|  |  |  |  |
| --- | --- | --- | --- |
| ***SPRAWOZDANIE NR 8*** | | | |
| Nazwa ćwiczenia | Ćwiczenie IX | | Obraz zawierający tekst  Opis wygenerowany automatycznie |
| Przedmiot | Programowanie obiektowe – laboratorium | |
| Student  grupa | Marcin Ogórkiewicz, grupa 7 | |
| Data ćwiczeń | 22.12.2023 | 29.01.2023 | Data oddania sprawozdania |

**ImageApp**

package com.example.programowanieobiektowe9obrazki;  
  
import javafx.application.Application;  
import javafx.fxml.FXMLLoader;  
import javafx.scene.Parent;  
import javafx.scene.Scene;  
import javafx.stage.Stage;  
  
public class ImageApp extends Application {  
  
 public static void main(String[] args) {  
 *launch*(args);  
 }  
  
 @Override  
 public void start(Stage primaryStage) throws Exception {  
 FXMLLoader loader = new FXMLLoader(getClass().getResource("image-app.fxml"));  
 Parent root = loader.load();  
  
 Scene scene = new Scene(root, 600, 400);  
 primaryStage.setTitle("");  
 primaryStage.setScene(scene);  
 primaryStage.show();  
  
 ImageAppController controller = loader.getController();  
 controller.init();  
 }  
}

**Obraz zawierający człowiek, Ludzka twarz, zrzut ekranu, mięsień

Opis wygenerowany automatycznie**

**Obraz zawierający Ludzka twarz, człowiek, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie**

**Obraz zawierający tekst, zrzut ekranu, komputer, czarne i białe

Opis wygenerowany automatycznie**

**Obraz zawierający zrzut ekranu, człowiek, ubrania, Ludzka twarz

Opis wygenerowany automatycznie**

**ImageAppController**

package com.example.programowanieobiektowe9obrazki;  
  
import javafx.event.ActionEvent;  
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 java.io.File;  
  
public class ImageAppController {  
  
 @FXML  
 private ImageView image\_box;  
  
 @FXML  
 private Button image\_button;  
  
 @FXML  
 private Label image\_label;  
  
 private File[] imageFiles;  
 private int currentImageIndex = 0;  
  
 @FXML  
 private void initialize() {  
 File imagesDirectory = new File("obrazki");  
 imageFiles = imagesDirectory.listFiles();  
 }  
  
 public void init() {  
 updateImageView();  
 }  
  
 @FXML  
 private void handleDisplayButton(ActionEvent event) {  
 currentImageIndex = (currentImageIndex + 1) % imageFiles.length;  
 updateImageView();  
 }  
  
 private void updateImageView() {  
 Image image = new Image(imageFiles[currentImageIndex].toURI().toString());  
 image\_box.setImage(image);  
 image\_label.setText("Obraz " + (currentImageIndex + 1) + " z " + imageFiles.length);  
 }  
}

**image-app**

<?xml version="1.0" encoding="UTF-8"?>  
  
<?import javafx.scene.control.\*?>  
<?import javafx.scene.image.\*?>  
<?import javafx.scene.layout.\*?>  
  
<AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0" prefWidth="600.0" xmlns="http://javafx.com/javafx/17.0.2-ea" xmlns:fx="http://javafx.com/fxml/1" fx:controller="com.example.programowanieobiektowe9obrazki.ImageAppController">  
 <children>  
 <Button fx:id="image\_button" layoutX="268.0" layoutY="347.0" mnemonicParsing="false" onAction="#handleDisplayButton" text="Wyświetl" />  
 <Label fx:id="image\_label" layoutX="268.0" layoutY="33.0" prefHeight="10.0" prefWidth="63.0" text="Obraz 1 z 4" />  
 <ImageView fx:id="image\_box" fitHeight="261.0" fitWidth="330.0" layoutX="135.0" layoutY="70.0" pickOnBounds="true" preserveRatio="true" />  
 </children>  
</AnchorPane>

**Wniosek**

Ćwiczenie nauczyło mnie wyświetlania zdjęć w aplikacjach z graficznym interfejsem użytkownika, w javafx.