# TASK: DISPLAY THE RECORDS OF CLUSTERS (35 MARKS)

A clustering technique called "Unique Neighborhood Set Parameter Independent Density-Based Clustering with Outlier Detection" (PIDC-O) is applied on a dataset having 450 records and two attributes (X, Y). The number of clusters produced by PIDC-O from the dataset is four namely Cluster1, Cluster2, Cluster3 and Cluster4. A text file (**cluster.txt**) that contains the clusters will be uploaded into the subject interact site soon. For better understanding, the structure of the text file is discussed using the table below.

|  |  |  |
| --- | --- | --- |
| **X** | **Y** | **Cluster** |
| 2 | 4 | Cluster1 |
| 6 | 8 | Cluster2 |
| 7 | 8 | CLuster2 |
| 3 | 4 | Cluster1 |
| 2 | 5 | CLuster1 |
| 3 | 5 | Cluster1 |
| 7 | 9 | Cluster2 |
| 2.5 | 4.5 | Cluster1 |
| 6.5 | 8.5 | Cluster2 |

The number of records in the above table is nine, the number of attributes is two (X,Y) and the number of clusters is two (Cluster1, Cluster2) whereas the **cluster.txt** in the subject interact site will have 450 records, two attributes (X,Y) and four clusters (Cluster1, Cluster2, Cluster3, Cluster4).

Write a Java Graphical User Interface (GUI) program that will read the **cluster.txt** file and will display the records of the clusters in **cluster.txt** file in GUI. The records in a particular will be displayed using a single colour. Add legend for each cluster in the GUI, so that the clusters can be easily identified.

# REPORT

## File Directory

For task two, there is only one class PIDCO.java and Cluster.txt. the file should be placed in src folder of java project.

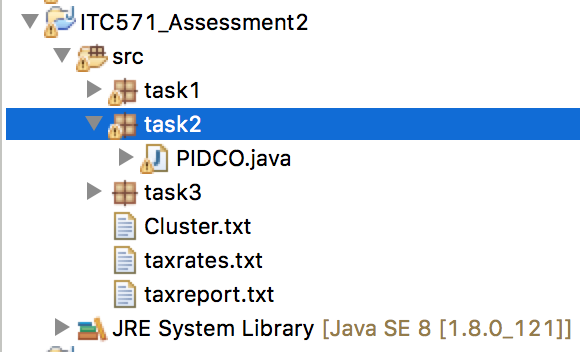


Figure 25: File directory for task 2

The program reads cluster.txt file from src folder as shown in Figure 26



Figure 26: Reading file path.

## Array List for Each Cluster

Program makes four array list to store the entries of each cluster, as shown in Figure 27.

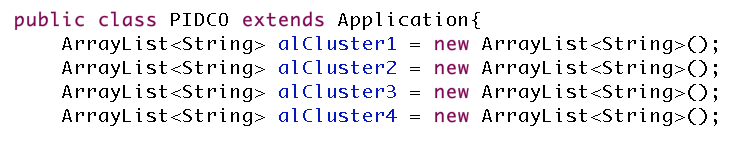


Figure 27: Array lists for each cluster.

## Cluster Data Stored in Format String x$y

Then program detects which cluster the entry belongs to and put that entry in the relevant array list, as shown in Figure 28. The entries are stored in the form of string and the format is: “x$y”, that is x and y values are separated by $ sign.

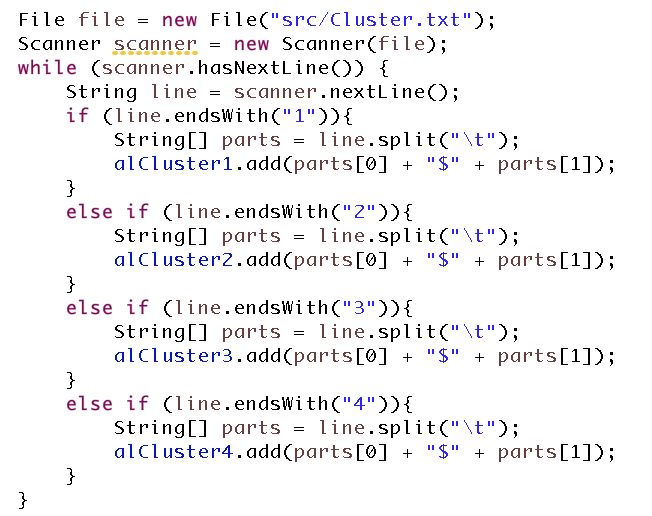


Figure 28: Filtering each entry according to cluster.

