

Real-Time Tick Data Aggregation

Objective:

Write a Python script that subscribes to a Redis pub/sub channel, receives real-time tick data, and aggregates the data minute-by-minute (e.g., for OHLCV or summary statistics).

Requirements:

1. *Redis Subscription*

- a. Use the redis Python library to subscribe to a channel (e.g., BTCUSDT).
- b. Receive tick data in real-time (simulate if needed).

2. *Tick Processing & Aggregation*

- a. For each incoming tick (JSON), extract fields like price, volume, and timestamp.
- b. Group and aggregate ticks by minute (use the timestamp).
- c. For each minute, calculate and print/store summary statistics (e.g., open, high, low, close, total volume)

3. *Publishing Aggregated Data*

- a. After aggregating each minute's data, publish the summary (as a JSON object) to a Redis key or channel named after the symbol and minute (e.g., BTCUSDT:20250727_1810).

4. *Logging*

- a. Log processing steps, errors, and summary outputs.

5. *Documentation*

- a. Comment your code and provide a short README describing how to run the script and any dependencies.

Bonus:

- Save minute-wise summaries to a file or database.
- Make the script configurable for different symbols via command-line arguments.
- Handle reconnections and errors gracefully.

Deliverables:

- Python script (.py)
- README file with setup and usage instructions
- requirements.txt listing dependencies
- Aggregated minute-wise data published to Redis keys