THA-027

### Overview

This assignment is designed to simulate a real-world scenario where you will apply your knowledge of Python, PostgreSQL, financial data analysis, and visualization to generate trading signals based on specific conditions. You will also practice version control using Git to manage your project's development lifecycle.

**Read the complete text before starting the assignment**. Please feel free to connect regarding the assignment or related guidance. I will be happy to guide you through it. Please do not expect direct answers.

### Objectives

* Set up a PostgreSQL database and load it with provided stock equities data.
* Develop a Python program to analyze stock data and generate trading signals.
* Calculate overall profit and loss for each stock based on generated signals.
* Visualize trading data and signal-related transactions.
* Use Git for version control, ensuring proper commit messages and project organization.

### Signal Logic

* Generate a \*\*buy signal\*\* if there's a crossover of 50-day and 500-day **moving average.**
* Generate a \*\*sell signal\*\* if there's a crossover of 20-day and 200-day moving averages.
* Close any existing \*\*buy positions\*\* if a crossover of 10-day and 20-day moving averages takes place
* Close any existing \*\*sell positions\*\* if a crossover of 5-day and 10-day moving averages takes place.

\*Please use Simple Moving Average (SMA)

### Instructions

Given a set of stock equities

1. Set up a Postgres database, and load the provided data into this database.
2. [optional, extra points] Clean data if needed.
3. Write a python program to pick up data for a specific stock, and generate a signal for the following conditions.
4. Generate a signal. You are free to build an ML model if needed, or just generate these signals directly using Python.
5. Calculate overall profit and loss for each stock data individually, while making these trades. Store the results into the database.
6. [optional, extra points] Visualise the results of your trades. You are free to use any plotting tool or library. The bare minimum expectation is the plot is able to show
7. Overall data for a stock as candlesticks.
8. Where the entry and exit took place.
9. Buy/Sell positions.

Use `matplotlib`, `plotly`, or `mplfinance`, or `tradingview charts` to create visualizations, or any other library of your choice.

### Submissions Criteria

* Name your project
* Submit the URL to your Git repository.
* Your repository should have at least 2-3 commits. Feel free to make more as needed.
* Please do not use Jupyter notebooks, use normal .py files. Jupyter notebooks are okay for any experimentation. And, you are free to commit those.
* Write clear, descriptive commit messages.
* Ensure your repository is public or provide access if it's private.
* Your repository should include all scripts, a `README.md` for setup and execution instructions, and any Jupyter notebooks used for experimentation.