().addAll(question, cb1, cb2, cb3, cb4, button, response);
        Scene scene = new Scene(root, 300, 300);
        primaryStage.setTitle("JavaFXQuiz");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

## 1.2. Plik źródłowy aplikacji – `JavaFXImageViewer.java`

```
import java.util.ArrayList;
import java.util.List;
import javafx.application.Application;
import javafx.geometry.Pos;
```

```
import javafx.scene.Scene;
import javafx.scene.Cursor;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.scene.layout.BorderPane;
import javafx.scene.layout.HBox;
import javafx.stage.Stage;

public class JavaFXImageViewer extends Application {

    private final List<String> list = new ArrayList<>();
    int currentImage = 0;
    double clickSceneX, releaseSceneX;
    Button leftButton, rightButton;
    ImageView imageView;
    Label label;

    @Override
    public void start(Stage primaryStage) {
        try {
            list.add("1.jpg");
            list.add("2.jpg");
            list.add("3.jpg");
            list.add("4.jpg");
            list.add("5.jpg");
            BorderPane root = new BorderPane();
            label = new Label(currentImage + 1 + "/" + list.size());
            Image images[] = new Image[list.size()];
            for (int i = 0; i < list.size(); i++) {
                    images[i] = new
                                Image(getClass().getResourceAsStream(list.get(i)));
            }

            imageView = new ImageView(images[currentImage]);
            imageView.setCursor(Cursor.CLOSED_HAND);

            leftButton = new Button("<");
            rightButton = new Button(">");

            leftButton.setOnAction(e -> {
                currentImage = currentImage - 1;
                if (currentImage < 0) {
                    currentImage = list.size() - 1;
                }
                imageView.setImage(images[currentImage]);
                label.setText(currentImage + 1 + "/" + list.size());
            });

            rightButton.setOnAction(e -> {
                currentImage = currentImage + 1;
                if (currentImage >= list.size()) {
                    currentImage = 0;
                }
                imageView.setImage(images[currentImage]);
                label.setText(currentImage + 1 + "/" + list.size());
            });

            imageView.setOnMousePressed(e -> {
                clickSceneX = e.getSceneX();
            });

            imageView.setOnMouseReleased(e -> {
                releaseSceneX = e.getSceneX();
```

```
                if (clickSceneX > releaseSceneX) {
                    leftButton.fire();
                } else {
                    rightButton.fire();
                }
            });
            imageView.setFitWidth(500);
            HBox hBox = new HBox();
            hBox.setSpacing(15);
            hBox.setAlignment(Pos.CENTER);
            hBox.getChildren().addAll(leftButton, imageView, rightButton);
            root.setCenter(hBox);
            BorderPane.setAlignment(label, Pos.CENTER);
            root.setTop(label);

            Scene scene = new Scene(root, 800, 800);
            primaryStage.setTitle("Image Viewer");
            primaryStage.setScene(scene);
            primaryStage.show();
        } catch (Exception e) {
        }
    }

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

### 1.3. Plik źródłowy aplikacji – `JavaFXTextViewer.java`

```
import java.io.*;
import javafx.application.Application;
import javafx.event.*;
import javafx.geometry.Insets;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.scene.layout.*;
import javafx.stage.*;

public class JavaFXTextViewer extends Application {

    BorderPane root = new BorderPane();
    MenuBar mb = new MenuBar();
    VBox top = new VBox();
    VBox center = new VBox();
    TextArea textArea = new TextArea();

    public void createTop(Stage primaryStage) {
        final Stage stage = primaryStage;
        top.setSpacing(20);
        Menu menu = new Menu("File");
        MenuItem open = new MenuItem("Open");
        MenuItem close = new MenuItem("Close");
        MenuItem exit = new MenuItem("Exit");
        menu.getItems().addAll(open, close, exit);
        open.setOnAction((ActionEvent e) -> {
            String currentDir = System.getProperty("user.dir") + File.separator;
            FileChooserBuilder fcb = FileChooserBuilder.create();
            FileChooser fc = fcb.title("Open Dialog").initialDirectory(new
                                                        File(currentDir)).build();
            File selectedFile = fc.showOpenDialog(stage);
            StringBuilder sb = readFile(selectedFile);
            textArea.setText(sb.toString());
        });
        close.setOnAction((ActionEvent e) -> {
```

```
            textArea.setText("");
        });
        exit.setOnAction((ActionEvent e) -> {
            System.exit(0);
        });
        mb.getMenus().addAll(menu);
        top.getChildren().add(mb);
    }
    public void createCenter() {
        center.setPadding(new Insets(10, 10, 10, 10));
        Label label = new Label("File Contents");
        textArea.setWrapText(true);
        textArea.setPrefSize(800, 600);
        center.getChildren().add(label);
        center.getChildren().add(textArea);
    }
    public StringBuilder readFile(File selectedFile) {
        StringBuilder sb = new StringBuilder(1024);
        String curLine = "";
        try {
            FileReader fr = new FileReader(selectedFile);
            BufferedReader br = new BufferedReader(fr);
            while (curLine != null) {
                curLine = br.readLine();
                sb.append(curLine).append("\n");
            }
        } catch (IOException e) {
            e.getMessage();
        }
        return sb;
    }
    @Override
    public void start(Stage primaryStage) {
        createTop(primaryStage);
        createCenter();
        root.setTop(top);
        root.setCenter(center);
        primaryStage.setScene(new Scene(root, 800, 600));
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

