

LAPORAN TUGAS PRAKTIKUM PEMROGRAMAN JARINGAN

Mata Kuliah: Bahasa Pemrograman Jaringan Komputer



Di Susun Oleh:

Deslin Paredanun (231401009)

Dosen Pengampu: Ucok, S.kom.,MT

FAKULTAS ILMU KOMPUTER

PROGRAM STUDI TEKNIK INFORMATIKA

UNIVERSITAS INDONESIA TIMUR

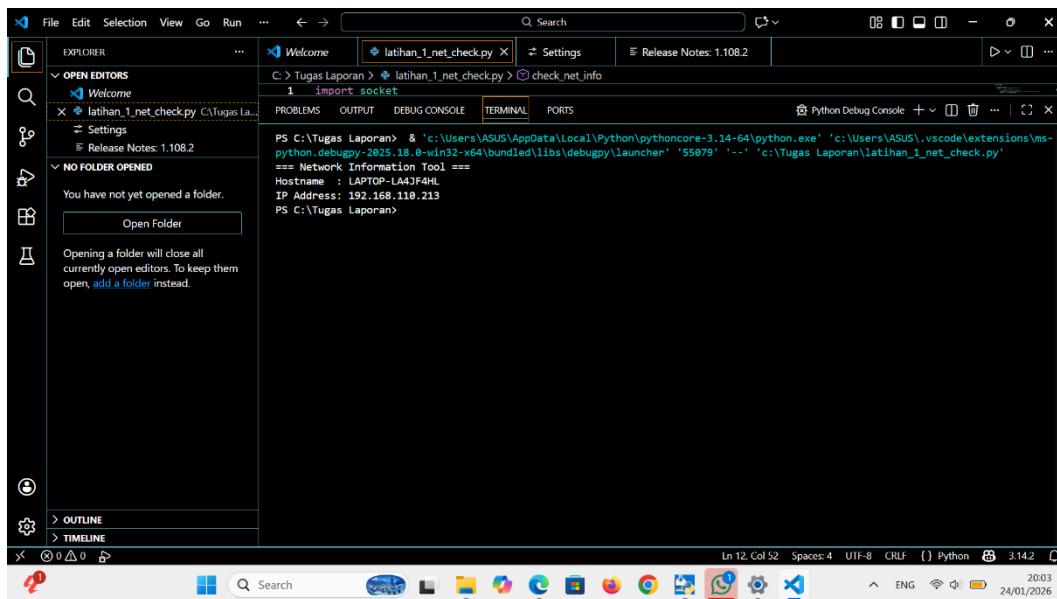
MAKASSAR

2026

BAB 1 KONSEP DASAR PEMROGRAMAN JARINGAN

Latihan ini membuat alat diagnostik jaringan sederhana menggunakan Python untuk mengecek identitas server. Script ini menampilkan hostname (nama komputer atau server) dan IP address lokal. Informasi ini berguna untuk memastikan aplikasi berjalan di server yang tepat, terutama saat deploy ke layanan cloud seperti AWS atau GCP.

Hasil:



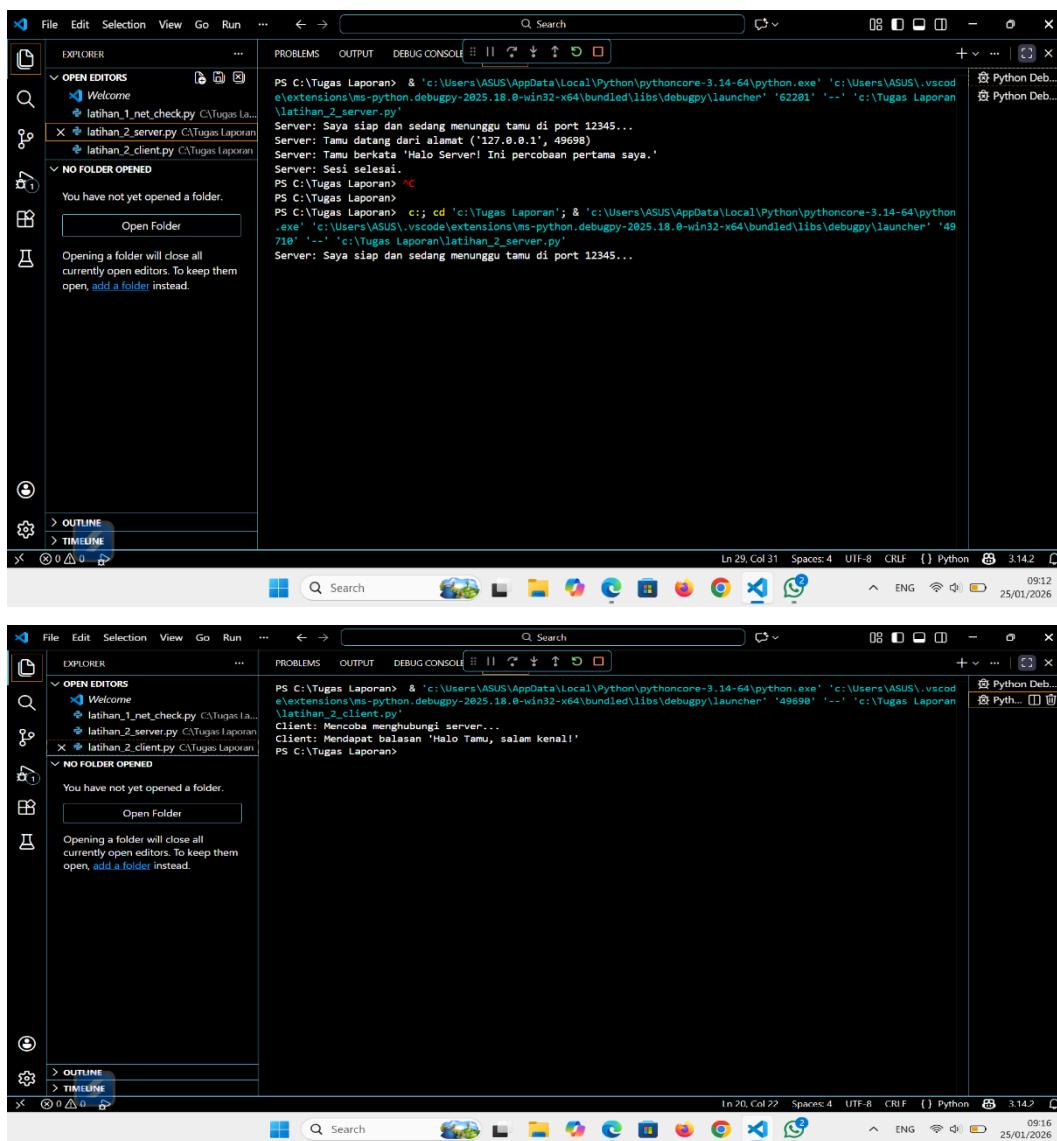
The screenshot shows the Visual Studio Code interface. The terminal tab is active, displaying the output of a Python script. The script imports the socket module and prints the host name and IP address. The output is as follows:

```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '55079' '--' 'c:\Tugas Laporan\latihan_1_net_check.py'
== Network Information Tool ==
Hostname : LAPTOP-LA4JF4HL
IP Address: 192.168.118.213
PS C:\Tugas Laporan>
```

BAB 2 SOCKET API DASAR

Latihan ini memperkenalkan konsep komunikasi client-server menggunakan TCP socket di Python. Server dibuat terlebih dahulu, menunggu koneksi dari client melalui `accept()`, sementara client menghubungi server menggunakan `connect()`. Client mengirim pesan, server menerima dan membalas, kemudian koneksi ditutup di kedua sisi. Latihan ini mengajarkan bagaimana server dan client saling bertukar data secara sinkron, bagaimana blocking call bekerja di server, serta pentingnya encoding/decoding data dari string ke bytes dan sebaliknya. Konsep ini adalah dasar dari semua aplikasi jaringan, mulai dari chat sederhana hingga layanan cloud yang kompleks.

Hasil:



The image shows two side-by-side terminal windows in VS Code, each displaying the interaction between a client and a server for a simple socket application.

Terminal 1 (Server):

```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '62201' '--' 'c:\Tugas Laporan\latihan_2_server.py'
Server: Saya siap dan sedang menunggu tamu di port 12345...
Server: Tamu datang dari alamat ('127.0.0.1', 49698)
Server: Tamu berkata 'Halo Server! Ini percobaan pertama saya.'
Server: Sesi selesai.
PS C:\Tugas Laporan> ^C
PS C:\Tugas Laporan>
PS C:\Tugas Laporan> cd 'c:\Tugas Laporan'; & 'c:\Users\ASUS\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '49710' '--' 'c:\Tugas Laporan\latihan_2_server.py'
Server: Saya siap dan sedang menunggu tamu di port 12345...
```

