**Experiment – 8**

Q1) Write a program that displays five texts vertically, as shown in Figure. Set a random color and opacity for each text and set the font of each text to Times Roman, bold, italic, and 22 pixels.



Ans:

**Program:**

**package** Exp\_8;

**import** javafx.application.Application;

**import** javafx.geometry.Pos;

**import** javafx.stage.Stage;

**import** javafx.scene.Scene;

**import** javafx.scene.Group;

**import** javafx.scene.paint.Color;

**import** javafx.scene.text.Font;

**import** javafx.scene.text.FontPosture;

**import** javafx.scene.text.FontWeight;

**import** javafx.scene.text.Text;

**import** javafx.scene.layout.\*;

**public** **class** First **extends** Application{

**public** **static** **void** main(String[] args) {

*launch*(args);

}

@Override

**public** **void** start(Stage stage) **throws** Exception {

Text t1 = **new** Text();

Text t2 = **new** Text();

Text t3 = **new** Text();

Text t4 = **new** Text();

Text t5 = **new** Text();

t1.setText("Java");

t1.setFont(Font.*font*("Arial", FontWeight.***BOLD***, FontPosture.***ITALIC***, 24));

t1.setFill(Color.***DARKRED***);

t1.setRotate(90);

t2.setText("Java");

t2.setFont(Font.*font*("Arial", FontWeight.***BOLD***, FontPosture.***ITALIC***, 24));

t2.setFill(Color.***BLACK***);

t2.setRotate(90);

t3.setText("Java");

t3.setFont(Font.*font*("Arial", FontWeight.***BOLD***, FontPosture.***ITALIC***, 24));

t3.setFill(Color.***BLUE***);

t3.setRotate(90);

t4.setText("Java");

t4.setFont(Font.*font*("Arial", FontWeight.***BOLD***, FontPosture.***ITALIC***, 24));

t4.setFill(Color.***AQUA***);

t4.setRotate(90);

t5.setText("Java");

t5.setFont(Font.*font*("Arial", FontWeight.***BOLD***, FontPosture.***ITALIC***, 24));

t5.setFill(Color.***GOLD***);

t5.setRotate(90);

HBox hbox = **new** HBox(10);

hbox.getChildren().addAll(t1,t2,t3,t4,t5);

hbox.setAlignment(Pos.***CENTER***);

StackPane root = **new** StackPane();

root.getChildren().add(hbox);

Scene scene = **new** Scene(root,300,150);

stage.setScene(scene);

stage.setTitle("Exercise 14\_04");

stage.show();

}

}

**Output:**

A screenshot of a computer

AI-generated content may be incorrect.

Q2) Write a program that uses a bar chart to display the percentages of the overall grade represented by projects, quizzes, midterm exams, and the final exam, as shown in Figure b. Suppose that projects take 20 percent and are displayed in red, quizzes take 10 percent and are displayed in blue, midterm exams take 30 percent and are displayed in green, and the final exam takes 40 percent and is displayed in orange. Use the Rectangle class to display the bars. Interested readers may explore the JavaFXBarChart class for further study.

A screenshot of a quiz

AI-generated content may be incorrect.

Ans:

**Program:**

**package** Exp\_8;

**import** javafx.scene.chart.\*;

**import** javafx.application.Application;

**import** javafx.geometry.Pos;

**import** javafx.stage.Stage;

**import** javafx.scene.Scene;

**import** javafx.scene.Group;

**import** javafx.scene.PerspectiveCamera;

**import** javafx.scene.paint.Color;

**import** javafx.scene.paint.PhongMaterial;

**import** javafx.scene.shape.Box;

**import** javafx.scene.text.Font;

**import** javafx.scene.text.FontPosture;

**import** javafx.scene.text.FontWeight;

**import** javafx.scene.text.Text;

**import** javafx.scene.layout.\*;

**public** **class** Second **extends** Application{

**public** **static** **void** main(String[] args) {

*launch*(args);