### 1.4. Plik źródłowy aplikacji – `JavaFXMultiScene.java`

```
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.layout.VBox;
import javafx.stage.Stage;

public class JavaFXMultiScene extends Application {
    Scene scene1, scene2;
    @Override
    public void start(Stage primaryStage) {
        primaryStage.setTitle("JavaFX Multi Scene GUI");
        //Scene 1
        Label label1 = new Label("This is the first scene");
        Button button1 = new Button("Go to scene 2");
        button1.setOnAction(e -> primaryStage.setScene(scene2));
        VBox layout1 = new VBox(20);
        layout1.getChildren().addAll(button1, label1);
```

```
        scene1 = new Scene(layout1, 300, 250);
        //Scene 2
        Label label2 = new Label("This is the second scene");
        Button button2 = new Button("Go to scene 1");
        button2.setOnAction(e -> primaryStage.setScene(scene1));
        VBox layout2 = new VBox(20);
        layout2.getChildren().addAll(button2, label2);
        scene2 = new Scene(layout2, 300, 250);
        primaryStage.setScene(scene1);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }

}
```

### 1.5. Plik źródłowy aplikacji – JavaFXTreeDemo.java

```
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.control.TreeItem;
import javafx.scene.control.TreeView;
import javafx.scene.layout.StackPane;
import javafx.stage.Stage;

public class JavaFXTreeDemo extends Application {

    @Override
    public void start(Stage primaryStage) {
        TreeItem<String> java = new TreeItem<String>("Java");
        TreeItem<String> lecture = new TreeItem<String>("Lecture");
        TreeItem<String> laboratory = new TreeItem<String>("Laboratory");
        TreeItem<String> project = new TreeItem<String>("Project");
        java.getChildren().addAll(lecture, laboratory, project);
        TreeItem<String> lec1 = new TreeItem<String>("Lec01");
        TreeItem<String> lec2 = new TreeItem<String>("Lec02");
        TreeItem<String> lec3 = new TreeItem<String>("Lec03");
        lecture.getChildren().addAll(lec1, lec2, lec3);
        TreeItem<String> lab1 = new TreeItem<String>("Lab01");
        TreeItem<String> lab2 = new TreeItem<String>("Lab02");
        TreeItem<String> lab3 = new TreeItem<String>("Lab03");
        laboratory.getChildren().addAll(lab1, lab2, lab3);
        TreeView<String> treeView = new TreeView<>(java);
        treeView.getSelectionModel().selectedItemProperty().addListener(
                (observable, oldValue, newValue)
                -> System.out.println("Selected Text : " + newValue.getValue()));
        StackPane root = new StackPane();
        root.getChildren().add(treeView);
        Scene scene = new Scene(root, 300, 250);
        primaryStage.setTitle("JavaPRO");
        primaryStage.setScene(scene);
        primaryStage.show();
    }

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

### 1.6. Plik źródłowy aplikacji – MenuFX.java

```
import javafx.application.Application;
import javafx.application.Platform;
import javafx.scene.Scene;
```

```
import javafx.scene.control.Menu;
import javafx.scene.control.MenuBar;
import javafx.scene.control.MenuItem;
import javafx.scene.control.SeparatorMenuItem;
import javafx.scene.layout.BorderPane;
import javafx.scene.paint.Color;
import javafx.stage.Stage;

public class MenuFX extends Application {

    @Override
    public void start(Stage primaryStage) {
        BorderPane root = new BorderPane();
        Scene scene = new Scene(root, 300, 250, Color.WHITE);
        MenuBar menuBar = new MenuBar();
        menuBar.prefWidthProperty().bind(primaryStage.widthProperty());
        root.setTop(menuBar);
        Menu fileMenu = new Menu("File");
        MenuItem newMenuItem = new MenuItem("New");
        MenuItem saveMenuItem = new MenuItem("Save");
        MenuItem exitMenuItem = new MenuItem("Exit");
        exitMenuItem.setOnAction(actionEvent -> Platform.exit());
        fileMenu.getItems().addAll(newMenuItem, saveMenuItem, new
                                                SeparatorMenuItem(), exitMenuItem);
        Menu editMenu = new Menu("Edit");
        Menu viewMenu = new Menu("View");
        menuBar.getMenus().addAll(fileMenu, editMenu, viewMenu);
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

### 1.7. Plik źródłowy aplikacji – `ToolBarFX.java`

```
import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.ToolBar;
import javafx.scene.control.Separator;
import javafx.stage.Stage;

public class ToolBarFX extends Application {

    @Override
    public void start(Stage primaryStage) {
        Button bt1 = new Button("Button 1");
        bt1.setOnAction(new EventHandler<ActionEvent>() {
            @Override
            public void handle(ActionEvent e) {
                System.out.println("Button 1 clicked!");
            }
        });
        Button bt2 = new Button("Button 2");
        bt2.setOnAction(new EventHandler<ActionEvent>() {
            @Override
            public void handle(ActionEvent e) {
                System.out.println("Button 2 clicked!");
            }
        });
```

```
        ToolBar toolBar = new ToolBar();
        toolBar.getItems().add(bt1);
        toolBar.getItems().add(new Separator());
        toolBar.getItems().add(bt2);
        toolBar.setPrefWidth(600);
        Group root = new Group();
        root.getChildren().add(toolBar);
        Scene scene = new Scene(root, 600, 400);
        primaryStage.setTitle("ToolBar");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

### 1.8. Plik źródłowy aplikacji – JavaFXCheckBoxDemo.java

