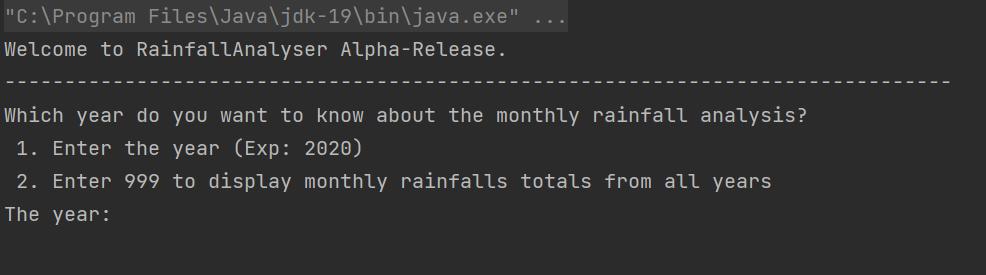
**Rainfall Analyzer (Alpha):**

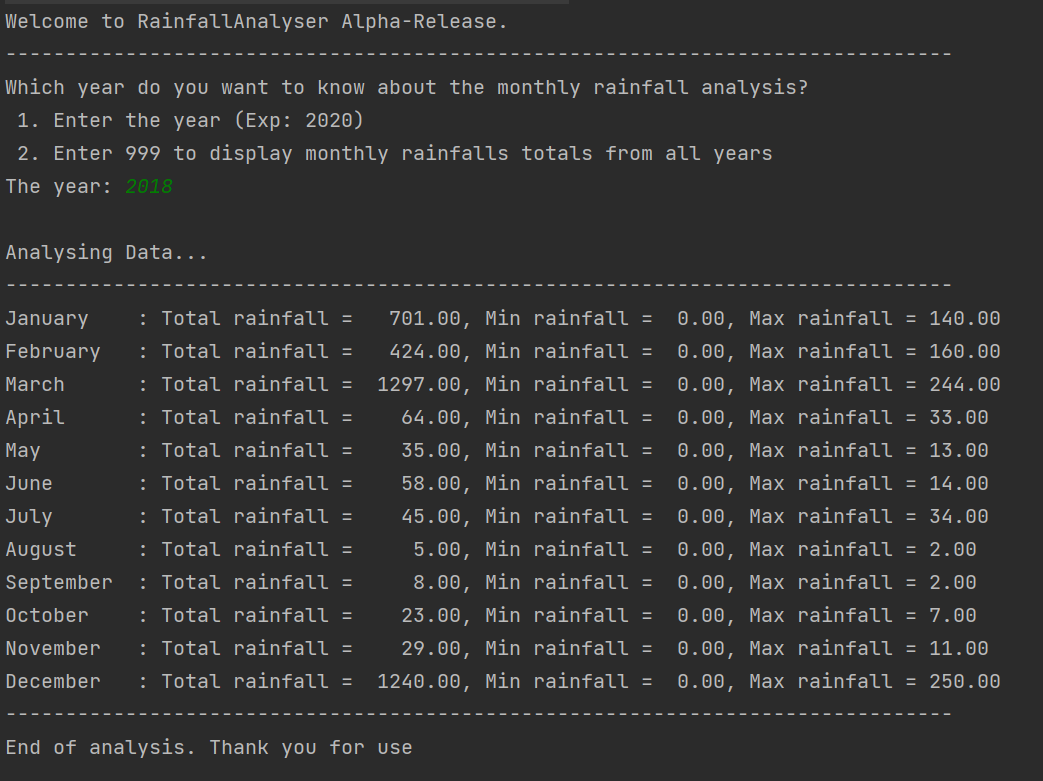
1. Description:

The file path to the rainfall CSV file is stored in the final String analysisFilename variable of the Alpha version analyzer. The analyzer will examine the saved file's rainfall data when the software runs. The file path may need to be manually typed if the user wants to analyze a different one. In this version, the CSV file is read with the help of the Java library common CSV. Users need to enter the year they wish to examine when the software launches, after which it will provide the monthly total, daily minimum, and daily maximum rainfall totals for that year. The user can input "999," which means to examine the rainfall data from all years, to analyze all of the rainfall data in the CSV file. The analyzed data will be presented to the user in the format of “Month: Total rainfall = 00.00, Min rainfall = 00.00, Max rainfall = 00.00” for all twelve months. The data.txt file will be used to hold the analyzed data. The final String variable, analysisResultFilename, contains the location where the analyzed data is saved ("src/main/resources/data.txt"). The program can also handle situations such as if the CSV file does not exist, the CSV file is empty, and other IO issues with the help of try and catch statement. There is only one class RainfallAnalyser for this program. In the class, different static methods are created instead of all stored under the main method. The main method only read the CSV file with the try-and-catch statement and calls the static methods.