}

**public** **void** start(Stage stage) **throws** Exception {

CategoryAxis xAxis = **new** CategoryAxis();

NumberAxis yAxis = **new** NumberAxis();

BarChart<String, Number> barChart = **new** BarChart<>(xAxis, yAxis);

XYChart.Series<String, Number> dataSeries = **new** XYChart.Series<>();

dataSeries.setName("Survey");

dataSeries.getData().add(**new** XYChart.Data<>("Project", 20));

dataSeries.getData().add(**new** XYChart.Data<>("Quiz", 10));

dataSeries.getData().add(**new** XYChart.Data<>("Midterm", 30));

dataSeries.getData().add(**new** XYChart.Data<>("Final", 40));

yAxis.setTickLabelsVisible(**false**);

yAxis.setTickMarkVisible(**false**);

yAxis.setOpacity(0);

barChart.getData().add(dataSeries);

Scene scene = **new** Scene(barChart, 600, 400);

stage.setTitle("Exercise 14\_12");

stage.setScene(scene);

stage.show();

}

}

**Output:**

A screenshot of a graph

AI-generated content may be incorrect.

Q3) Write a program that displays a rectanguloid, as shown in Figure a. The cube should grow and shrink as the window grows or shrinks.

A screenshot of a computer

AI-generated content may be incorrect.

Ans:

**Program:**

**package** Exp\_8;

**import** javafx.application.Application;

**import** javafx.scene.Scene;

**import** javafx.scene.canvas.Canvas;

**import** javafx.scene.canvas.GraphicsContext;

**import** javafx.scene.layout.Pane;

**import** javafx.stage.Stage;

**public** **class** Third **extends** Application {

@Override

**public** **void** start(Stage primaryStage) {

Canvas canvas = **new** Canvas(400, 400);

GraphicsContext gc = canvas.getGraphicsContext2D();

// Cube coordinates

**double** size = 100;

**double** cx = 200; // Center X

**double** cy = 200; // Center Y

**double** offset = size / 2;

// Front face

gc.strokeLine(cx - size/2, cy - size/2, cx + size/2, cy - size/2); // Top

gc.strokeLine(cx + size/2, cy - size/2, cx + size/2, cy + size/2); // Right

gc.strokeLine(cx + size/2, cy + size/2, cx - size/2, cy + size/2); // Bottom

gc.strokeLine(cx - size/2, cy + size/2, cx - size/2, cy - size/2); // Left

// Back face

gc.strokeLine(cx - size/2 - offset, cy - size/2 - offset, cx + size/2 - offset, cy - size/2 - offset); // Top

gc.strokeLine(cx + size/2 - offset, cy - size/2 - offset, cx + size/2 - offset, cy + size/2 - offset); // Right

gc.strokeLine(cx + size/2 - offset, cy + size/2 - offset, cx - size/2 - offset, cy + size/2 - offset); // Bottom

gc.strokeLine(cx - size/2 - offset, cy + size/2 - offset, cx - size/2 - offset, cy - size/2 - offset); // Left

// Connect front and back

gc.strokeLine(cx - size/2, cy - size/2, cx - size/2 - offset, cy - size/2 - offset); // Top-left

gc.strokeLine(cx + size/2, cy - size/2, cx + size/2 - offset, cy - size/2 - offset); // Top-right

gc.strokeLine(cx + size/2, cy + size/2, cx + size/2 - offset, cy + size/2 - offset); // Bottom-right

gc.strokeLine(cx - size/2, cy + size/2, cx - size/2 - offset, cy + size/2 - offset); // Bottom-left

Pane pane = **new** Pane(canvas);

Scene scene = **new** Scene(pane, 400, 400);

primaryStage.setTitle("Cube");

primaryStage.setScene(scene);

primaryStage.show();

}

**public** **static** **void** main(String[] args) {

*launch*(args);

}

}

**Output:**

A screenshot of a computer

AI-generated content may be incorrect.