```
import javafx.application.Application;
import javafx.stage.Stage;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.scene.control.Label;
import javafx.scene.control.CheckBox;
import javafx.scene.text.Font;
import javafx.scene.text.FontWeight;

public class JavaFXCheckBoxDemo extends Application {

    public static void main(String[] args) {

       Application.launch(args);
    }

    public void start(Stage primaryStage) {

      primaryStage.setTitle("CheckBox");

      Group root = new Group();
      Scene scene = new Scene(root, 700, 200);

      Label lab = new Label("CheckBox Unselected");
      lab.setLayoutX(200);
      lab.setLayoutY(100);
      lab.setFont(Font.font("Bell Gothic Std", FontWeight.EXTRA_BOLD, 15));

      CheckBox cb = new CheckBox("CheckBox");
      cb.setIndeterminate(false);
      cb.setLayoutX(100);
      cb.setLayoutY(100);
      cb.setOnAction(new EventHandler<ActionEvent>() {

            public void handle(ActionEvent event) {

                if(cb.isSelected())
                    lab.setText("CheckBox Selected");
                else
                    lab.setText("CheckBox Unselected");
            }
        });

      root.getChildren().add(lab);
```

```
        root.getChildren().add(cb);

        primaryStage.setScene(scene);
        primaryStage.show();
    }
}
```

### 1.9. Plik źródłowy aplikacji – JavaFXComboBoxDemo.java

```
import javafx.application.Application;
import javafx.event.Event;
import javafx.geometry.Insets;
import javafx.scene.Scene;
import javafx.scene.control.ComboBox;
import javafx.scene.control.Label;
import javafx.scene.layout.StackPane;
import javafx.scene.layout.VBox;
import javafx.stage.Stage;

public class JavaFXComboBoxDemo extends Application {

    @Override
    public void start(Stage primaryStage) {

        final Label label = new Label("Select");
        final ComboBox comboBox = new ComboBox();
        comboBox.getItems().addAll("Item 1", "Item 2", "Item 3", "Item 4");

        comboBox.setOnAction((Event event) -> {
            label.setText("Selected: " + comboBox.getValue());
        });

        VBox vBox = new VBox();
        vBox.setPadding(new Insets(20, 20, 20, 20));
        vBox.setSpacing(5);
        vBox.getChildren().addAll(label, comboBox);

        StackPane root = new StackPane();
        root.getChildren().add(vBox);

        Scene scene = new Scene(root, 300, 250);

        primaryStage.setTitle("JavaFXComboBoxDemo");
        primaryStage.setScene(scene);
        primaryStage.show();
    }

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

### 1.10. Plik źródłowy aplikacji – TabPaneSample.java

```
import javafx.application.Application;
import javafx.event.*;
import javafx.geometry.*;
import javafx.scene.*;
import javafx.scene.control.*;
import javafx.scene.layout.*;
import javafx.scene.paint.Color;
import javafx.stage.Stage;

public class TabPaneSample extends Application {
```

```
    public static void main(String[] args) {
        Application.launch(args);
    }
    @Override
    public void start(Stage primaryStage) {
        primaryStage.setTitle("TabPane");
        Group root = new Group();
        Scene scene = new Scene(root, 640, 480, Color.WHITE);
        TabPane tabPane = new TabPane();
        BorderPane borderPane = new BorderPane();
        for (int i = 0; i < 5; i++) {
            Tab tab = new Tab();
            tab.setText("Tab" + i);
            HBox hbox = new HBox();
            hbox.getChildren().add(new Label("Tab" + i));
            hbox.setAlignment(Pos.CENTER);
            tab.setContent(hbox);
            tabPane.getTabs().add(tab);
        }
        tabPane.setSide(Side.LEFT);
        //tabPane.setSide(Side.TOP);
        //tabPane.setSide(Side.RIGHT);
        //tabPane.setSide(Side.BOTTOM)
        borderPane.prefHeightProperty().bind(scene.heightProperty());
        borderPane.prefWidthProperty().bind(scene.widthProperty());
        borderPane.setCenter(tabPane);
        root.getChildren().add(borderPane);
        primaryStage.setScene(scene);
        primaryStage.show();
    }
}
```

### 1.11. Plik źródłowy aplikacji – `SplitPaneFX.java`

```
import javafx.application.Application;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.scene.control.Label;
import javafx.scene.control.SplitPane;
import javafx.scene.layout.StackPane;
import javafx.stage.Stage;
public class SplitPaneFX extends Application {
    @Override
    public void start(Stage primaryStage) {
        Group root = new Group();
        SplitPane sp = new SplitPane();
        final StackPane sp1 = new StackPane();
        sp1.getChildren().add(new Label("1"));
        final StackPane sp2 = new StackPane();
        sp2.getChildren().add(new Label("2"));
        sp.getItems().addAll(sp1, sp2);
        sp.setDividerPositions(0.333f, 0.666f);
        root.getChildren().add(sp);
        Scene scene = new Scene(root, 300, 300);
        sp.prefWidthProperty().bind(scene.widthProperty());
        sp.prefHeightProperty().bind(scene.heightProperty());
        primaryStage.setTitle("SplitPane");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

### 1.12. Plik źródłowy aplikacji `FileChooserFX.java`

