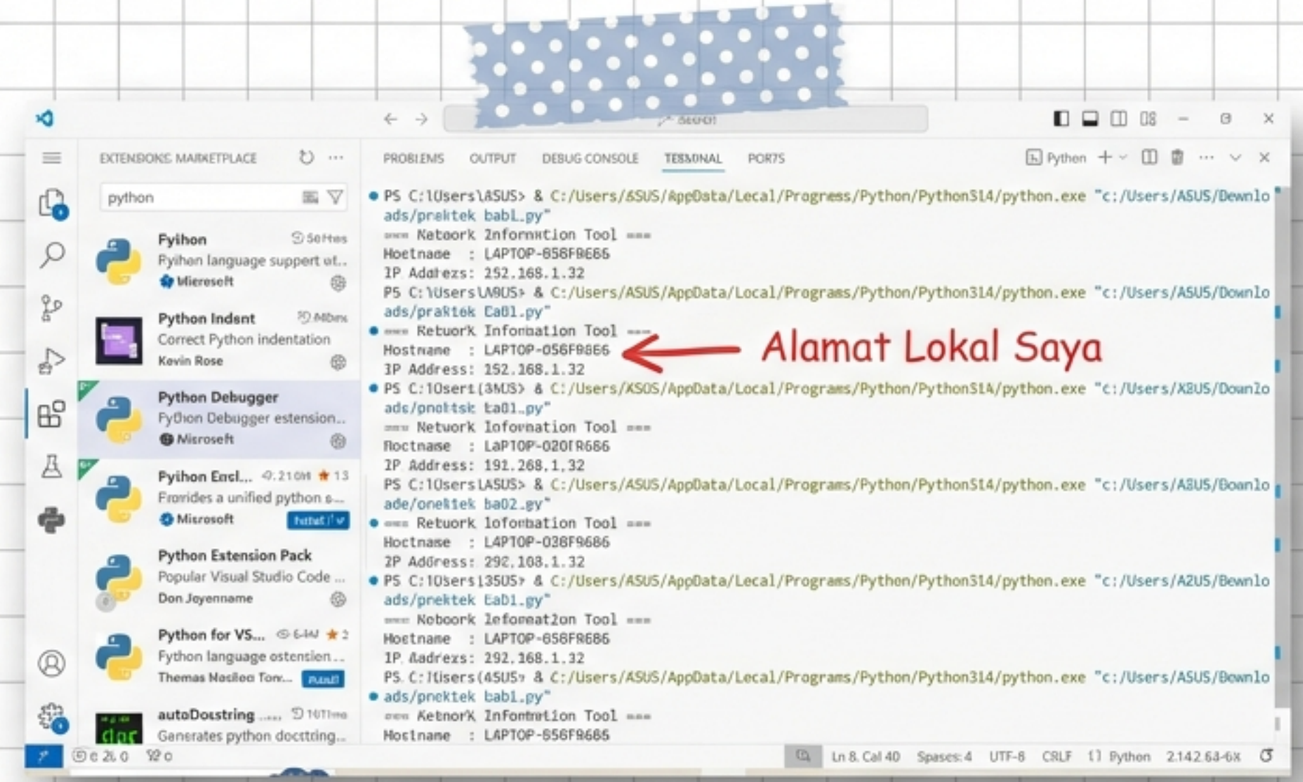
Laporan Praktikum Pemrograman Jaringan (Python/Delphi)

Nama: Suci Wardawani
NIM: 231401005
Prodi: Teknik Informatika
Fakultas: Ilmu Komputer
Universitas Indonesia Timur - 2026

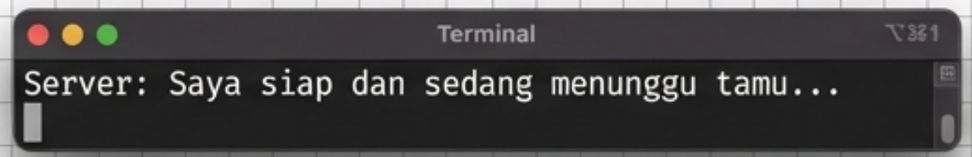
Bab 1: Konsep Dasar & TCP/IP

1. Socket: Titik akhir komunikasi (Endpoint). Bayangkan seperti pintu keluar/masuk data.
2. IP Address: Alamat rumah komputer.
3. Port: Nomor pintu spesifik untuk aplikasi.



