Dokumentacja aplikacji CryptoCurrency.com

Michał Binda, Wojciech Michaluk, Mikołaj Mróz

23 stycznia 2023

Spis treści

1 Klasy

1.1 My GUI

This class creates a GUI that displays a line chart of cryptocurrency prices. It reads data from a JSON file and uses it to create a line chart using the JFreeChart library. The GUI consists of two panels, one for the left containing buttons for different cryptocurrencies and date pickers, and one for the right showing the graph. It also has a "Show"button which updates the graph with the selected cryptocurrency and date range when clicked. It uses the FlatLightLaf look and feel theme to style the interface.

1.2 LineChart

Java class called "LineChart"creates a line chart using the JFreeChart library. It creates a ChartPanel object which contains the JFreeChart object, which is used to display the chart. The class has a constructor that takes four parameters: startDate, endDate, name, dates and prices. The startDate and endDate are used to filter the dates array passed to only include dates within the startDate and endDate range. The name parameter is used as the title and the series name of the chart. The dates and prices parameters are used to populate the data in the chart. The class sets the background color of the chart to black, sets the font color of the axis labels and tick labels to white and removes the chart legend. The class also contains a helper method called setAxisFontColor which is used to set the font color of the axis labels and tick labels.

1.3 FileReader

Java class called "FileReaderJ"reads and parses a JSON file. It contains two private variables, graphDates and graphPrices, which are both HashMap objects that store the date and the prices in respective ArrayList. The class has a constructor that initializes these two variables. The class has a method called readFile(String fileName) that reads the contents of the fileName passed as parameter, parse it and extract the data, assetName, date and priceUsd. It formats the date to "yyyy-MM-dd"format and truncates the price to 4 decimal places. It then maps the assetName to the respective dates and prices and stores it in the graphDates and graphPrices HashMap respectively. The class also has two methods, getGraphDates() and getGraphPrices(), that return the HashMap objects graphDates and graphPrices respectively.

1.4 DateButton

This is a Java class called "DateButton"which extends the JButton class and implements the ActionListener interface. It contains a JDatePicker object to display the current date and a Calendar object to store the selected date by the user. When the button is clicked, the selected date is retrieved from the JDatePicker and stored in the Calendar object, and the class has a method called getDate() which returns the date in the format of "yyyy-mm-dd".

1.5 StringButton

This class is a custom JButton class called "StringButton"which is used to display a button with a string label. It has a constructor that takes in a string parameter, which is used as the label for the button. The constructor sets the label of the button, and adds an action listener to it. When the button is clicked, it calls the onClick() method, which prints the label of the button to the console and adds it to the cryptoList ArrayList. It also has a getString() method that returns the label of the button.

2 Podsumowanie

"Cryptocurrency.com"is a graphical user interface (GUI) application that allows the user to view historical prices of different cryptocurrencies in the form of line charts. The application reads data from a JSON file and uses it to populate the charts. The user can select different cryptocurrencies and date ranges using buttons and date pickers on the left side of the interface. The application uses the JFreeChart library to create the charts and the FlatLightLaf look and feel theme to style the interface. The "StringButton"and "DateButton"classes are custom classes that extend the functionality of JButton class, while "FileReaderJ"reads the json file and "LineChart"creates the line chart based on the data read. Overall, the application seems to be functional and allows the user to view historical prices of different cryptocurrencies in a user-friendly way.