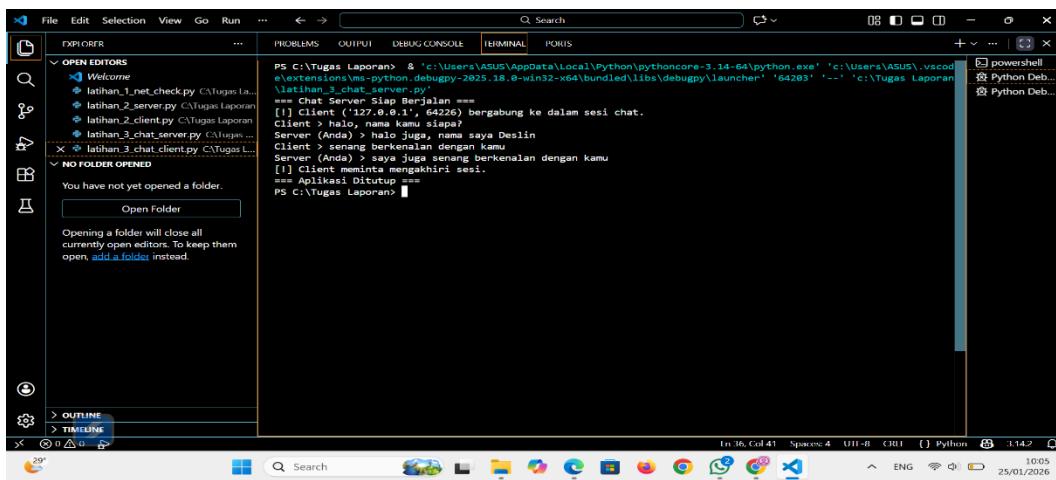
Terminal 2 (Client):

```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '49698' '--' 'c:\Tugas Laporan\latihan_2_client.py'
Client: Mencoba menghubungi server...
Client: Mendapat balasan 'Halo Tamu, salam kenal!'
PS C:\Tugas Laporan>
```

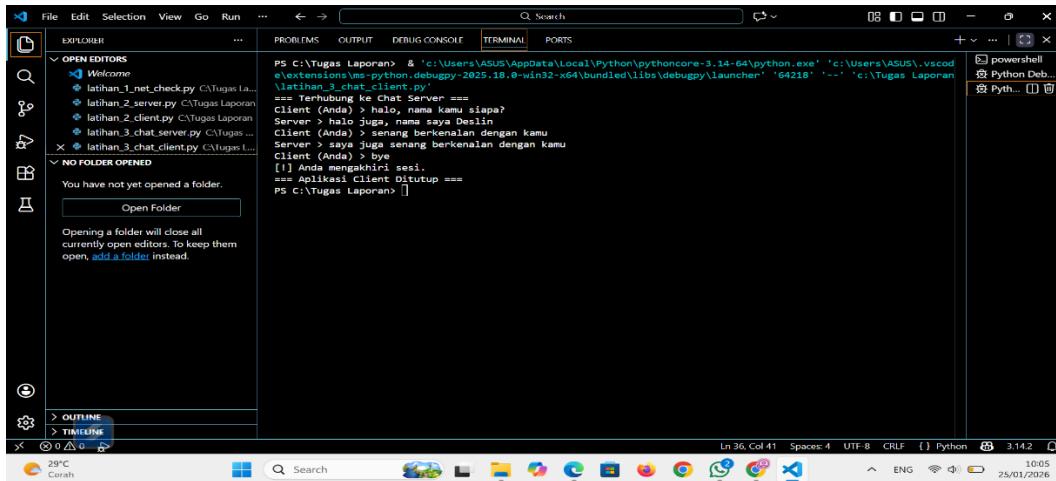
BAB 3 PROTOKOL TCP (APLIKASI CHAT)

Latihan ini memperkenalkan konsep persistent connection pada aplikasi chat, di mana server dan client dapat saling bertukar pesan berulang kali tanpa perlu membuat koneksi baru setiap kali. Dengan menggunakan loop tak terbatas (`while True`) dan kondisi keluar ("bye"), komunikasi tetap terjaga sampai salah satu pihak memutuskan untuk mengakhiri sesi. Latihan ini juga menunjukkan keterbatasan model blocking karena `recv()` dan `input()` menunggu giliran, percakapan hanya bisa berjalan secara gantian seperti walkie-talkie. Hal ini menjadi dasar untuk memahami mengapa threading atau concurrency diperlukan agar chat bisa real-time, memungkinkan kedua pihak mengirim dan menerima pesan secara bersamaan, mirip aplikasi seperti WhatsApp atau Telegram.

Hasil:



```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '6426' '--' 'c:\Tugas Laporan\latihan_3_chat_server.py'
[!] Chat Server Siap Berjalan ===
Client > halo, nama kamu siapa?
Server (Anda) > halo juga, nama saya Deelin
Client > senang berkenalan dengan kamu
Server (Anda) > saya juga senang berkenalan dengan kamu
[!] Client Meminta Keluar Dari Sesi
[!] Anda mengakhiri sesi.
*** Aplikasi Ditutup ===
PS C:\Tugas Laporan>
```



```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '6428' '--' 'c:\Tugas Laporan\latihan_3_chat_client.py'
[!] Terhubung Ke Chat Server ===
Client (Anda) > halo, nama kamu siapa?
Server > halo juga, nama saya Deelin
Client (Anda) > senang berkenalan dengan kamu
Server > saya juga senang berkenalan dengan kamu
Client (Anda) > bye
[!] Anda mengakhiri sesi.
*** Aplikasi Client Ditutup ===
PS C:\Tugas Laporan>
```