```
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.scene.image.*;
import javafx.scene.layout.*;
import javafx.stage.*;
import javax.imageio.*;
public class Main extends Application {
    ImageView myImageView;
    @Override
    public void start(Stage primaryStage) {
        Button btnOpen = new Button("Open");
        btnOpen.setOnAction(btnEventListener);
        myImageView = new ImageView();
        VBox rootBox = new VBox();
        rootBox.getChildren().addAll(btnOpen, myImageView);
        Scene scene = new Scene(rootBox, 300, 300);
        primaryStage.setTitle("FileChooser");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
    EventHandler<ActionEvent> btnEventListener = new EventHandler<ActionEvent>() {
        @Override
        public void handle(ActionEvent t) {
            FileChooser fileChooser = new FileChooser();
            FileChooser.ExtensionFilter extFilterJPG = new
                      FileChooser.ExtensionFilter("JPG files (*.jpg)", "*.JPG");
            FileChooser.ExtensionFilter extFilterPNG = new
                      FileChooser.ExtensionFilter("PNG files (*.png)", "*.PNG");
            fileChooser.getExtensionFilters().addAll(extFilterJPG, extFilterPNG);
            File file = fileChooser.showOpenDialog(null);
            try {
                BufferedImage bufferedImage = ImageIO.read(file);
                Image image = SwingFXUtils.toFXImage(bufferedImage, null);
                myImageView.setImage(image);
            } catch (IOException ex) {
                Logger.getLogger(FileChooserFX.class.getName()).log(Level.SEVERE,
null, ex);
            }
        }
    };
}
```

### 1.13. Plik źródłowy aplikacji – `PieChartSample.java`

```
package piechartsample;
import javafx.application.Application;
import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.scene.Scene;
import javafx.stage.Stage;
import javafx.scene.chart.*;
import javafx.scene.Group;
public class PieChartSample extends Application {
    @Override
    public void start(Stage stage) {
        Scene scene = new Scene(new Group());
        stage.setTitle("Programming languages");
        stage.setWidth(500);
        stage.setHeight(500);
        ObservableList<PieChart.Data> pieChartData
                = FXCollections.observableArrayList(
```

```
                        new PieChart.Data("Java", 17),
                        new PieChart.Data("C", 17),
                        new PieChart.Data("C++", 6),
                        new PieChart.Data("C#", 4),
                        new PieChart.Data("PHP", 3),
                        new PieChart.Data("Python", 3));
        final PieChart chart = new  PieChart(pieChartData);
        chart.setTitle("TIOBE INDEX");
        ((Group) scene.getRoot()).getChildren().add(chart);
        stage.setScene(scene);
        stage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

### 1.14. Plik źródłowy aplikacji – JavaFXWebBrowser.java

```
package javafx1;
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.layout.StackPane;
import javafx.stage.Stage;
import javafx.scene.web.WebEngine;
import javafx.scene.web.WebView;
public class JavaFXWebBrowser extends Application {
    @Override
    public void start(Stage primaryStage) {
        WebView view = new WebView();
        WebEngine engine = view.getEngine();
        engine.load("http://web.prz.edu.pl");
        StackPane root = new StackPane();
        root.getChildren().add(view);
        Scene scene = new Scene(root, 1200, 800);
        primaryStage.setTitle("My Simple Browser !");
        primaryStage.setScene(scene);
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

### 1.15. Plik źródłowy aplikacji - HTMLEditorSample.java

```
package htmleditorsample;
import javafx.application.Application;
import javafx.scene.Scene;
import javafx.scene.web.HTMLEditor;
import javafx.stage.Stage;
public class HTMLEditorSample extends Application {
    private final String INITIAL_TEXT = "<html><body>Lorem ipsum ...</body></html>";
    @Override
    public void start(Stage stage) {
        stage.setTitle("HTMLEditor Sample");
        stage.setWidth(800);
        stage.setHeight(600);
        final HTMLEditor htmlEditor = new HTMLEditor();
        Scene scene = new Scene(htmlEditor);
        htmlEditor.setHtmlText(INITIAL_TEXT);
        stage.setScene(scene);
        stage.show();
    }
    public static void main(String[] args) {
```

```
        launch(args);
    }
}
```

### 1.16. Plik źródłowy aplikacji – JavaFX3D.java