## Setting Stage

The stage is set by setting all the relevant parameters (Figure 29).

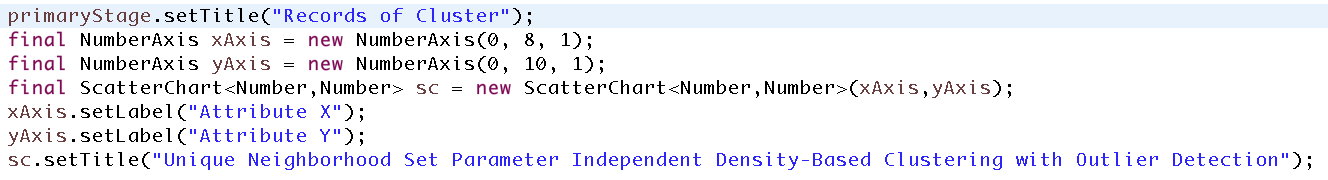


Figure 29: Stage set with all the required parameters.

## Cluster Plotting

After that, each cluster is plotted by;

* iteratively taking the string from the relevant array list,
* separating x and y value by using $ sign in middle
* adding the x and y value into that cluster and plotting it

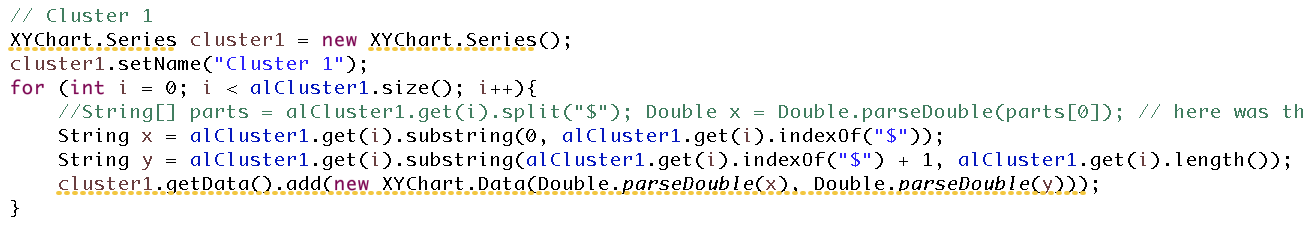


Figure 30: Cluster 1 plotting.

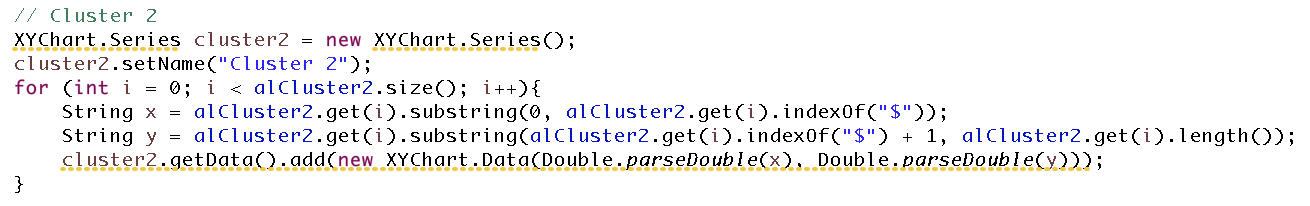


Figure 31: Cluster 2 plotting.

