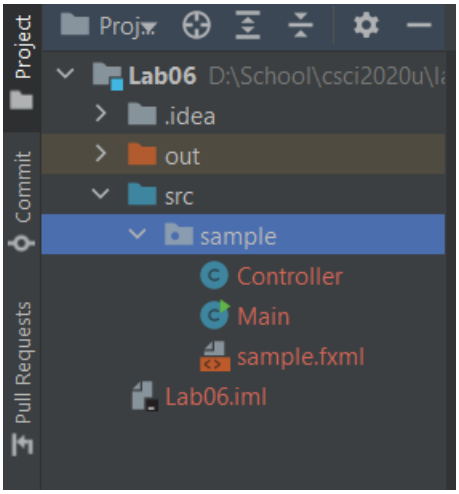
Lab06

Shayne Lewis

100506658

Local Repository



Github Repository

https://github.com/shayne-lewis/csci2020u_shaynelewis/tree/master/lab06

Main.java

```java
package sample;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.text.Text;
import javafx.stage.Stage;

import static java.awt.Font.*;
import static javafx.scene.text.Font.font;

public class Main extends Application {

    @Override
    public void start(Stage primaryStage) throws Exception{
        Parent root = FXMLLoader.load(getClass().getResource( name: "sample.fxml"));
        primaryStage.setTitle("Lab06");
        primaryStage.setScene(new Scene(root,  v: 1000,  v1: 600));
        primaryStage.show();
    }

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

Sample.fxml

```
<?import javafx.geometry.Insets?>
<?import javafx.scene.layout.GridPane?>
<?import javafx.scene.layout.*?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.text.Text?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.control.TableColumn?>

<?import javafx.scene.canvas.Canvas?>
<GridPane alignment="center" hgap="10" vgap="10" xmlns:fx="http://javafx.com/fxml" fx:controller="sample.Controller">

    <Canvas fx:id="mainCanvas" GridPane.columnIndex="0" GridPane.rowIndex="0" height="600" width="1000"></Canvas>

</GridPane>
```

Controller.java

```java
package sample;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.TextField;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.shape.ArcType;
import javafx.scene.text.Text;
import javafx.scene.canvas.Canvas;
import javafx.scene.canvas.GraphicsContext;
import javafx.scene.paint.*;

public class Controller {
    @FXML private Canvas mainCanvas;
    @FXML private GraphicsContext gc;

    private static double[] avgHousingPricesByYear = { 247381.0,264171.4,287715.3,294736.1,308431.4,322635.9,340253.0,363153.7};
    private static double[] avgCommercialPricesByYear = { 1121585.3,1219479.5,1246354.2,1295364.8, 1335932.6,1472362.0,1583521.9,1613246.3};
    private static String[] ageGroups = { "18-25", "26-35", "36-45", "46-55", "56-65", "65+"};
    private static int[] purchasesByAgeGroup = { 648, 1021, 2453, 3173, 1868, 2247};
    private static Color[] pieColours = { Color.AQUA, Color.GOLD, Color.DARKORANGE, Color.DARKSALMON, Color.LAWNGREEN, Color.PLUM};

    @FXML public void initialize(){
        gc = mainCanvas.getGraphicsContext2D();
        drawGraph( w: 120,  h: 300, avgHousingPricesByYear, avgCommercialPricesByYear, Color.RED, Color.BLUE,  startX: 150);
        drawPie( x: 300,  y: 300, purchasesByAgeGroup, pieColours);
    }

    public void drawGraph(int w, int h, double[] data, double[] data2, Color color, Color color2, int startX){
        gc.setFill(color);
        double xInc = w / data.length;
        double maxVal = Double.NEGATIVE_INFINITY;
        double minVal = Double.MAX_VALUE;
        for(double val : data){
            if(val > maxVal) maxVal = val;
            if(val < minVal) minVal = val;
        }
        for(double val : data2){
            if(val > maxVal) maxVal = val;
            if(val < minVal) minVal = val;
        }
        //plot
        double x = startX;
        for(double val : data){
            double height = ((val - minVal) / (maxVal - minVal)) * h;
            gc.fillRect( v: x - 80,  v1: (h - height) + 200, xInc,  v3: height+20);
            x += xInc * 2.5;
        }
        gc.setFill(color2);

        x = 150 + xInc;
        for(double val : data2){
            double height = ((val - minVal) / (maxVal - minVal)) * h;
            gc.fillRect( v: x - 80,  v1: (h - height) + 200, xInc,  v3: height+20);
            x += xInc * 2.5;
        }
    }
    public void drawPie(int x, int y, int[] numbers, Color[] colors){
        double sum = 0;
        double startAng = 0;
        for(int entry : numbers){
            sum += entry;
        }
        for(int i = 0; i < numbers.length; i++){
            gc.setFill(colors[i]);
            double chartFill = numbers[i]/sum;
            double endAng = startAng + chartFill * 360;
            gc.fillArc( v: x+300,  v1: y-100,  v2: 300,  v3: 300,startAng,  v5: endAng - startAng, ArcType.ROUND);
            startAng = endAng;
        }
    }
}
```

Output Window