```
import javafx.application.Application;
import javafx.scene.*;
import javafx.scene.paint.Color;
import javafx.scene.paint.PhongMaterial;
import javafx.scene.shape.*;
import javafx.stage.Stage;
public class JavaFX3D extends Application {
   @Override
   public void start(Stage stage) {
      PhongMaterial material = new PhongMaterial();
      material.setDiffuseColor(Color.YELLOW);
      material.setSpecularColor(Color.rgb(30,30,30));
      Sphere sphere = new Sphere(500);
      sphere.setMaterial(material);
      sphere.setTranslateX(500);
      sphere.setTranslateY(500);
      sphere.setTranslateZ(500);
      sphere.setDrawMode(DrawMode.FILL);
      sphere.setCullFace(CullFace.BACK);
      PointLight pointLight = new PointLight(Color.ANTIQUEWHITE);
      pointLight.setTranslateX(800);
      pointLight.setTranslateY(-100);
      pointLight.setTranslateZ(-1000);
      Group root = new Group(sphere);
      root.getChildren().add(pointLight);
      Scene scene = new Scene(root, 1000, 1000, true);
      scene.setFill(Color.rgb(10, 10, 40));
      scene.setCamera(new PerspectiveCamera(false));
      scene.setCamera(new PerspectiveCamera(false));
      stage.setScene(scene);
      stage.setTitle("JavaFX 3D");
      stage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

### 1.17. Plik źródłowy aplikacji – JavaFXAudioClip.java

```
import java.net.URL;
import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.geometry.Insets;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.layout.StackPane;
import javafx.scene.media.AudioClip;
import javafx.stage.Stage;

public class JavaFXAudioClip extends Application {
  public static void main(String[] args) {
    launch(args);
  }
  @Override
  public void start(Stage primaryStage) {
    final URL resource = getClass().getResource("audioclip.wav");
    final AudioClip clip = new AudioClip(resource.toString());
```

```
    final Button button = new Button("Play");
    button.setOnAction(new EventHandler<ActionEvent>() {
      @Override
      public void handle(ActionEvent event) {
        clip.play(1.0);
      }
    });
    final StackPane stackPane = new StackPane();
    stackPane.setPadding(new Insets(10));
    stackPane.getChildren().add(button);
    final Scene scene = new Scene(stackPane, 200, 200);
    primaryStage.setTitle("AudioClip");
    primaryStage.setScene(scene);
    primaryStage.show();
  }
}
```

### 1.18. Plik źródłowy aplikacji – `JavaFXKeyEvent.java`

```
import javafx.application.Application;
import javafx.event.EventHandler;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.scene.control.TextField;
import javafx.scene.input.KeyEvent;
import javafx.stage.Stage;
public class JavaFXKeyEvent extends Application {
  public static void main(String[] args) {
    Application.launch(args);
  }
  @Override
  public void start(Stage primaryStage) {
    Group root = new Group();
    Scene scene = new Scene(root, 300, 300);
    TextField textBox = new TextField();
    textBox.setOnKeyPressed(new EventHandler<KeyEvent>() {
        public void handle(KeyEvent e) {
            System.out.println("Key Pressed: " + e.getText() + "   " + e.getCode()+
                                                              "   "+ e.getEventType());
        }
    });
    textBox.setOnKeyReleased(new EventHandler<KeyEvent>() {
        public void handle(KeyEvent e) {
            System.out.println("Key Released: " + e.getText()+ "   " + e.getCode()+
                                                              "   "+ e.getEventType());
        }
    });
    root.getChildren().add(textBox);
    primaryStage.setScene(scene);
    primaryStage.show();
  }
}
```

### 1.19. JavaFX - CSS

**Zmienić wygląd aplikacji z punktów 1.1 – 1.18 stosując arkusz stylów**

```
scene.getStylesheets().add("stylesheet.css");
```

**Przykładowe formatowanie**

```
.root {
     -fx-background-image: url("background.jpg");
    -fx-background-color: #1d1d1d;
}
```

```
.label {
    -fx-font-size: 12px;
    -fx-font-weight: bold;
    -fx-text-fill: #333333;
    -fx-effect: dropshadow( gaussian , rgba(255,255,255,0.5) , 0,0,0,1 );
    -fx-opacity: 0.6;
}
.button {
    -fx-text-fill: white;
    -fx-font-family: "Arial";
    -fx-font-weight: bold;
    -fx-background-color: linear-gradient(#61a2b1, #2A5058);
    -fx-effect: dropshadow( three-pass-box , rgba(0,0,0,0.6) , 5, 0.0 , 0 , 1 );
}
```

# 2. Przykłady aplikacji JavaFX – FXML

## 2.1. FXML - Aplikacja MVC

- **Model -** `JavaFXApplication.java`

```
package javafxapplication;
import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.stage.Stage;
public class JavaFXApplication extends Application {
    @Override
    public void start(Stage stage) throws Exception {
        Parent root = FXMLLoader.load(getClass().getResource("FXMLDocument.fxml"));
        Scene scene = new Scene(root);
        stage.setScene(scene);
        stage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

- **View -** `FXMLDocument.fxml`

```
<?xml version="1.0" encoding="UTF-8"?>
<?import java.lang.*?>
<?import java.util.*?>
<?import javafx.scene.*?>
<?import javafx.scene.control.*?>
<?import javafx.scene.layout.*?>
<AnchorPane id="AnchorPane" prefHeight="200"
            prefWidth="320"
            xmlns:fx="http://javafx.com/fxml/1"
            fx:controller="javafxapplication.FXMLDocumentController">
    <children>
        <Button layoutX="126" layoutY="90" text="Click Me!"
                onAction="#handleButtonAction" fx:id="button" />
        <Label layoutX="126" layoutY="120"
               minHeight="16" minWidth="69"
                fx:id="label" />
    </children>
</AnchorPane>
```

- **Controller -** `FXMLDocumentController.java`