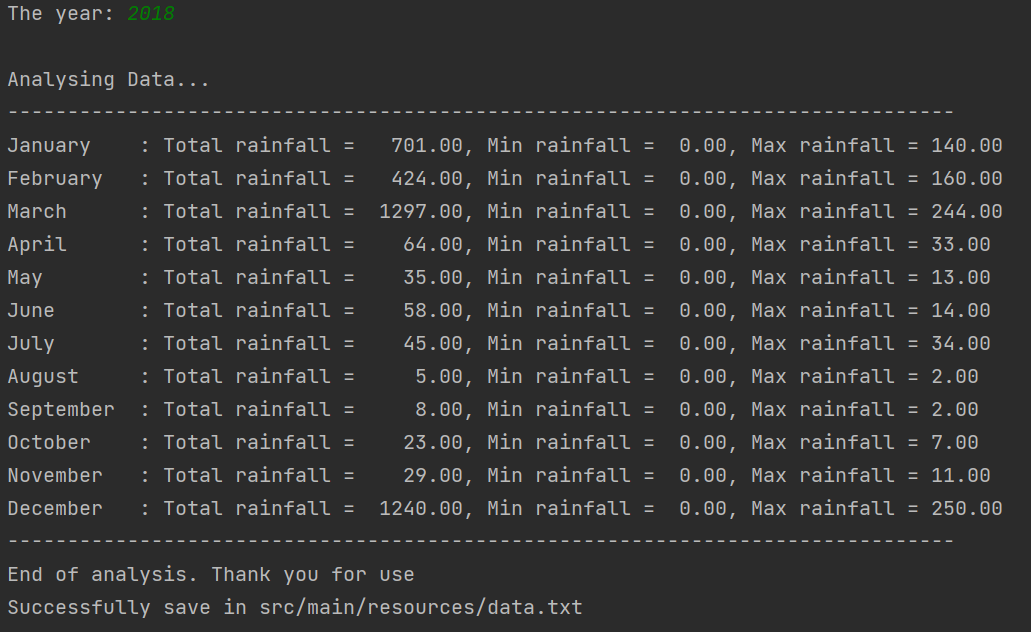
1. Showing welcome messages and get the year user selected to analyze.



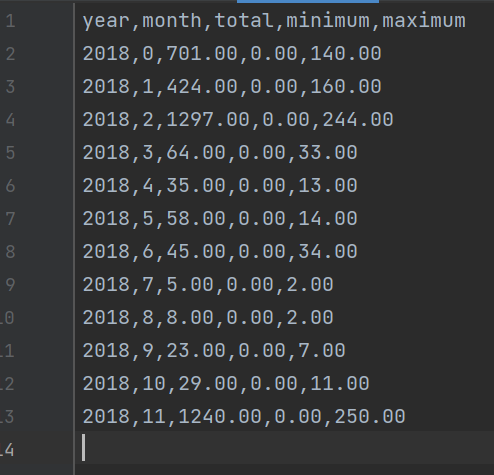
1. Display analyzing message and the analyzed rainfall information in formatted text.



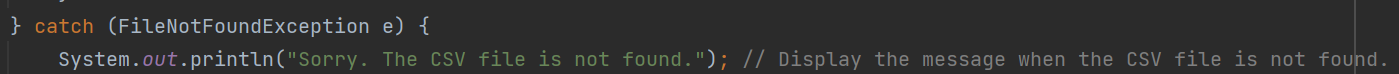
1. Display end of analyze message and where the processed data is stored.

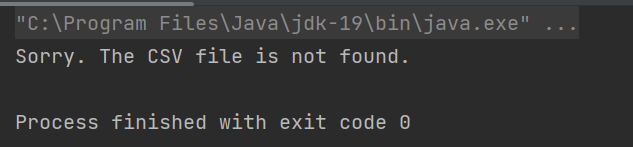


1. Processed data stored in the txt file will be like:

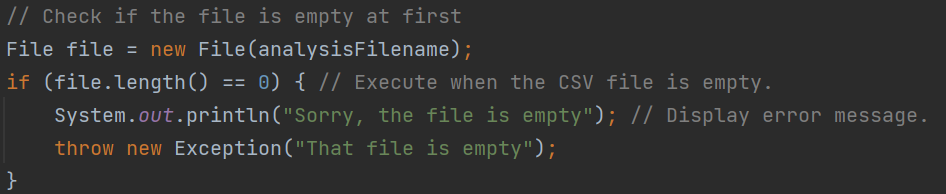


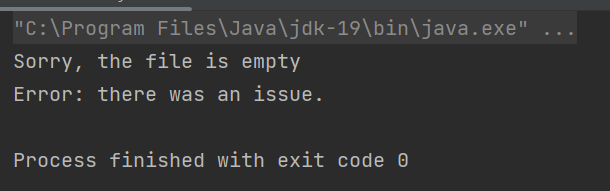
1. When the CSV does not exist.