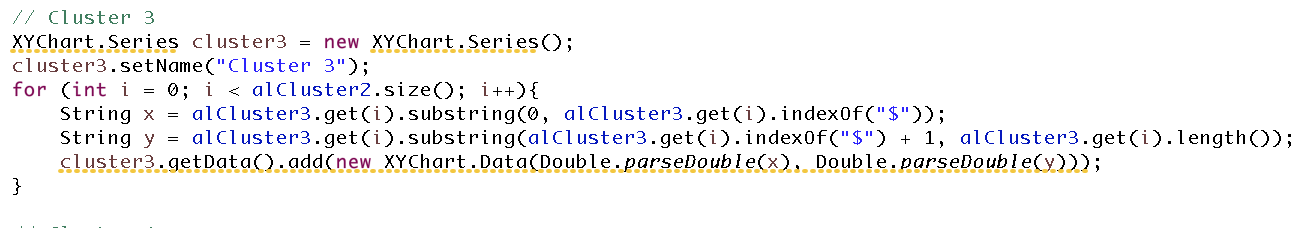


Figure 32: Custer 3 plotting

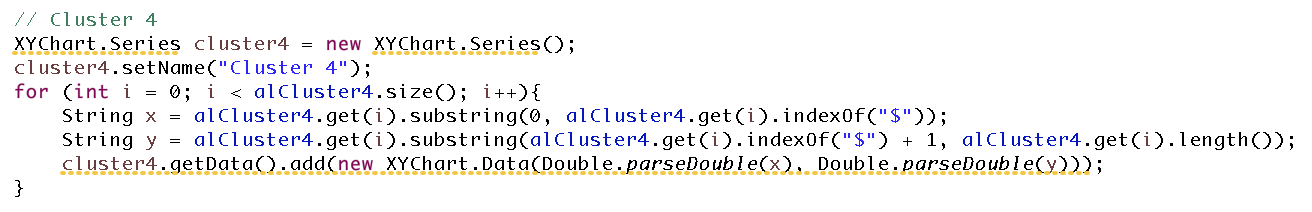


Figure 33: Cluster 4 plotting.

## Result

And finally, after plotting all four clusters, following is the graph shows their patterns (Figure 34).

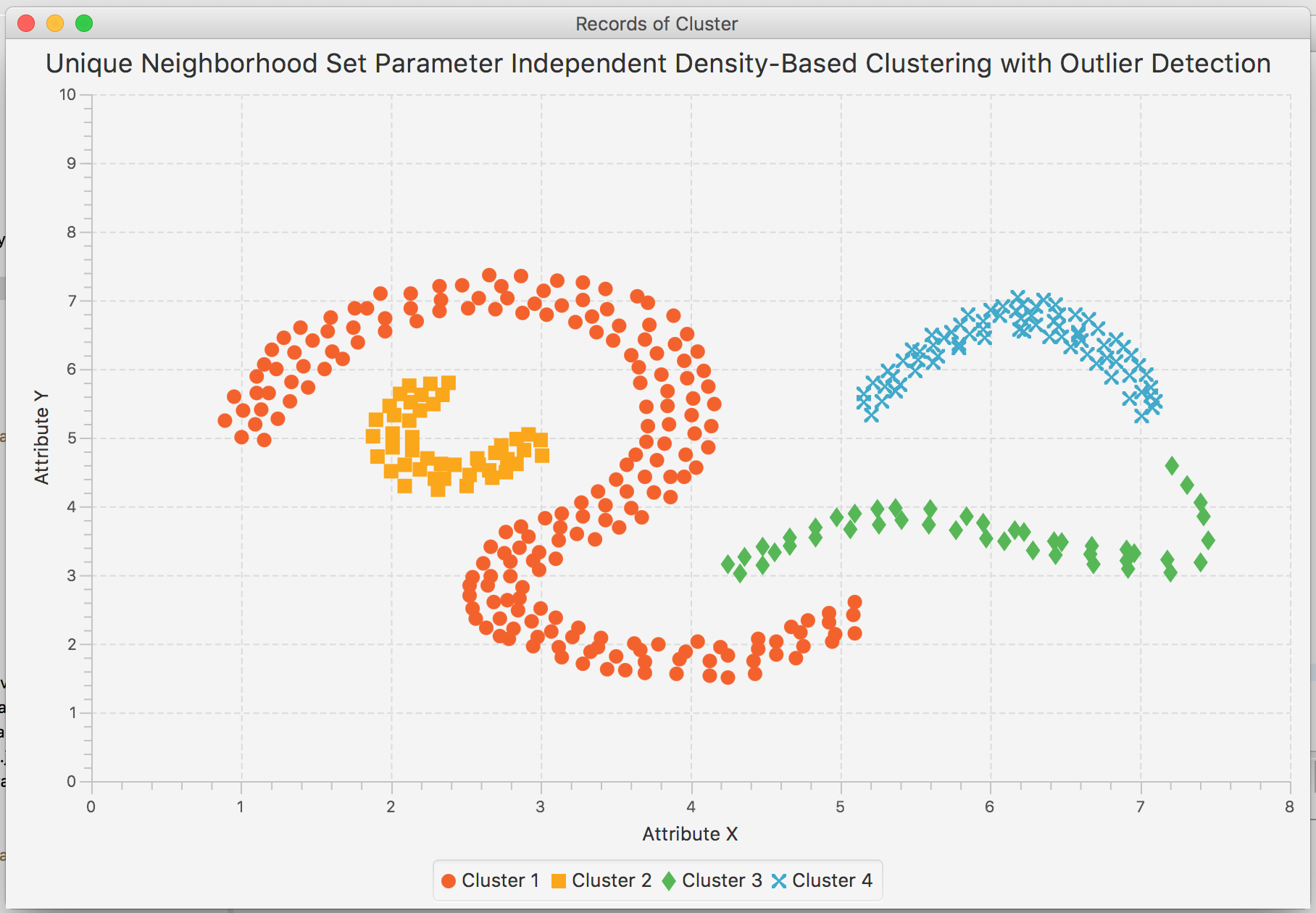


Figure 34: Records of four clusters.