```
package javafxapplication;
import java.net.URL;
import java.util.ResourceBundle;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.Initializable;
import javafx.scene.control.Label;
public class FXMLDocumentController implements Initializable {
    @FXML
    private Label label;
    @FXML
    private void handleButtonAction(ActionEvent event) {
        System.out.println("You clicked me!");
        label.setText("Hello World!");
    }
    @Override
    public void initialize(URL url, ResourceBundle rb) {
```

```
    }
}
```

## 2.2. FXML - Aplikacja Menu Display Pane

- **Main -** `MyMenuMain.java`

```
import java.io.*;
import javafx.application.Application;
import javafx.fxml.*;
import javafx.scene.*;
import javafx.scene.control.*;
import javafx.scene.layout.*;
import javafx.stage.*;
public class MyMenuMain extends Application {
  private static final BorderPane root = new BorderPane();
  public static BorderPane getRoot() {
    return root;
  }
  @Override
  public void start(Stage stage) throws Exception {
    MenuBar bar = FXMLLoader.load(getClass().getResource("MyMenu.fxml"));
    root.setTop(bar);
    Scene scene = new Scene(root, 600, 400);
    stage.setScene(scene);
    stage.show();
  }
  public static void main(String[] args) {
    launch(args);
  }
}
```

- **Controller -** `MyMenuController.java`

```
import java.io.*;
import javafx.event.*;
import javafx.fxml.*;
import javafx.scene.control.*;
import javafx.scene.layout.*;
public class MyMenuController {
  @FXML
  private MenuItem displayOne;
  @FXML
  private MenuItem displayTwo;
  @FXML
  void switchToOne(ActionEvent event) {
    try {
      AnchorPane paneOne = FXMLLoader.load(getClass().getResource("PaneOne.fxml"));
      BorderPane border = MyMenuMain.getRoot();
      border.setCenter(paneOne);
    }
     catch (IOException e) {
      e.printStackTrace();
    }
  }
  @FXML
  void switchToTwo(ActionEvent event) {
    try {
      AnchorPane paneTwo = FXMLLoader.load(getClass().getResource("PaneTwo.fxml"));
      BorderPane border = MyMenuMain.getRoot();
      border.setCenter(paneTwo);
    }
    catch (IOException e) {
      e.printStackTrace();
    }
  }
```

```
}
```

- **Menu -** `MyMenu.fxml`

```
<?xml version="1.0" encoding="UTF-8"?>
<?import java.lang.*?>
<?import javafx.scene.control.*?>
<MenuBar
  xmlns="http://javafx.com/javafx/8.0"
  xmlns:fx="http://javafx.com/fxml/1"
  fx:controller="MyMenuController">
    <Menu
      mnemonicParsing="false"
      text="Pane">
      <items>
        <MenuItem
          fx:id="displayOne"
          mnemonicParsing="false"
          onAction="#switchToOne"
          text="One" />
        <MenuItem
          fx:id="displayTwo"
          mnemonicParsing="false"
          onAction="#switchToTwo"
          text="Two" />
      </items>
    </Menu>
</MenuBar>
```

- **Panel 1 -** `PaneOne.fxml`

```
<?xml version="1.0" encoding="UTF-8"?>
<?import javafx.scene.text.*?>
<?import javafx.scene.control.*?>
<?import java.lang.*?>
<?import javafx.scene.layout.*?>
<AnchorPane
  prefWidth="600.0"
  prefHeight="400.0"
  xmlns:fx="http://javafx.com/fxml/1"
  xmlns="http://javafx.com/javafx/8.0">
    <Label
      layoutX="150.0"
      layoutY="100.0"
      text="Pane One">
      <font>
        <Font size="65.0" />
      </font>
    </Label>
</AnchorPane>
```

- **Panel 2 -** `PaneTwo.fxml`

```
<?xml version="1.0" encoding="UTF-8"?>
<?import javafx.scene.text.*?>
<?import javafx.scene.control.*?>
<?import java.lang.*?>
<?import javafx.scene.layout.*?>
<AnchorPane
  prefWidth="600.0"
  prefHeight="400.0"
  xmlns:fx="http://javafx.com/fxml/1"
  xmlns="http://javafx.com/javafx/8.0">
    <Label
      layoutX="150.0"
      layoutY="100.0"
      text="Pane Two">
      <font>
```

```
        <Font size="65.0" />
      </font>
    </Label>
</AnchorPane>
```

## 2.3. FXML - Aplikacja Table View

- **Main -** `Main.java`

```
package jpro;
import java.io.IOException;
import java.net.URL;
import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.scene.layout.AnchorPane;
import javafx.stage.Stage;
public class Main extends Application {
    @Override
    public void start(Stage primaryStage) throws IOException {
        URL url = getClass().getResource("TableView.fxml");
        AnchorPane pane = FXMLLoader.load( url );
        Scene scene = new Scene( pane );
        primaryStage.setScene( scene );
        primaryStage.setTitle( "JavaFX TableView" );
        primaryStage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

