

Data Import & Basic Analysis

Task 1: Import the CSV files into Python or PostgreSQL.

CSV Files: nifty_tick_data.csv and option_chain_tick_data.csv.

Instructions:

Use Python (Pandas) or PostgreSQL to import the data.

Provide basic descriptive statistics (mean, median, min, max) for the following key columns:

Nifty Tick Data: Open, Close, Volume

Option Chain Data: Strike Price, Volume, Open Interest

Task 2: Calculate the total volume for each option type (Call/Put) based on the provided data.

Instructions:

Group the data by Option Type and calculate the total volume for each type (Call and Put).

Data Cleaning & Formatting

Task 3: Identify and remove any rows with missing or inconsistent data.

Instructions:

Remove rows with missing timestamps or key values (such as missing strike prices, volumes).

Define and remove out-of-range values, if applicable (e.g., unusually high or low prices).

Task 4: Format the date columns into a standard format (YYYY-MM-DD) and correct any anomalies.

Instructions:

Ensure the Timestamp column is correctly parsed into a date-time format.

Convert the Timestamp to YYYY-MM-DD format and ensure the data is chronologically sorted.

Data Manipulation

Task 5: Write an SQL query or a Python script to filter the data for options with:

Criteria: Strike Price > 15000 and Volume > 1000.

Instructions:

Provide the SQL query or Python script that filters the option_chain_tick_data.csv for options meeting the above criteria.

Return the first 5 rows of the filtered data.

Visualization

Task 6: Create a time-series plot showing the trend of the Last Traded Price (LTP) for a specific option (e.g., NIFTY 15000 CE).

Instructions:

Use the option_chain_tick_data.csv to create a time-series plot for the LTP of a specific option (e.g., NIFTY 15000 CE).

Plot the LTP over time based on the available timestamps.

Task 7: Visualize the open interest for Calls and Puts over time for the last expiry.

Instructions:

Plot the open interest (OI) for both Call and Put options over time.

Use the last expiry date data available in the option_chain_tick_data.csv.

Strategy Development

Task 8: Develop a basic backtesting rule for buying a Call option.

Rule: Buy a Call option when the Last Traded Price (LTP) increases by 5% within a 10-minute window.

Instructions:

Create a Python script or function that scans the option_chain_tick_data.csv for opportunities to buy Call options when the LTP increases by 5% within 10 minutes.

Make an assumption that trades are executed at the bid price, and sales happen when the LTP decreases by 3% or at the end of the day.

Task 9: Calculate and plot the performance of this strategy over a specific time period.

Instructions:

Calculate the performance (profit/loss) of the backtesting strategy over a given week or time period based on the data.

Plot the cumulative returns or performance trend over time.