

Tugas Scraping Stockbit

Project ini bertujuan untuk melakukan pengambilan data orderbook saham dari API Stockbit, menyimpan data tersebut ke dalam database MySQL, serta menampilkan hasilnya melalui dashboard sederhana.

Daftar Isi

1. [Metode yang Digunakan](#)
 2. [Cara Menjalankan Program](#)
 3. [Dashboard Sederhana](#)
 4. [Penjelasan File Utama](#)
 5. [Output Program](#)
 6. [Kesimpulan](#)
-

Metode yang Digunakan

Metode yang digunakan adalah HTTP API Scraping (REST API), bukan web scraping berbasis HTML.

Alur Kerja Program

1. Program mengirim request ke API Stockbit menggunakan Bearer Token
2. API mengembalikan data dalam format JSON
3. Data orderbook (BID & ASK) disimpan ke database MySQL
4. Data dapat ditampilkan kembali melalui dashboard

Alasan Pemilihan Metode

1. Lebih stabil dibanding scraping HTML
2. Data bersifat real-time / near real-time
3. Struktur data lebih terstruktur (JSON)

Cara Menjalankan Program

1. Mendapatkan Bearer Token
 - Tutorial pencarian Bearer Token tersedia di folder Lain-lain
 - Token diambil melalui Network tab (DevTools browser)
2. Konfigurasi Token
 - Paste Bearer Token ke dalam file config.py
3. Menjalankan Program Scraping
 - jalankan "main.py"

```
cd TugasScraping/ProjectScraping
python main.py
```

Dashboard Sederhana

Dashboard dibuat untuk menampilkan hasil scraping yang sudah tersimpan di database.

Cara Menjalankan Dashboard

1. Masuk ke folder project:

```
cd TugasScraping/ProjectScraping
```

2. Jalankan dashboard:

```
streamlit run dashboard.py
```

Penjelasan File Utama

1. main.py

File utama sebagai entry point program. File ini hanya mengatur alur eksekusi, bukan logika API.

Fungsi utama:

- Membaca daftar saham dari saham.txt
- Melakukan looping ke seluruh emiten
- Memanggil fungsi scraping di fetcher.py
- Menampilkan progress, elapsed time, dan ETA

2. fetcher.py

File inti yang menangani pengambilan dan validasi data API.

Fungsi utama:

- Mengirim HTTP request ke API Stockbit
- Menangani error HTTP dan token expired
- Parsing JSON secara aman
- Membedakan kondisi:
 - Data valid
 - Orderbook kosong
 - Struktur data berbeda
 - Menyimpan data BID & ASK ke database

3. db.py

File khusus untuk interaksi database MySQL.

Fungsi utama:

- Membuka koneksi database
- Menyimpan data orderbook ke tabel
- Menangani error database agar program tetap berjalan

4. config.py

File konfigurasi terpusat.

Berisi:

- URL API Stockbit
- Bearer Token
- Timeout request
- Konfigurasi database MySQL

Tujuannya agar konfigurasi mudah diubah tanpa mengubah logika program.

5. saham.txt

- Berisi daftar kode saham (ticker) yang akan disrape.
- Satu saham per baris
- Total ±851 emiten

Output Program

Contoh output berhasil tersedia di folder Lain-lain:

- stock_db_output.csv: hasil export database
- stock_db_dump.sql: dump database MySQL
- File Notes.txt berisi contoh log eksekusi program yang berhasil.

Kesimpulan

Program berhasil melakukan scraping data orderbook saham dari Stockbit, menyimpannya ke database, dan menampilkannya kembali melalui dashboard. Perbedaan struktur data dan orderbook kosong merupakan kondisi normal dari API, bukan kesalahan implementasi.