- **Controller -** `TableController.java`

```
package jpro;
import javafx.collections.ObservableList;
import javafx.fxml.FXML;
import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.cell.PropertyValueFactory;
public class TableController {
    @FXML
    private TableView<Customer> customerTable;
    @FXML
    private TableColumn<Customer, String> firstNameColumn;
    @FXML
    private TableColumn<Customer, String> lastNameColumn;
    @FXML
    private TableColumn<Customer, String> emailNumberColumn;
    @FXML
    protected void initialize() {
        PropertyValueFactory<Customer, String> firstNameProperty =
                new PropertyValueFactory<Customer, String>("firstName");
        PropertyValueFactory<Customer, String> lastNameProperty =
                new PropertyValueFactory<Customer, String>("lastName");
        PropertyValueFactory<Customer, String> emailNumberProperty =
                new PropertyValueFactory<Customer, String>("email");
        firstNameColumn.setCellValueFactory( firstNameProperty );
        lastNameColumn.setCellValueFactory( lastNameProperty );
        emailNumberColumn.setCellValueFactory( emailNumberProperty );
        DataSource data = new DataSource();
        ObservableList<Customer> tableItems = data.getData();
        customerTable.setItems( tableItems );
    }
```

```
}
```

- **Customer -** `Customer.java`

```
package jpro;
import javafx.beans.property.SimpleStringProperty;
public class Customer {
    private final SimpleStringProperty firstName;
    private final SimpleStringProperty lastName;
    private final SimpleStringProperty email;
    public Customer(
            String firstName,
            String lastName,
            String email
    )
    {
        this.firstName = new SimpleStringProperty( firstName );
        this.lastName = new SimpleStringProperty( lastName );
        this.email = new SimpleStringProperty( email );
    }
    public String getFirstName() {
        return firstName.get();
    }
    public void setFirstName(String firstName) {
        this.firstName.set( firstName );
    }
    public String getLastName() {
        return lastName.get();
    }
    public void setLastName(String lastName) {
        this.lastName.set( lastName );
    }
    public String getEmail() {
        return email.get();
    }
    public void setEmail(String phone) {
        this.email.set( phone );
    }
}
```

- **Data -** `DataSource.java`

```
package jpro;
import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
public class DataSource {
    private final ObservableList<Customer> data =
FXCollections.observableArrayList();
    public ObservableList<Customer> getData() {
        return data;
    }
    public DataSource() {
        data.add(new Customer("Jan", "Kowalski", "jk@px.xy.pl"));
        data.add(new Customer("Andrzej", "Nowak", "an@vc.lk.pl"));
        data.add(new Customer("Piotr", "Makowski", "pm@ed.ki.pl"));
    }
}
```

- **View -** `TableView.fxml`

```
<?xml version="1.0" encoding="UTF-8"?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.layout.AnchorPane?>
```

```
<AnchorPane prefHeight="400.0" prefWidth="400.0"
xmlns="http://javafx.com/javafx/8.0.121" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="jpro.TableController">
    <children>
        <TableView fx:id="customerTable" editable="true" layoutX="14.0"
layoutY="14.0" prefHeight="300.0" prefWidth="300.0">
            <columns>
                <TableColumn fx:id="firstNameColumn" prefWidth="104.0" text="First
Name" />
                <TableColumn fx:id="lastNameColumn" prefWidth="121.0" text="Last
Name" />
                <TableColumn fx:id="emailNumberColumn" prefWidth="147.0" text="E-
mail" />
            </columns>
        </TableView>
    </children>
</AnchorPane>
```

## 2.4. FXML - Aplikacja Login

- **Main -** `Main.java`

```
package jp;
import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.stage.Stage;
public class Main extends Application {
    @Override
    public void start(Stage primaryStage) {
        try {
            Parent root = FXMLLoader.load(getClass().getResource("loginPane.fxml"));
            Scene scene = new Scene(root,300,300);
            primaryStage.setTitle("Login Panel");
            primaryStage.setScene(scene);
            primaryStage.show();
        } catch(Exception e) {
            e.printStackTrace();
        }
    }
    public static void main(String[] args) {
        launch(args);
    }
}
```

- **Controller - Controller .** `java`

```
package jp;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.stage.Stage;
public class Controller {
    @FXML
    private Label status;
    @FXML
    private TextField txtUserName;
    @FXML
    private PasswordField txtPassword;
    public void Login(ActionEvent event) throws Exception {
        if (txtUserName.getText().equals("admin") &&
txtPassword.getText().equals("admin")) {
```

```
            status.setText("Login Success");
            Stage stage1 = (Stage) status.getScene().getWindow();
            stage1.close();
            Stage primaryStage = new Stage();
            Parent root = FXMLLoader.load(getClass().getResource("mainPane.fxml"));
            Scene scene = new Scene(root,640,400);
            primaryStage.setScene(scene);
            primaryStage.setTitle("Main Window");
            primaryStage.show();
        } else {
            status.setText("Login Failed");
        }

    }
}
```

- **Panel logowania -** `loginPane.fxml`

```
<?xml version="1.0" encoding="UTF-8"?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.PasswordField?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>
<AnchorPane xmlns="http://javafx.com/javafx/8.0.40"
xmlns:fx="http://javafx.com/fxml/1" fx:controller="jp.Controller">
    <children>
        <Label fx:id="status" layoutX="12.0" layoutY="42.0" text="Status"
textFill="#d02d1b">
            <font>
                <Font size="18.0" />
            </font>
        </Label>
        <TextField fx:id="txtUserName" layoutX="40.0" layoutY="100.0"
promptText="UserName">
            <font>
                <Font size="18.0" />
            </font>
        </TextField>
        <PasswordField fx:id="txtPassword" layoutX="40.0" layoutY="150.0"
promptText="Password">
            <font>
                <Font size="18.0" />
            </font>
        </PasswordField>
        <Button layoutX="120.0" layoutY="250.0" mnemonicParsing="true"
onAction="#Login" text="Login">
            <font>
                <Font size="18.0" />
            </font>
        </Button>
    </children>
</AnchorPane>
```