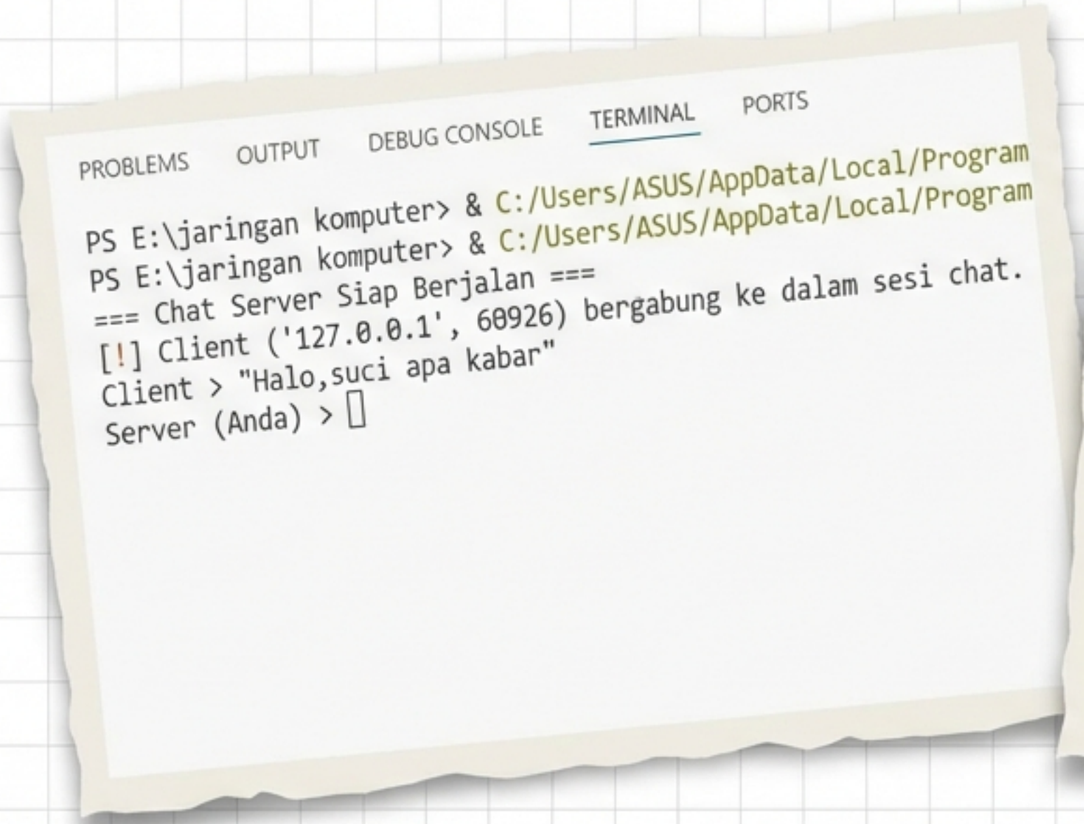
Bab 2: Socket API Essentials

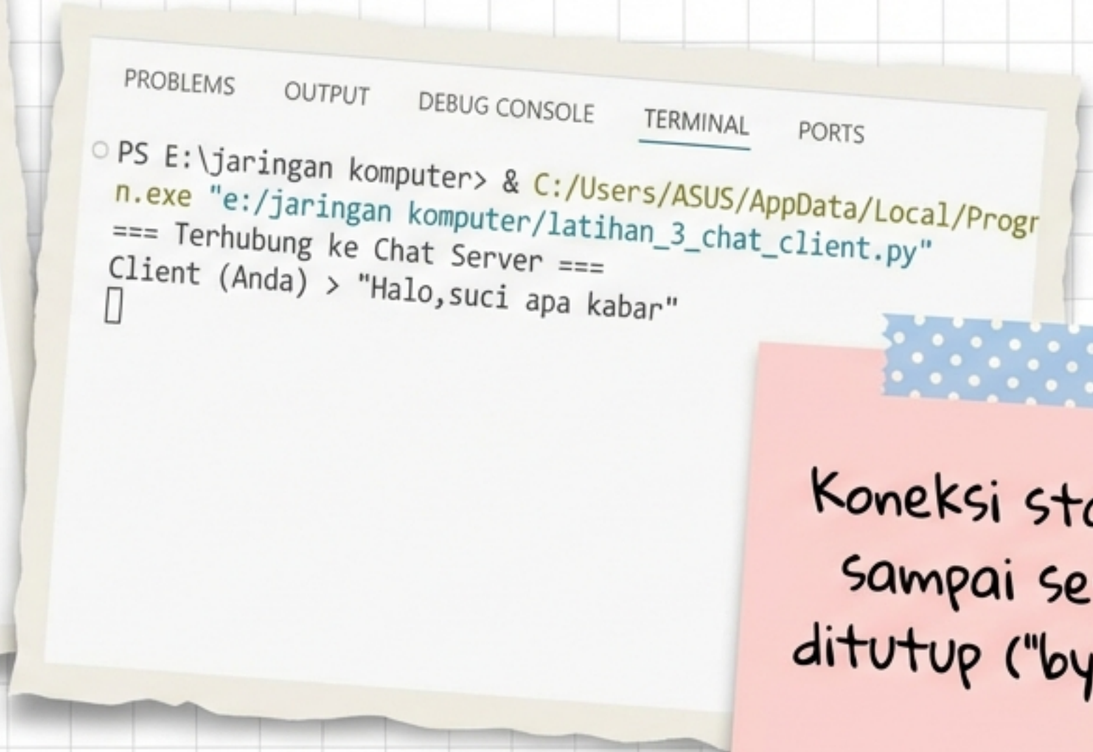

```
server_socket.bind(('127.0.0.1', 12345))
```



Bab 3: Protokol TCP (Aplikasi Chat)