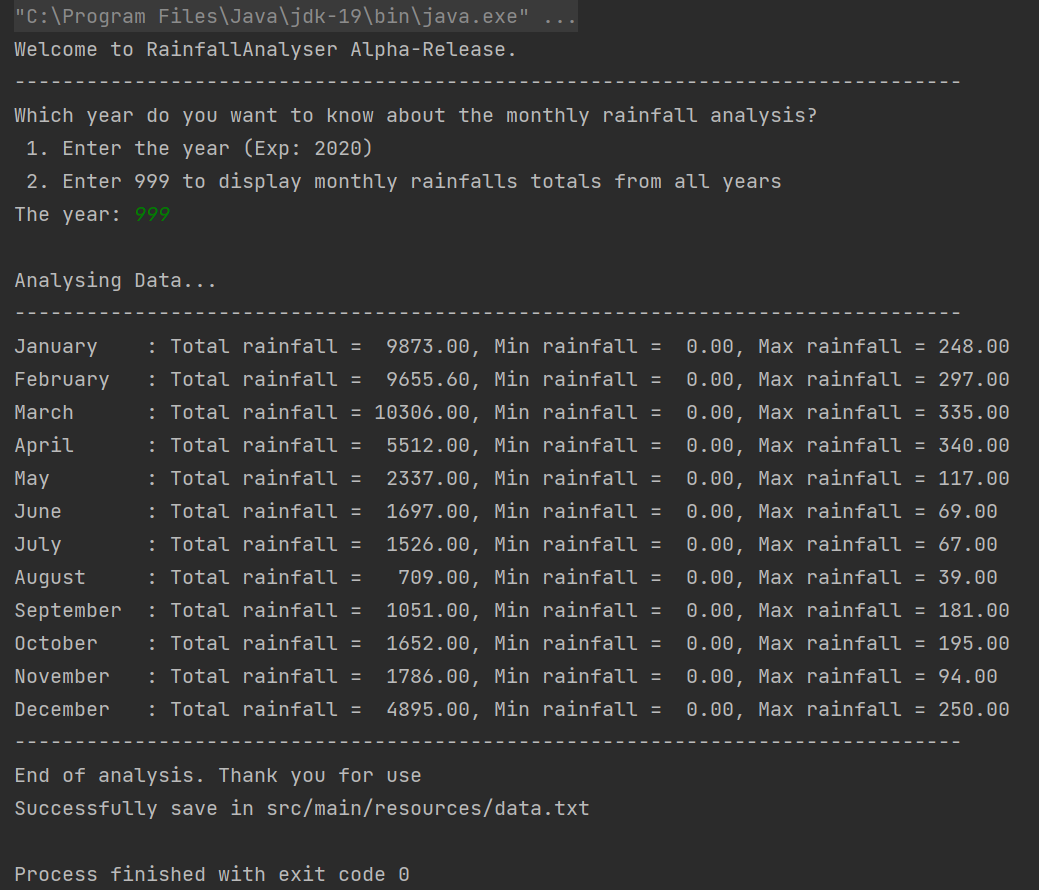


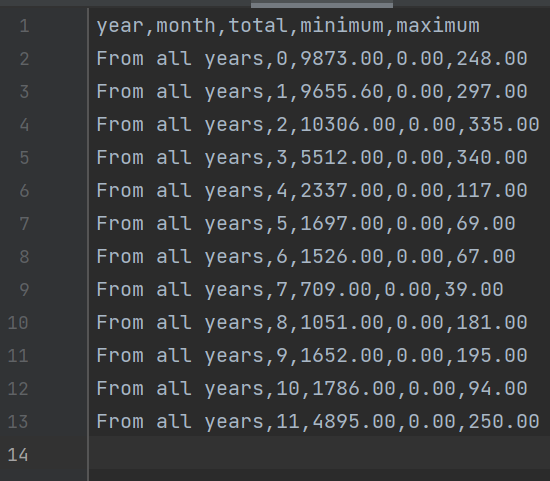
1. When the CSV file is empty.