- **Main Panel -** `mainPane.fxml`

```
<?xml version="1.0" encoding="UTF-8"?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>
<AnchorPane xmlns:fx="http://javafx.com/fxml/1"
xmlns="http://javafx.com/javafx/8.0.40">
    <children>
```

```
    <Label layoutX="250.0" layoutY="200.0" text="Main Window Panel">
        <font>
            <Font size="18.0" />
        </font>
    </Label>
</children>
</AnchorPane>
```

# 3. Wybrane klasy biblioteki JavaFX i ich przykładowe zastosowanie

Zbudować aplikacje języka JavaFX korzystając z następujących klas (szczegóły podaje prowadzący)

- **Klasa javafx.stage.FileChooser**

```
FileChooser fileChooser = new FileChooser();

fileChooser.setTitle("Open Resource File");
fileChooser.getExtensionFilters().addAll(
         new ExtensionFilter("Text Files", "*.txt"),
         new ExtensionFilter("Image Files", "*.png","*.jpg", "*.gif"),
         new ExtensionFilter("Audio Files", "*.wav","*.mp3", "*.aac"),
         new ExtensionFilter("All Files", "*.*"));
File selectedFile = fileChooser.showOpenDialog(mainStage);
if (selectedFile != null) {
    mainStage.display(selectedFile);
}
```

- **Klasa javafx.scene.control.ColorPicker**

```
final ColorPicker colorPicker = new ColorPicker();
colorPicker.setOnAction(new EventHandler() {
   public void handle(Event t) {
      Color c = colorPicker.getValue();
      System.out.println(
      "New Color's RGB = "
      +c.getRed()+" "+
      c.getGreen()+" "+
      c.getBlue());
   }
}
);
```

- **Klasa javafx.scene.control.DatePicker**

```
final DatePicker datePicker = new DatePicker();
datePicker.setOnAction(new EventHandler() {
   public void handle(Event t) {
      LocalDate date = datePicker.getValue();
      System.err.println("Selected date: " + date);
   }
});
```

- **Klasa javafx.scene.control.PasswordField**

```
final PasswordField pb = new PasswordField();
pb.setOnAction(new EventHandler<ActionEvent>() {
      @Override
      public void handle(ActionEvent e) {
        if (!pb.getText().equals("abc")) {
          message.setText("Your password is incorrect!");
          message.setTextFill(Color.web("red"));
        } else {
          message.setText("Your password has been confirmed");
          message.setTextFill(Color.web("black"));
        }
        pb.setText("");
      }
});
```

- **Klasa javafx.scene.control.SplitPane**

```
SplitPane splitPane = new SplitPane();
splitPane.prefWidthProperty().bind(scene.widthProperty());
```

```
splitPane.prefHeightProperty().bind(scene.heightProperty());
splitPane.getItems().add(left);
splitPane.getItems().add(right);
```

- **Klasa javafx.scene.control.Tooltip**

```
Button button = new Button("Hover Over Me");
button.setTooltip(new Tooltip("Tooltip for Button"));
```

- **Klasa javafx.scene.media.MediaPlayer**

```
final URL resource = getClass().getResource("mp3file.mp3");
final Media media = new Media(resource.toString());
final MediaPlayer mediaPlayer = new MediaPlayer(media);
mediaPlayer.play();
```

- **Klasy javafx.scene.control.TreeView, javafx.scene.control.TreeItem**

```
TreeItem<String> root = new TreeItem<String>("Root Node");
root.setExpanded(true);
root.getChildren().addAll(
   new TreeItem<String>("Item 1"),
   new TreeItem<String>("Item 2"),
   new TreeItem<String>("Item 3")
);
TreeView<String> treeView = new TreeView<String>(root);
```

- **Klasa javafx.scene.control.Menu**

```
MenuBar menuBar = new MenuBar();
menuBar.prefWidthProperty().bind(primaryStage.widthProperty());
Menu menu = new Menu("File");
menu.getItems().add(new MenuItem("New"));
menu.getItems().add(new MenuItem("Save"));
menu.getItems().add(new SeparatorMenuItem());
menu.getItems().add(new MenuItem("Exit"));
menuBar.getMenus().add(menu);
menuBar.prefWidthProperty().bind(primaryStage.widthProperty());
    MenuItem newItem = new MenuItem("New", null);
    newItem.setOnAction(new EventHandler<ActionEvent>() {
      public void handle(ActionEvent event) {
        System.out.println("Action");
      }
    });
```

- **Klasa javafx.scene.image.Image**

```
final ImageView selectedImage = new ImageView();
Image image1 = new Image(Main.class.getResourceAsStream("a.jpg"));
selectedImage.setImage(image1);
root.getChildren().addAll(selectedImage);
scene.setRoot(root);
```

## 4. Zadania

Podaje prowadzący w każdej grupie laboratoryjnej