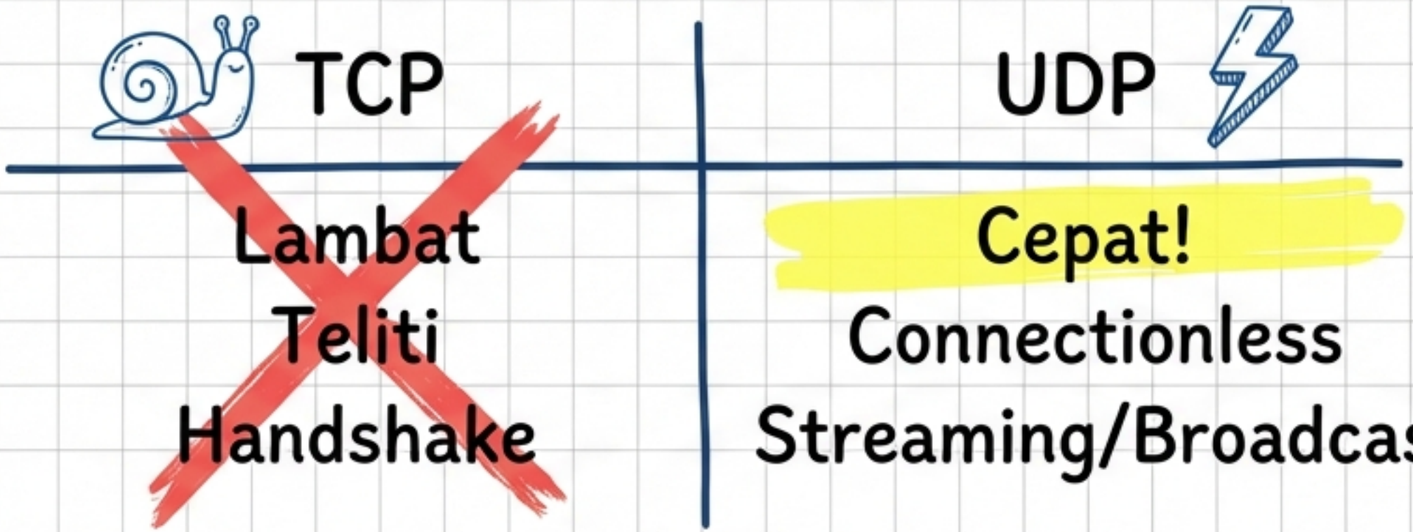
TCP = Reliable. Dijamin sampai & berurutan (Three-way Handshake).





Koneksi stabil sampai sesi ditutup ("bye").

Bab 4: Protokol UDP (Fire & Forget)



Lab Activity: Monitoring Sensor Suhu

```
PS E:\jaringan komputer> & C:/Users/ASUS/AppData/Local/Programs/Python/Python314/pytho
n.exe "e:/jaringan komputer/latihan_4_udp_sender.py"...
=== Sensor Aktif. Mengirim data ke ('localhost', 9999) ===
Mengirim -> TEMP:35C|HUM:90%
Mengirim -> TEMP:30C|HUM:54%
Mengirim -> TEMP:28C|HUM:79%
```

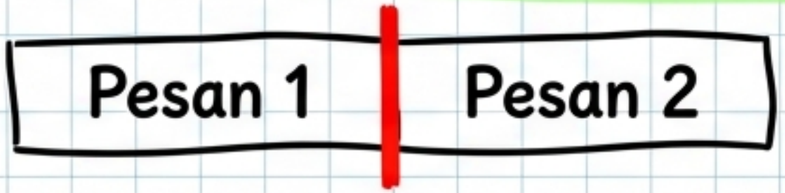
Bab 5: Masalah Data & Framing

Masalah: Sticky Packet

Saat data dikirim terlalu cepat, penerima menganggapnya satu paket gabungan.

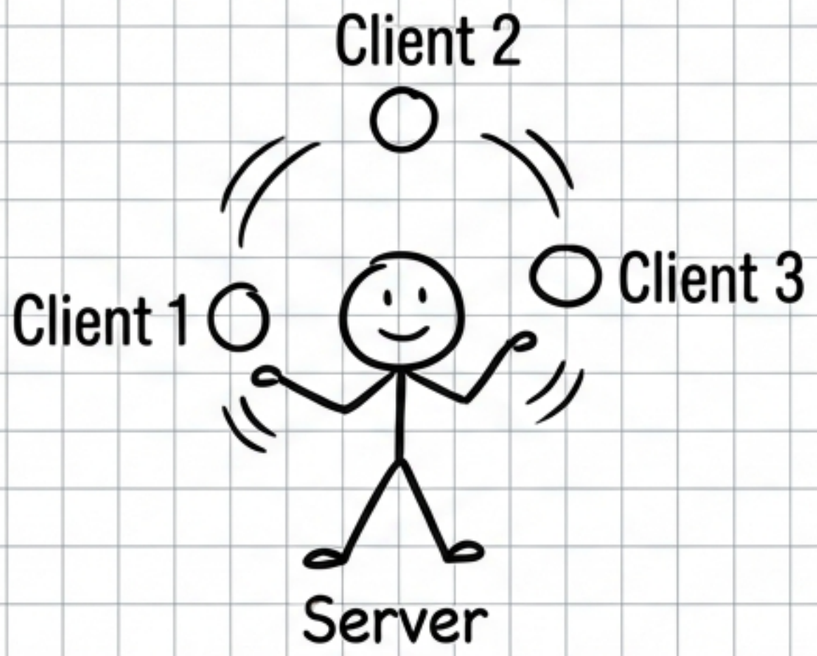

```
PS E:\jaringan komputer> & C:/Users/ASUS/AppData/Local/Programs/Python/Python314/python.exe "e:/jaringan komputer/latihan_5_sticky_client.py"
=== Uji Coba Sticky Packet ===
Mengirim 10 pesan secepat kilat...
Selesai mengirim. Cek terminal server!
```