1. If the user wants to analyze all years’ rainfall data.



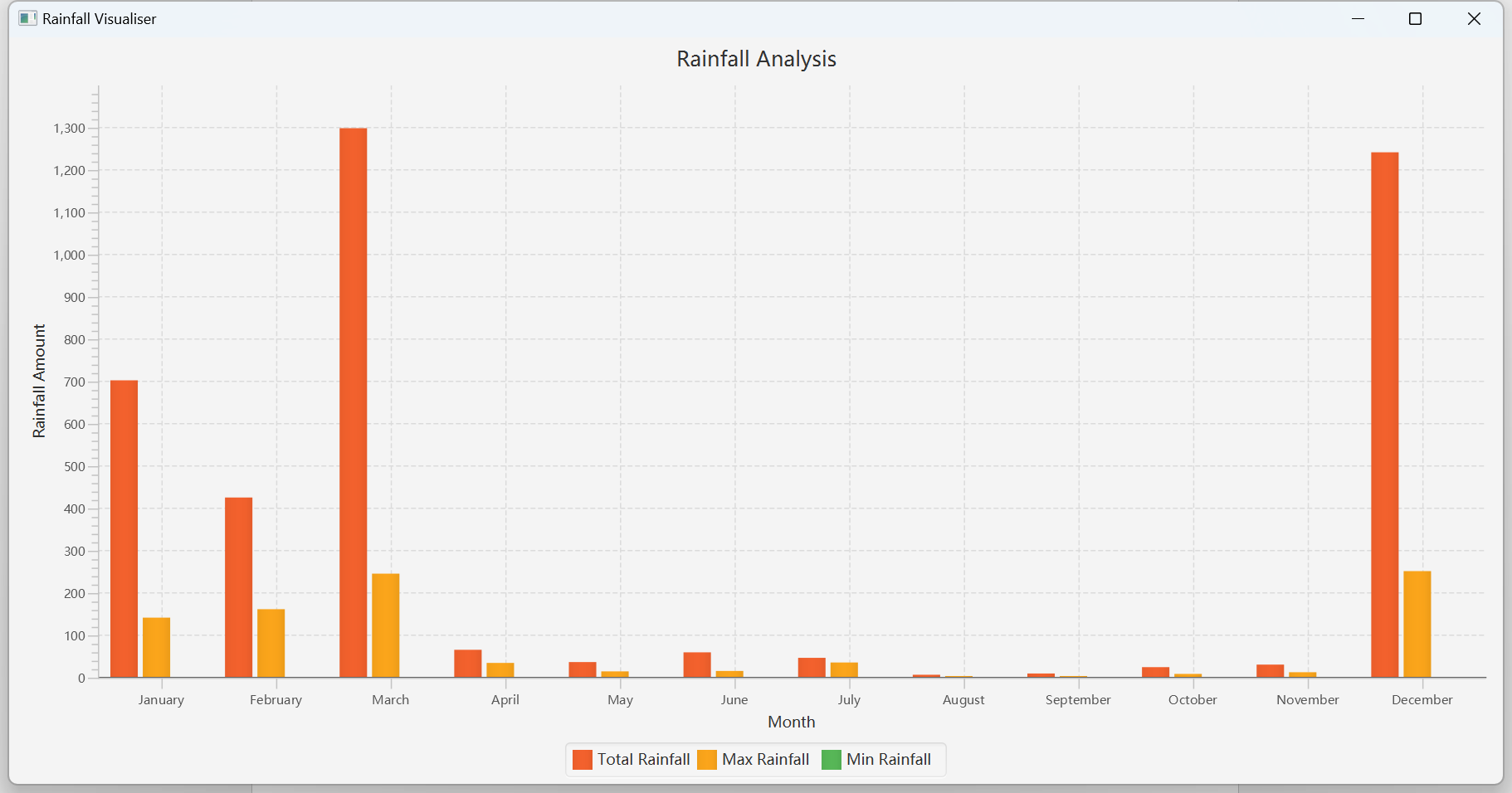


**Rainfall Visualizer (Alpha):**

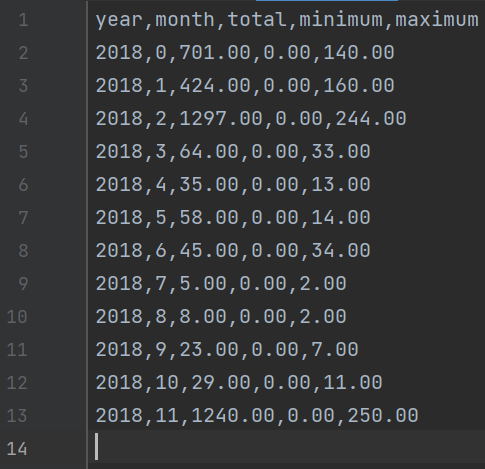
1. Description:

The analyzed rainfall data is required to create the graphical chart in the Alpha version rainfall visualizer (like the data.txt file generated from rainfall analyzer). A bar chart with the monthly totals, daily minimum, and daily maximum rainfall will be displayed when the software runs. The red bar depicts the total monthly precipitation, the yellow bar the maximum daily total for the month, and the green bar the minimum daily total. There is only one class RainfallVisualiser developed in this software. The program will first read the analyzed rainfall data, input them into the bar chart, and then display a JavaFX window showing the bar chart with rainfall information. SetupBarChart method is used to initialize the bar chart with title, x-axis, and y-axis labels. ProcessData method is used to process the data from the txt file to get all the analyzed information. InputDataToBarChart method is used to input the analyzed information into the data series. Then createXAxesOfBarChart will get the data series and add it to the bar chart.

1. The graphical chart:



1. The data file used to plot the bar chart:



**Rainfall Analyzer (Beta):**

1. Description:

RainfallProcessData, RainfallReader, and RainfallSaveFile classes are used in the Beta version of rainfall analyzer. The ClassTester class is used to test the functions of the three classes above to ensure the stability of the rainfall package.

**RainfallProcessData:**

This class is used to process the rainfall data stored in the CSV file. It provides methods to calculate the monthly total, daily minimum, and daily maximum rainfall. The processed data will be stored inside the variables monthlyTotal, dailyMin, and dailyMax in the class. Methods getMonthlyTotal, getDailyMin, and getDailyMax will be used to pass the processed data to the caller.

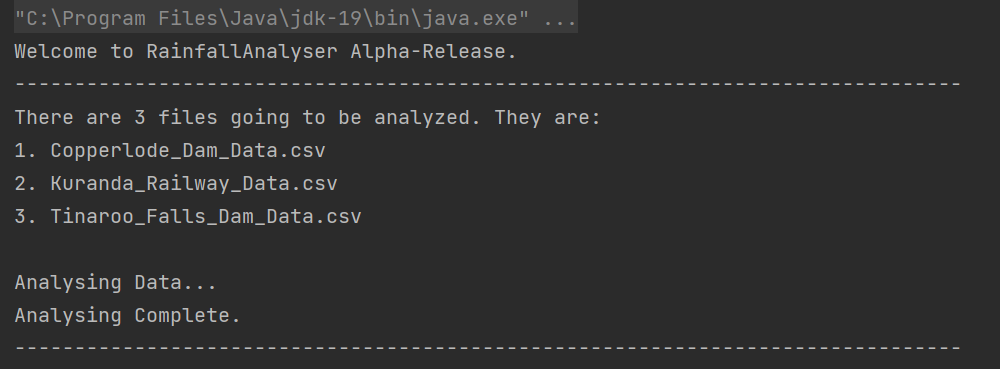
**RainfallSaveFile:**

This class requires a filename, monthName, monthlyTotal, dailyMin, and dailyMax when creating an object. This class is used to save the processed rainfall information in a txt file. DisplaySavedMessage method in the class is used to display whether the information is saved successfully, and where did the information save. The class provides a standard of IO exceptions, such that if there is already a file created in the destination path, an error message will be displayed.

