Assignment: JavaFX Animated Charts

Your job is to write a program to create animated bar and line charts. You will be implementing a JavaFX GUI and the supporting classes to read and parse real-life data and plot animated bar and line charts.

The main learning objectives of this assignment are:

- Sorting arrays and list: Define the natural order of a user-defined type and sort an array/list.
- Reading input files: Read text and XML files and parse them.
- Data visualization and graphics: Plot bar and line charts to produce an interesting visualization.

Design of Classes

In your design, you need to create classes that are given in this section. Besides, you can create some other classes and add new fields and method to given classes.

You must create an abstract Chart class, BarChart and LineChart classes that extends Chart class. You must create Bar and Line classes whose objects can added to BarChart and LineChart objects, respectively. Bar and Lines classes must implement the Comparable interface.

A *bar* collects related information (name, value, and category) for use in a bar chart.
A *line* collects related information (name, value, and category) for use in a line chart.
A *bar chart* is a data type that supports drawing static bar charts.
A *line chart* is a data type that supports drawing static line charts.
Chart is an abstract class that holds common properties of the bar and line charts.
AnimatedChart produces animated bar and line charts using BarChart and LineChar objects, respectively.

Getting Information from User

Your program should read both text and XML files in the correct format and then parse them to extract information.

Your program should read line-by-line the text file in which commas separate fields. In the text file, records are grouped by time periods.
Your program should also read and parse an XML file. In the XML file, each element enclosed by the two tags <record> and </record> contains the date, name, country, value, and category.

You can find file formats in ERUDM.

Documentation

At the end of the project, you must prepare a detailed document that describes your work. You must give details about your project, provide the class and use case diagrams. You can find a template for the project report on the ERUDM.

Academic Honesty

Honesty and integrity are essential. Never submit the work of others as your own. A student that copies the same code from an outside source or another one or allows someone to copy from his/her project will receive 0 points.

Submission

You should upload your project in zip format into ERUDM. The name of your submission must be your group number (e.g., group1.zip). You will not get any marks if you forget to name your submission as your group number. Each submission should consist of all java files required to compile and run your program plus a project report. We cannot grade something that does not run.

Note: No third-party toolkits or libraries are allowed. However, you are free to use any of the sample codes provided in this course.