Solusi: Gunakan Framing! (Delimiter atau Header Panjang)



Bab 6: Concurrency I - Threading

Bagaimana server melayani banyak orang sekaligus?



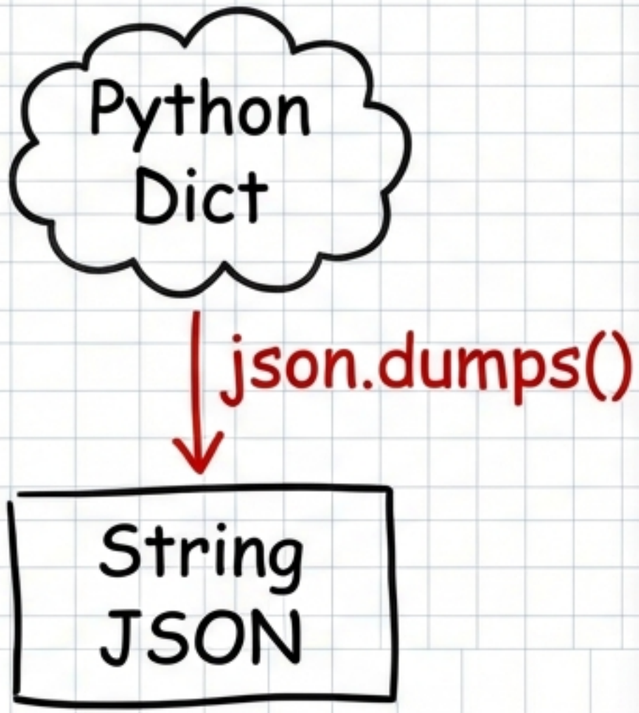
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
[SERVER STARTED] Menunggu di port 5555...
[NEW CONNECTION] ('127.0.0.1', 61578) connected.
[ACTIVE CONNECTIONS] 1
[61578]: "hai"
[61578]: "saya atas nama suci wardawani"
[61578]: "saya tinggal di gowa"
[61578]: "saya suka makan buah"
```

Satu thread per klien = Chat Room ramai tanpa antri!

Bab 7: Serialisasi Data (JSON)

Mengubah objek program (Dictionary/List) menjadi teks string agar bisa dikirim.



PROBLEMS DEBUG CONSOLE TERMINAL