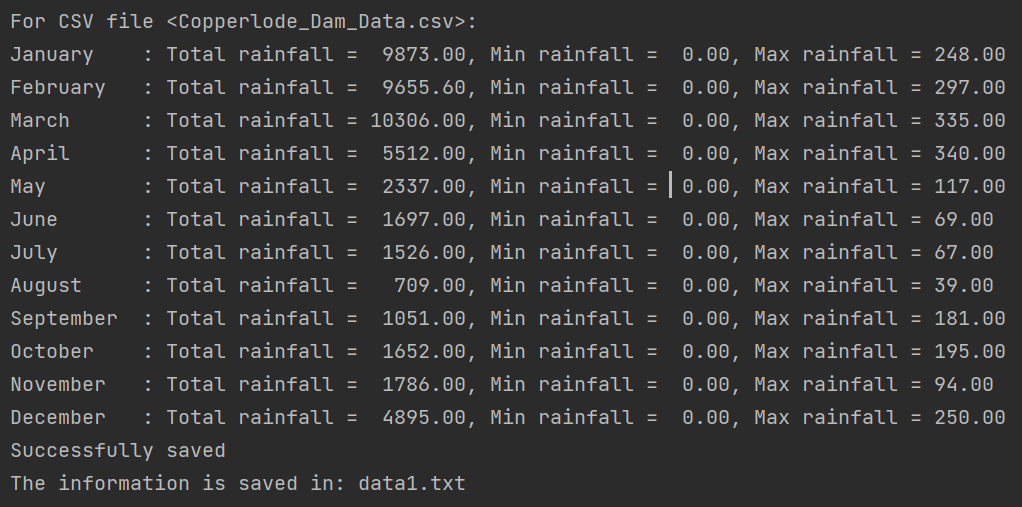
**RainfallReader:**

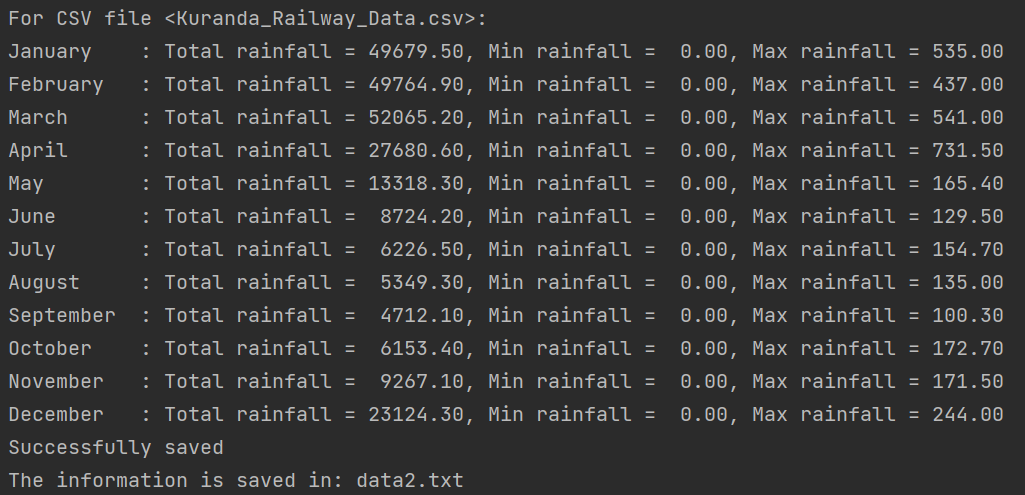
This class is the main part of the program. Under the main method of the class, a for loop is used to process all the CSV files listed in the String list filePathList. CSVReadData method is used to read rainfall data from the CSV file with the help of RainfallProcessData class. A try-and-catch statement is used under the method to catch IO issues such that a file is not found, and format error. LoadProcessedData method is used to load all the processed data from the RainfallProcessData object. CSVPrintData is used to display the analyzed rainfall data to the user. DisplayWelcomeMessages and DisplayEndingMessages methods are used at the start and the end of the program to display messages.

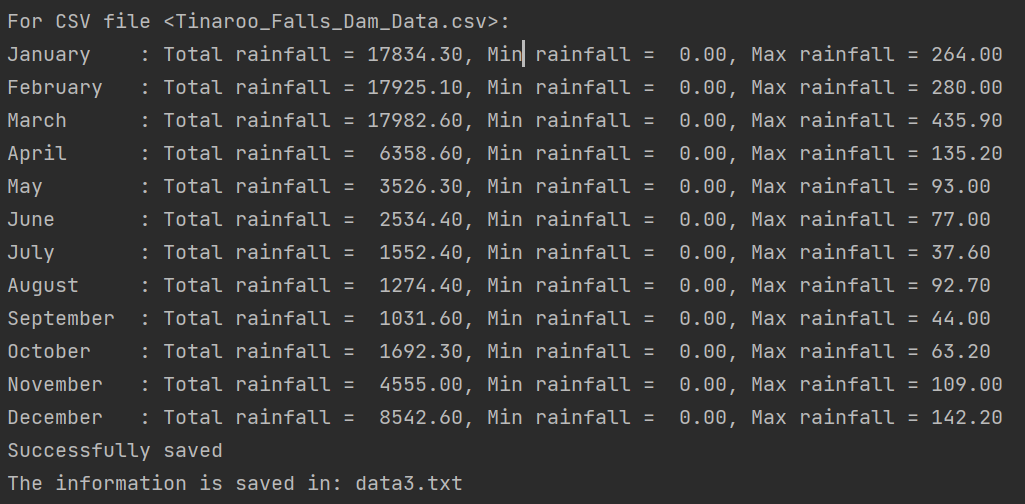
1. Load the name of the CSV file going to analyze:



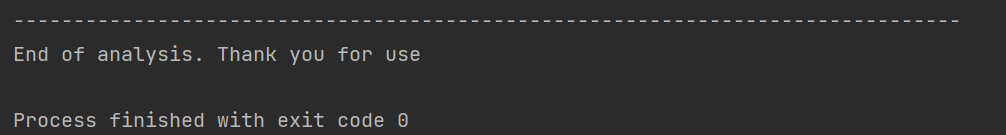
1. Display the analyzed information, if saved successfully, and where does it saved at:



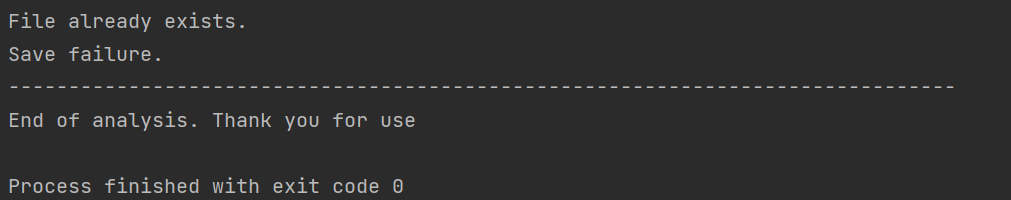




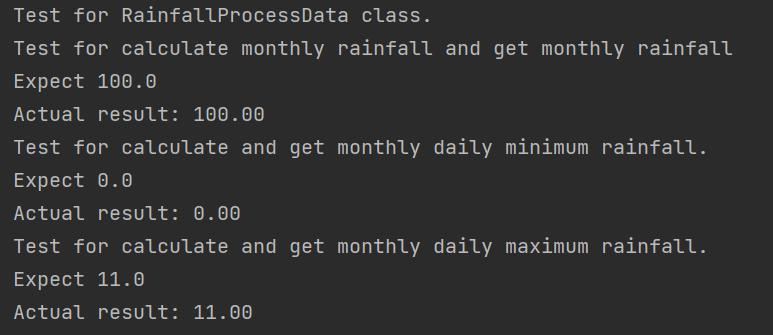
1. Display ending message:



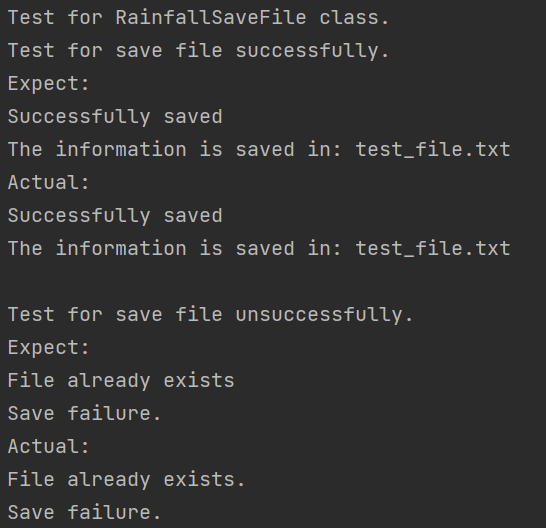
1. If the path where the file is going to save is already taken / saved file exists:



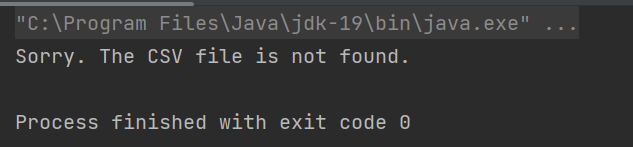
1. Test RainfallProcessData class (that used to process rainfall data in the CSV file):



1. Test RainfallSaveFile class:



1. When the CSV does not exist.



**Rainfall Visualizer (Beta):**

1. Description:

The beta version rainfall analyzer package, rainfall, is used in the beta version rainfall visualizer. In this visualizer, there are a total of three windows can be displayed to the user. They are the home page, selection page, and visualization page.

**Home page:**

On the home page, there is a welcome message in the center, followed by a status message that shows "Please load a rainfall CSV to be analyzed" at the start. There are also three buttons placed at the bottom of the window. The "Load Rainfall Data From Computer" button is used for the user to load the CSV file from their device. The "Load Rainfall Data From List" button is used to display a list of CSV files for the user to select to analyze. Then the "Start Visualizer" button is used to generate the graphical chart of analyzed rainfall information. When a load is loaded either from the user's device or the list, the status bar will change and display a message telling which CSV file has the user loaded.

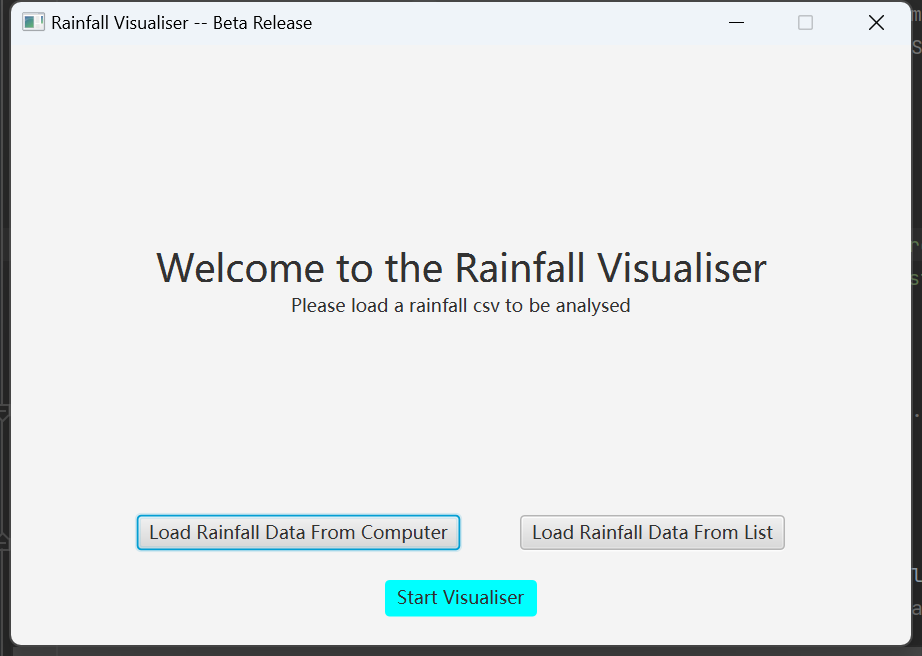
**Selection page:**

The selection page will be displayed after pressing the "Load Rainfall Data From List" button. Users can select one of the CSV files in the list and then click the "Start Analyze" button to load the CSV file and jump back to the home page.