Each cluster is shown by unique colour, as shown in Figure 35



Figure 35: Unique colour for each cluster.

## Complete Screen Shot

Figure 36 shows complete screen shot of the program and shows program runs smoothly with no error in console.

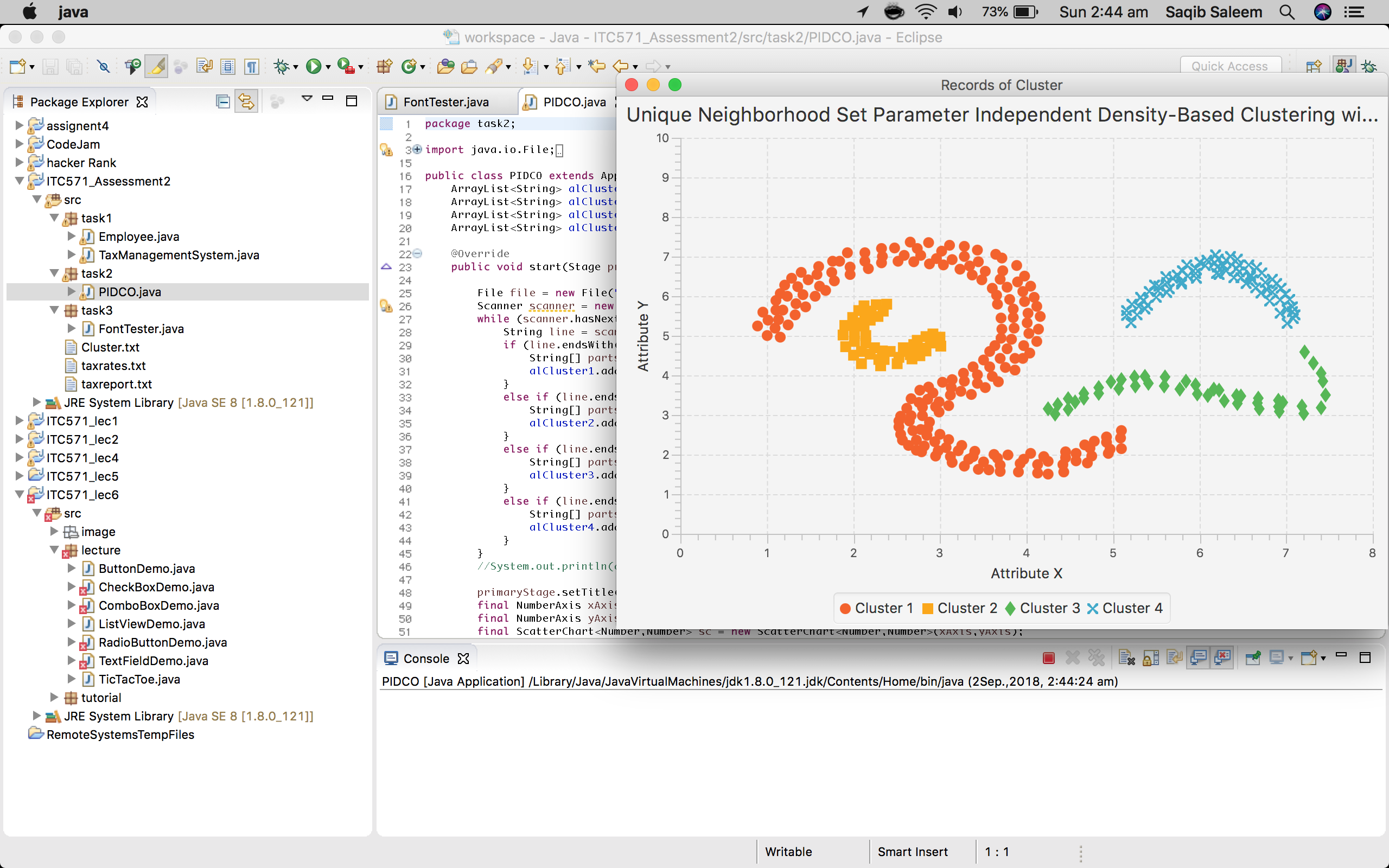


Figure 36: Complete screen shot shows no error in console