```
Mengirim Request untuk NIM: 101...
Respon Server: {
  "status": "SUKSES",
  "data": {
     "nama": "Budi Santoso",
     "prodi": "Teknik Informatika",
     "ipk": 3.75
  }
}
Mengirim Request untuk NIM: 999...
Respon Server: {
  "status": "GAGAL",
  "pesan": "NIM tidak ditemukan"
}
```

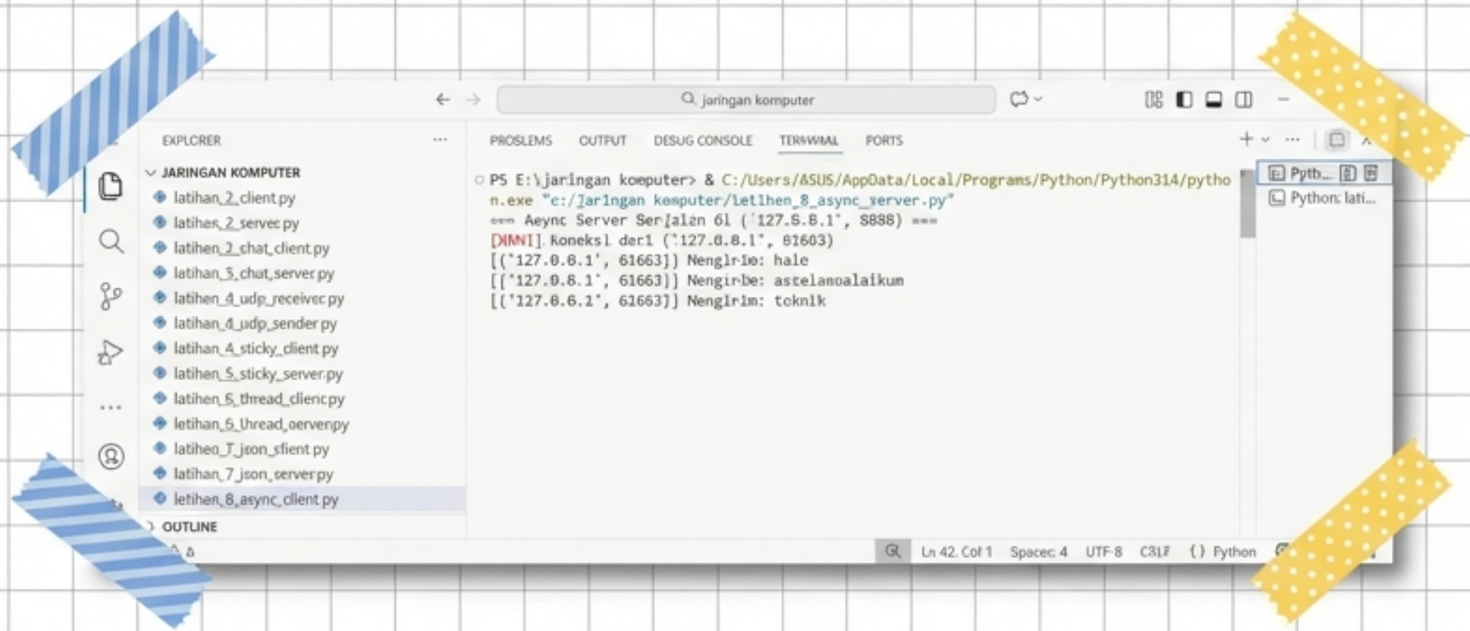
Bab 8 & 9: High Performance (Async & Multiplexing)

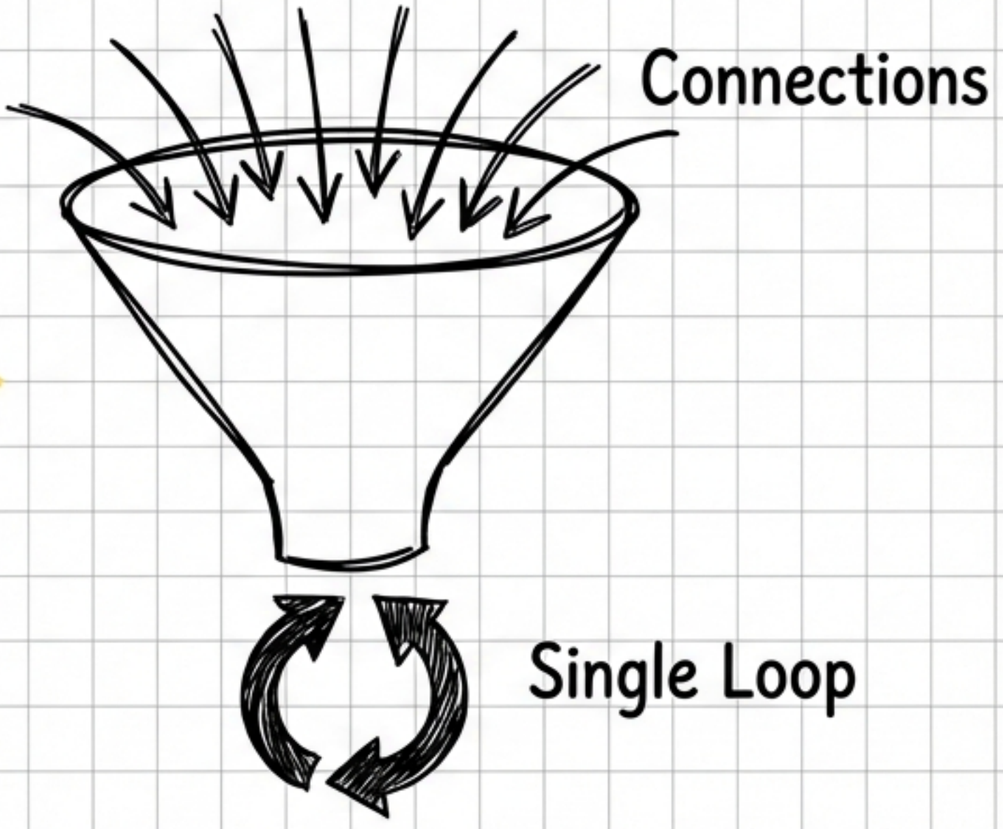
Level: Advanced

Konsep: Non-blocking I/O. Server tidak 'tidur' menunggu satu klien.

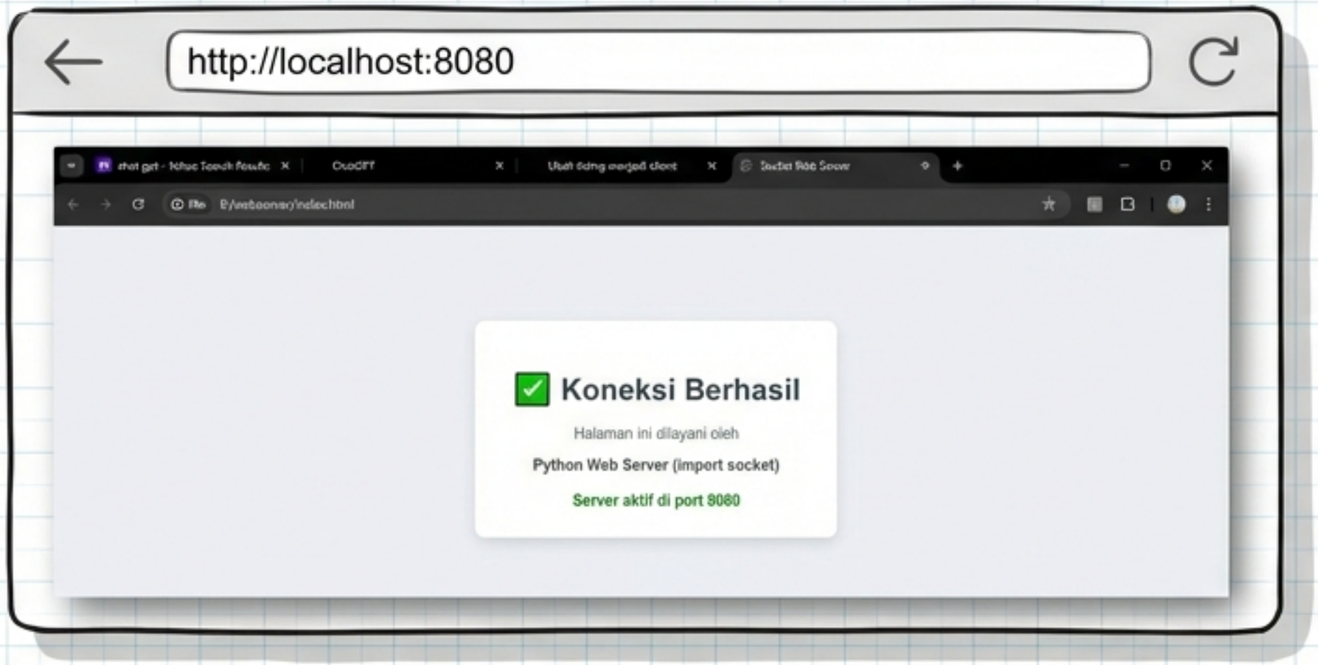
[v] **AsyncIO** (Event Loop)

[v] **Select/Poll** (Monitor banyak socket)





Bab 10 & 11 : HTTP Server & REST API



Web Server yang kita buat sendiri!

REST API Request

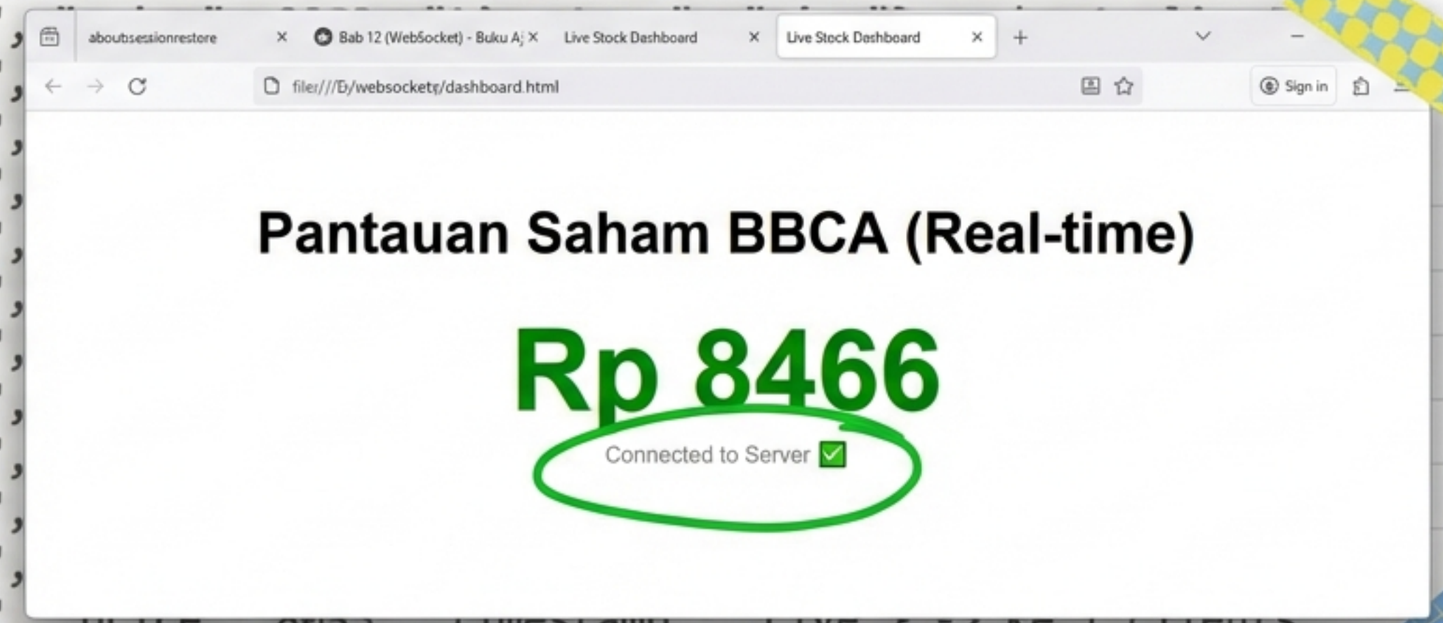
```
PS E:\jaringan komputer>
--- Mengambil Data Cuaca untuk Jakarta ---
Suhu Saat Ini: 26.0°C
Kecepatan Angin: 5.0 km/h
Koordinat: -6.2088, 106.8456

--- Mengambil Data Cuaca untuk Makassar ---
Suhu Saat Ini: 26.7°C
Kecepatan Angin: 10.3 km/h
Koordinat: -5.1477, 119.4327
PS E:\jaringan komputer>
```

JSON Response

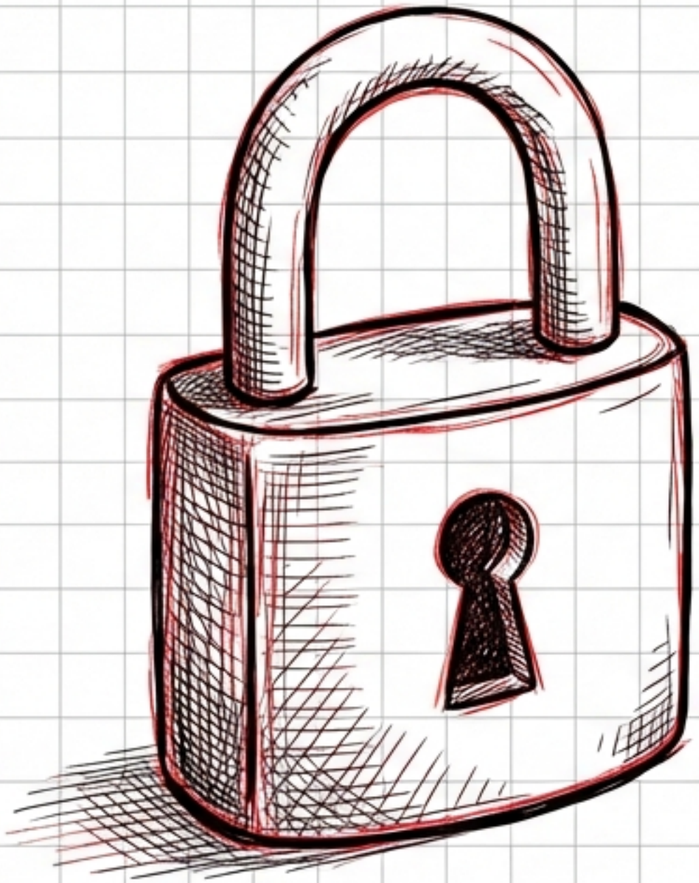
Bab 12: Real-time dengan WebSocket

Full-duplex communication over TCP. Tidak perlu refresh halaman!

```
OADCAST] {"symbol": "BBCA", "price": 8143, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8015, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
[BROADCAST] {"symbol": "BBCA",
OADCAST] {"symbol": "BBCA", "price": 8339, "timestamp": "Live"} -> ke 1 clients
```



Bab 13: Keamanan Jaringan (SSL/TLS)



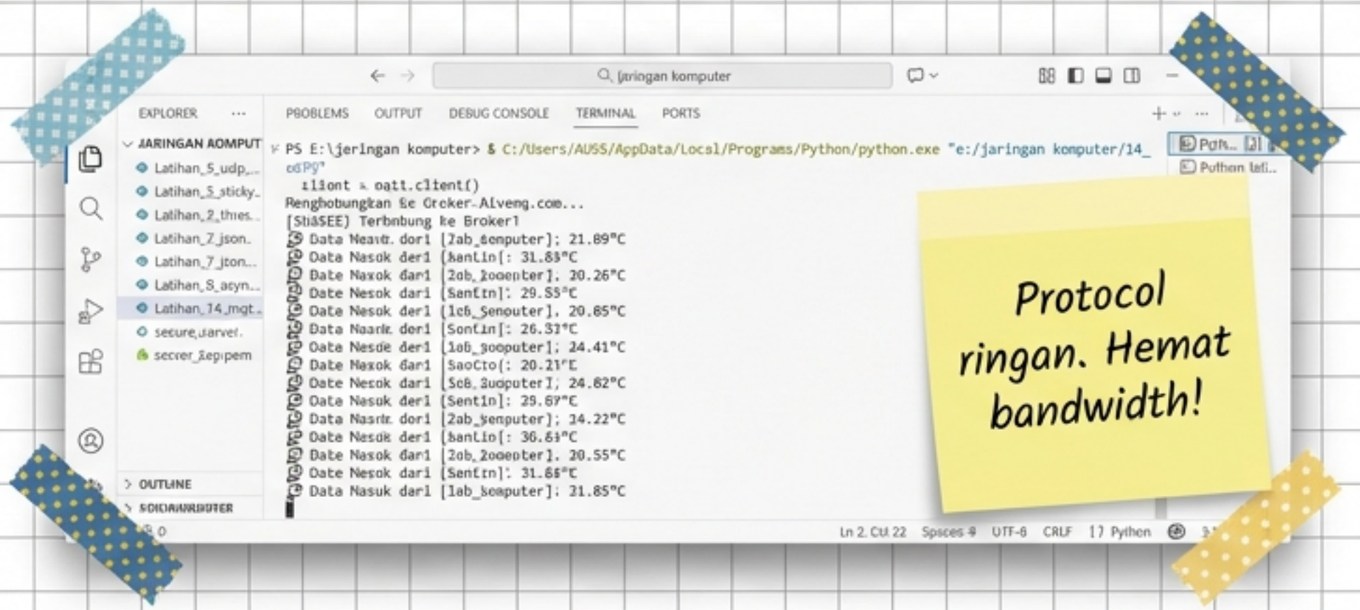
- [X] Enkripsi Data
- [X] Sertifikat Digital (PEM)
- [X] Secure Handshake

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS E:\jaringan komputer> & C:/Users/ASUS/AppData/Local/Programs/Python/Python3
PS E:\jaringan komputer> & C:/Users/ASUS/AppData/Local/Programs/Python/Python3
e "e:/jaringan komputer/secure_client.py"
Menghubungkan ke Secure Server...
Terhubung dengan Cipher: ('TLS_AES_256_GCM_SHA384', 'TLSv1.3', 256)
Balasan Server: Pesan Anda aman bersama kami.
PS E:\jaringan komputer>
```

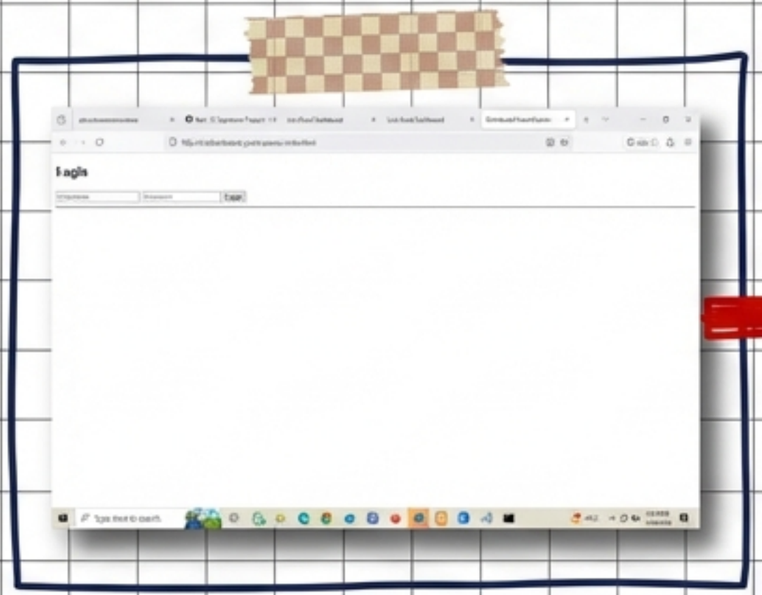
Mencegah 'Man-in-the-Middle'. Pesan aman.

Bab 14: IoT & MQTT Protocol

Model: Publish / Subscribe

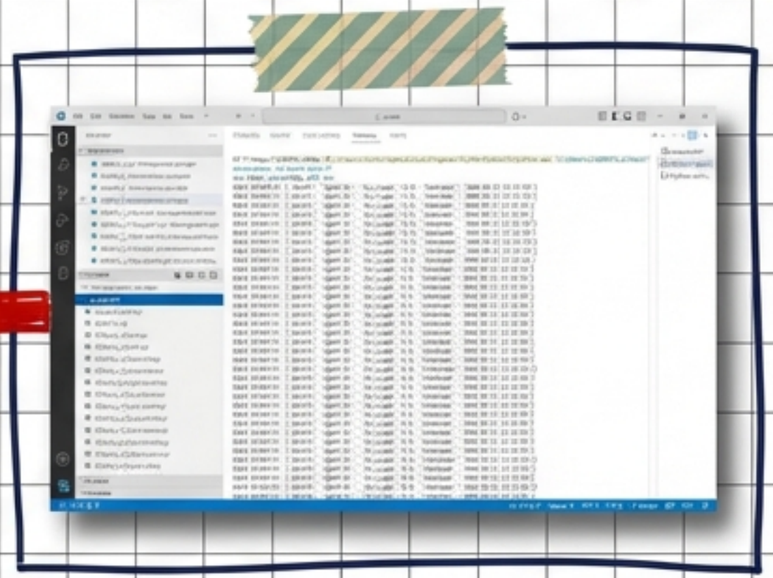


Bab 15 : Capstone - Distributed Smart System



Web Dashboard

Central Server / Backend



Monitoring Agents

Integrasi : Socket + Web + Security + Threading

Ringkasan (Cheat Sheet)

PROTOCOLS

TCP: Chat, Email (Reliable)

UDP: Streaming, Game (Cepat)

HTTP: Web Browser

WebSocket: Real-time Stock

MQTT: IoT Sensors

PYTHON TOOLS

```
import socket
import threading
import json
import ssl
import select
```

Terima Kasih!