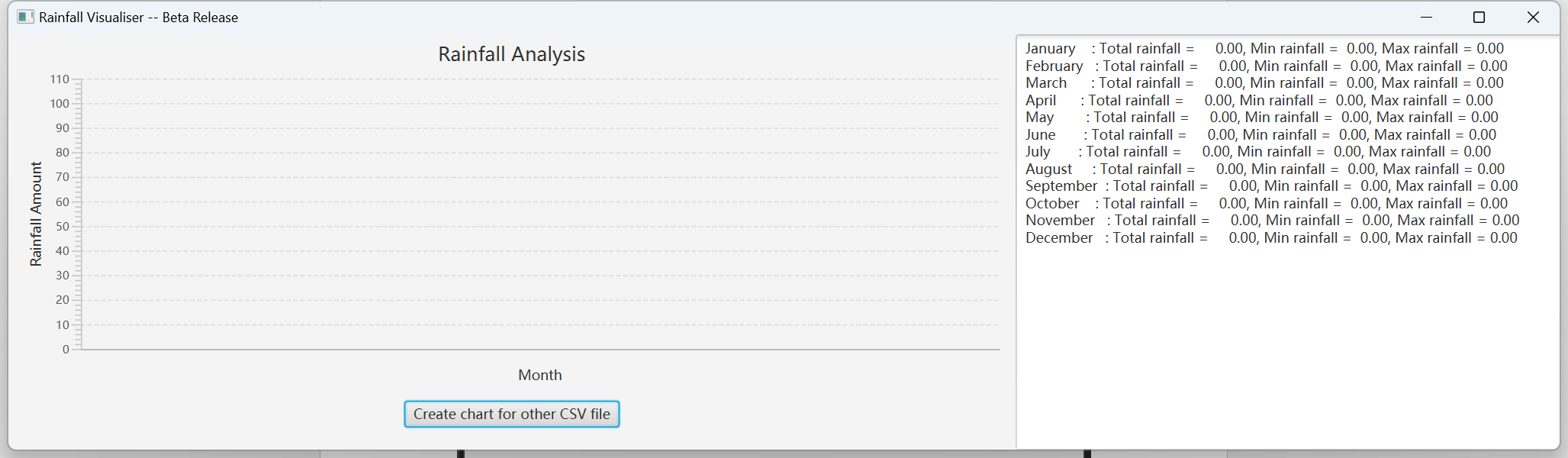
**Visualization page:**

The visualization page will be displayed after pressing the "Start Visualiser" button. The graphical chart is presented at the left of the window, and a text format of analyzed rainfall information is presented at the right of the window. If no CSV file is not loaded, the visualization page will be still shown to the user, but the bar chart is empty, and the text information displayed is 0.00 for all 12 months of rainfall data. Users can press the "Create a chart for other CSV file" button to go back to the home page to load another and analyze it.

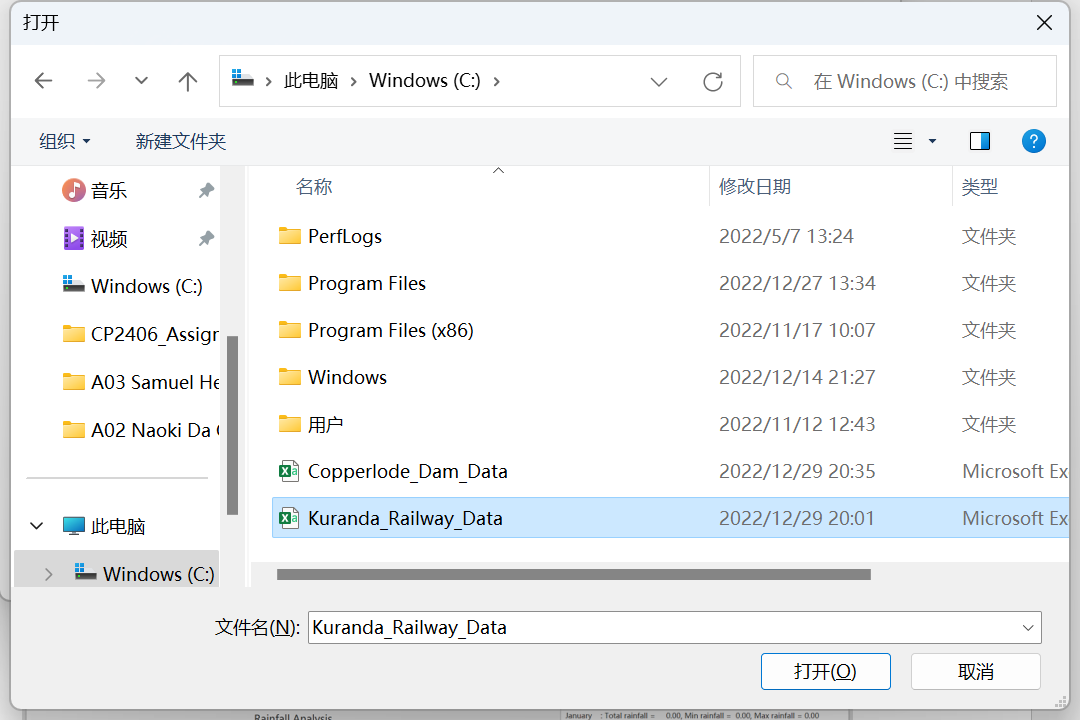
1. Home page of rainfall visualizer:



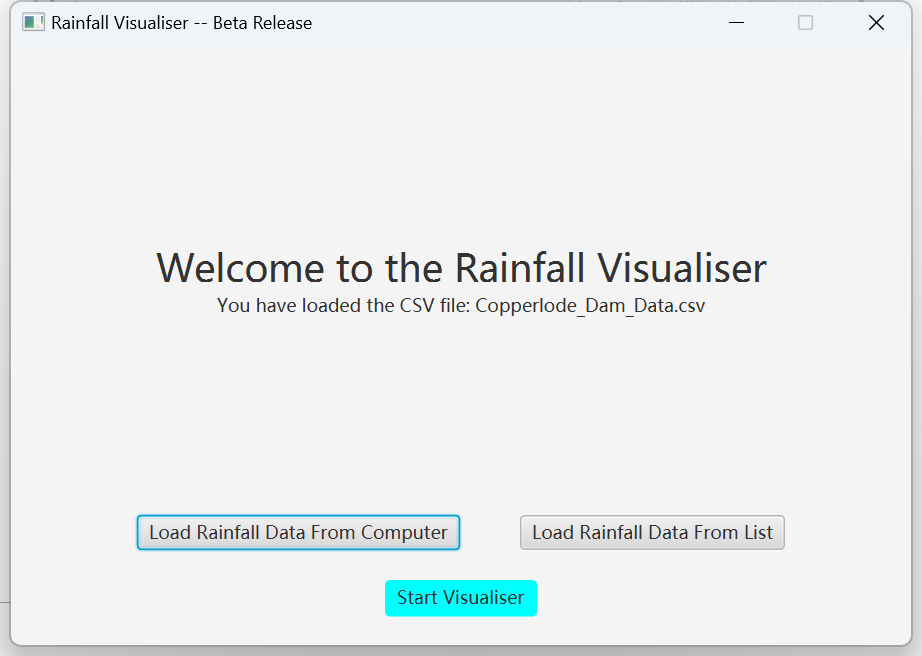
1. Directly start visualizer without load any file:



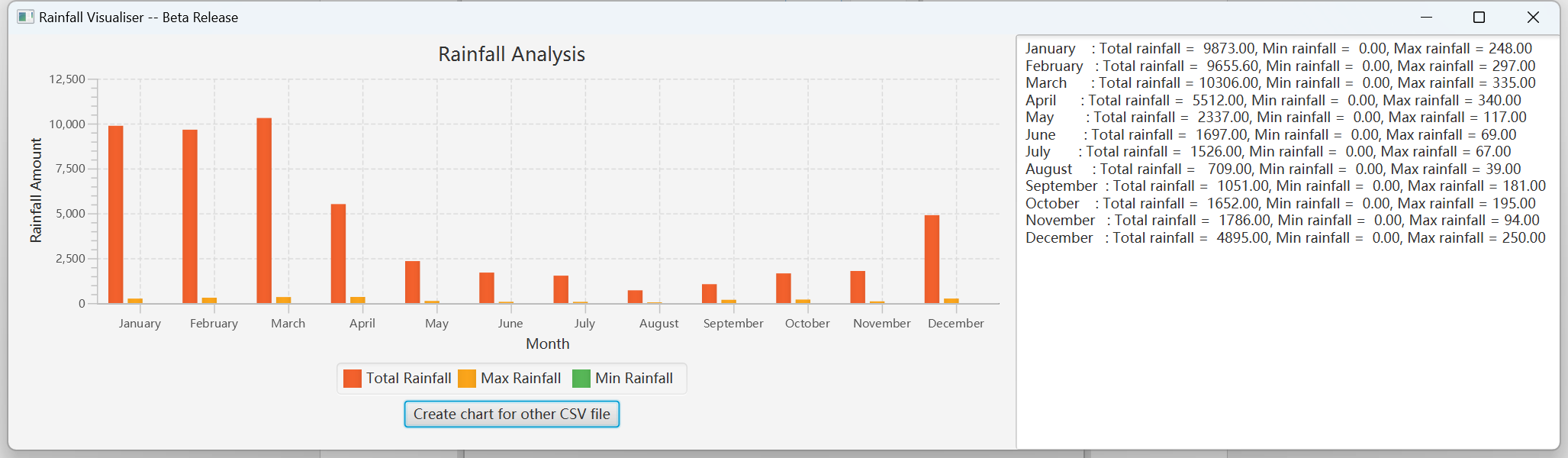
1. After press “Load Rainfall Data from computer” Button:



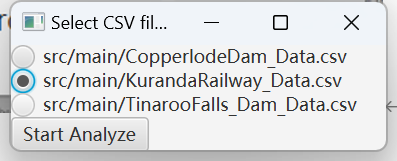
1. After load the “Copperload\_Dam\_Data” CSV file:



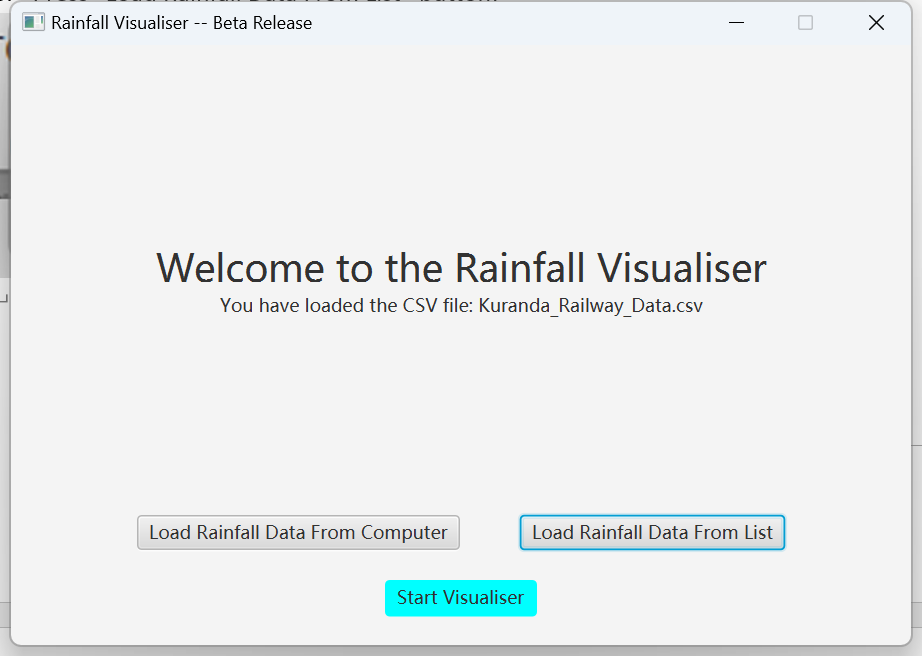
1. Start Visualiser after load file:



1. Press “Load Rainfall Data From List” button:



1. After press “Start Analyze” button:



1. Then start visuliser:

