

# Real-Time Stock Prediction and Visualization Project

## 1. Introduction

The **Real-Time Stock Prediction and Visualization** project aims to fetch real-time stock data, process it, predict future stock prices, and visualize the data using enhanced charts. The project leverages the **Alpha Vantage API** to obtain stock data and uses **XChart** for visualization.

## 2. Features

- **Real-Time Data Fetching:** Retrieves live stock data for specified stock symbols.
- **Data Processing:** Cleans and processes the fetched stock data.
- **Price Prediction:** Implements a simple prediction algorithm to estimate future prices.
- **Visualization:** Uses line charts with moving averages for better insights.

## 3. Setup

### 3.1 Prerequisites

Ensure you have the following installed:

- **Java Development Kit (JDK 11 or higher)**
- **Maven**
- **Internet Connection** (for fetching stock data)

### 3.2 Installation

1. **Clone the repository:**
2. `git clone https://github.com/yourusername/stock-prediction.git`
3. `cd stock-prediction`
4. **Configure the API key:**
  - Obtain a free API key from **Alpha Vantage**.
  - Create a `config.properties` file in the `src/main/resources` directory with the following content:
    - `api.url=https://www.alphavantage.co/query`
    - `api.key=YOUR_ALPHA_VANTAGE_API_KEY`
5. **Build the project:**

```
mvn clean install
```

## 4. Usage

### 4.1 Running the Application

To run the application, execute the following command:

```
mvn exec:java
```

The application will fetch real-time stock data for **Apple Inc. (AAPL)**, process the data, predict future prices, and display the results in a chart.

## 4.2 Customizing the Stock Symbol

To fetch data for a different stock, update the following line in Main.java:

```
String jsonData = apiClient.fetchStockData("AAPL");
```

Replace "AAPL" with another stock symbol (e.g., "MSFT" for Microsoft).

## 5. Explanation of the Chart

The chart generated by your application provides insights into the predicted stock price movements for the specified stock symbol.

### 5.1 Understanding the Chart

- **X-Axis (Time):** Represents time intervals (e.g., minutes or hours).
- **Y-Axis (Price):** Represents the stock price at each time interval.
- **Data Points:** Show the predicted stock prices at various time intervals.
- **Line Chart:** The trend of the predicted stock prices over time.
- **Moving Average:** An additional line smoothing out fluctuations and highlighting trends.

### 5.2 Example Interpretation

For example, if your chart shows the following data points:

Time Interval Price (USD)	
0	150
1	151.5
2	153.0
3	154.5
4	156.0
5	157.5

It suggests that the stock price is predicted to **increase** over these intervals. The **moving average** line helps smooth out fluctuations and provide a clearer trend.

## 6. Conclusion

This project successfully integrates real-time stock data retrieval, processing, and visualization. By leveraging **Alpha Vantage API** and **XChart**, users can analyze stock trends effectively. The modular design allows easy customization for different stock symbols and prediction models.