# TASK DESCRIPTION

This program basically shows the cluster data in the form of scatter chart. It takes data values from text files and plots on to the scatter chart to visualize the data. Whole program is completed by following steps:

1. Text file has been read and stores all the data in array list in the form of objects.
2. Then, unique cluster names have been separated.
3. Scatter chart is defined and its properties have been set.
4. All the data stored in array list is taken and stored in individual series of data according to cluster label names.
5. Then, this series data is passed to scatter chart to plot the data to visualize.

# TASK OUTPUT

|  |  |
| --- | --- |
| **Test Data** | **Screenshot** |
| Notes: | add -Xlint:unchecked to complie the java files  eg:  javac -Xlint:unchecked ClusterPlot.java  java ClusterPlot |
| Small screen |  |
| Full screen |  |

# TASK CODE

**import** java.io.File;

**import** java.util.ArrayList;

**import** java.util.Scanner;

**import** javafx.application.Application;

**import** javafx.scene.Scene;

**import** javafx.scene.chart.NumberAxis;

**import** javafx.scene.chart.ScatterChart;

**import** javafx.scene.chart.XYChart;

**import** javafx.scene.chart.XYChart.Series;

**import** javafx.stage.Stage;

**public** **class** ClusterPlot **extends** Application {

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

{

ArrayList<Cluster> clustersData = *readClusterFile*(); // class arraylist to store all cluster data objects

ArrayList<String> labelNames = **new** ArrayList<String>(); // arraylist to store all unique cluster names from the data

labelNames.add(0, clustersData.get(0).getLabel());

// using for loop in all cluster name list, stores unique cluster names to array to cluster the data on graph

**for**(**int** i = 0; i < clustersData.size(); i++)

{

**if**(!labelNames.contains(clustersData.get(i).getLabel()))

{

labelNames.add(clustersData.get(i).getLabel());

}

}

stage.setTitle("Scatter Chart Cluster Sample"); // sets stage title

NumberAxis xAxis = **new** NumberAxis(0, 8, 2); // defines x-axis values

NumberAxis yAxis = **new** NumberAxis(0, 10, 1); // defines y-axis values

ScatterChart<Number,Number> scatterChart = **new** ScatterChart<Number,Number>(xAxis,yAxis); // defining scatterchart with two parameters x-axis and y-axis

xAxis.setLabel("X-Axis"); // sets x-axis label

yAxis.setLabel("Y-Axis"); // sets y-axis label

scatterChart.setTitle("Scatter Chart Cluster Sample"); // sets title for scatter chart

// using for loop through all the data and storing values of x-axis and y-axis according to cluster name

**for**(**int** i = 0; i < labelNames.size(); i++)

{

// creates series to store data

XYChart.Series series = **new** XYChart.Series();

series.setName(labelNames.get(i)); // sets series name

**for**(**int** j = 0; j < clustersData.size(); j++)

{

**if**(labelNames.get(i).equals(clustersData.get(j).getLabel()))

{

// adds x-axis and y-axis values according to cluster name in series

series.getData().add(**new** XYChart.Data(clustersData.get(j).getXAxis(), clustersData.get(j).getYAxis()));

}

}

// adding all the series to scatter chart

scatterChart.getData().addAll(series);

}

Scene scene = **new** Scene(scatterChart, 800, 500); // adding scatter chart to scene

stage.setScene(scene); // adding scene to stage

stage.show(); // displays stage

}

// defining method to read the cluster data from the file

**public** **static** ArrayList<Cluster> readClusterFile() **throws** Exception

{

ArrayList<Cluster> clustersList = **new** ArrayList<Cluster>(); // creating class array list to store cluster data objects

// reads file

File clusterFile = **new** File("Cluster.txt");

Scanner scan = **new** Scanner(clusterFile); // scans the file

scan.nextLine(); // scans first line

// using while loop to get all the data and store it

**while**(scan.hasNext())

{

Cluster clusters = **new** Cluster(); // defines object of class cluster to store the data

String[] clusterFileDetails = (scan.nextLine().trim()).split("\\s+"); // splits data with space

clusters.setXAxis(Double.*parseDouble*(clusterFileDetails[0])); // stores x-axis value

clusters.setYAxis(Double.*parseDouble*(clusterFileDetails[1])); // stores y-axis value

clusters.setLabel(clusterFileDetails[2]); // stores label attached with the record

clustersList.add(clusters); // adds the object to arraylist of clusters object

}

scan.close();

**return** clustersList;

}

**public** **static** **void** main(String[] args) **throws** Exception

{

*launch*(args);

}

}

// defines class to store cluster data records as object

**class** Cluster {

// declaring variables to store values of x-axis, y-axis and cluster label name

**private** **double** xAxis;

**private** **double** yAxis;

**private** String label;

// defining no-args constructor

**public** Cluster()

{

}

// defining set() method to set value of x-axis

**public** **void** setXAxis(**double** newXAxis)

{

xAxis = newXAxis;

}

// defining get() mehtod to get value of x-axis

**public** **double** getXAxis()

{

**return** xAxis;

}

// defining set() method to set value of y-axis

**public** **void** setYAxis(**double** newYAxis)

{

yAxis = newYAxis;

}

// defining get() mehtod to get value of y-axis

**public** **double** getYAxis()

{

**return** yAxis;

}

// defining set() method to set value of label name

**public** **void** setLabel(String newLabel)

{

label = newLabel;

}

// defining get() mehtod to get value of label name

**public** String getLabel()

{

**return** label;

}

}