BAB 4 PROTOKOL UDP (STREAMING & BROADCASTING)

Latihan ini memperlihatkan bagaimana sensor IoT dapat mengirim data lingkungan secara real-time ke server pusat menggunakan protokol UDP, yang bersifat connectionless. Sensor (client) secara periodik membaca data suhu dan kelembaban, kemudian mengirim paket ke server tanpa perlu membuat koneksi permanen. Di sisi server, data diterima dari siapa saja dan langsung ditampilkan, tanpa harus melacak status masing-masing sensor. Pendekatan ini cepat, ringan, dan hemat resource, sehingga ideal untuk sistem monitoring banyak sensor sekaligus. Latihan ini juga mengajarkan perbedaan mendasar antara UDP dan TCP, serta bagaimana komunikasi IoT sederhana bisa diimplementasikan dengan efisien.

Hasil:

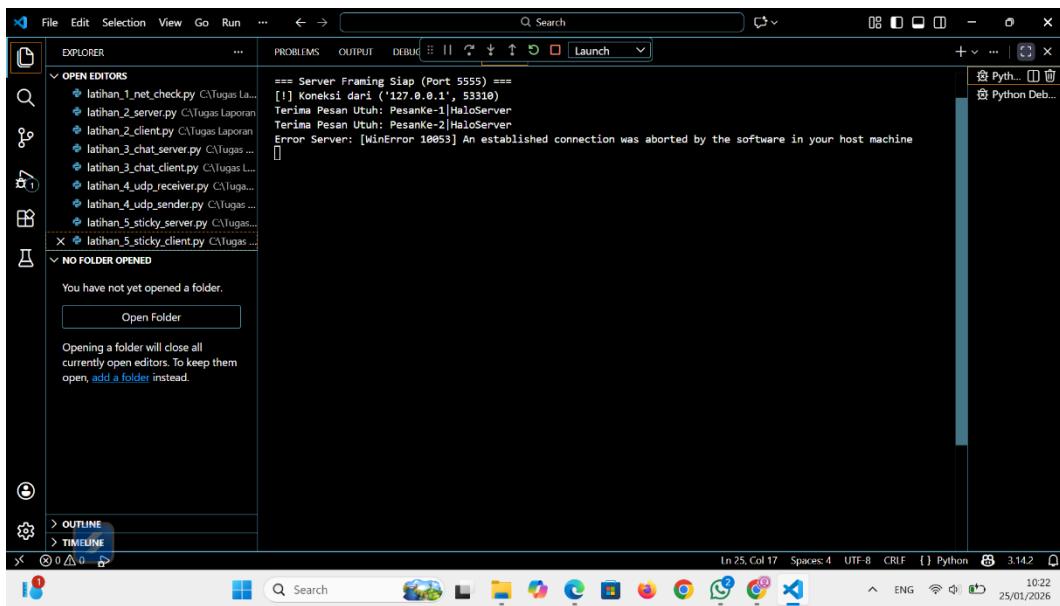
```
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:25C|HUM:68%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:22C|HUM:52%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:32C|HUM:58%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:23C|HUM:78%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:27C|HUM:76%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:32C|HUM:56%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:34C|HUM:49%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:27C|HUM:82%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:35C|HUM:89%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:26C|HUM:88%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:27C|HUM:74%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:23C|HUM:84%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:21C|HUM:84%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:29C|HUM:41%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:30C|HUM:44%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:31C|HUM:54%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:30C|HUM:88%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:32C|HUM:53%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:34C|HUM:45%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:28C|HUM:49%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:33C|HUM:60%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:29C|HUM:42%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:29C|HUM:46%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:34C|HUM:58%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:33C|HUM:84%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:21C|HUM:87%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:31C|HUM:53%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:26C|HUM:60%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:36C|HUM:41%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:32C|HUM:86%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:23C|HUM:87%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:21C|HUM:48%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:29C|HUM:54%
[Sensor ('127.0.0.1', 51195)] Melaporkan: TEMP:26C|HUM:45%
```

```
Mengirim -> TEMP:27C|HUM:76%
Mengirim -> TEMP:32C|HUM:58%
Mengirim -> TEMP:34C|HUM:49%
Mengirim -> TEMP:27C|HUM:82%
Mengirim -> TEMP:35C|HUM:89%
Mengirim -> TEMP:26C|HUM:88%
Mengirim -> TEMP:27C|HUM:74%
Mengirim -> TEMP:23C|HUM:84%
Mengirim -> TEMP:29C|HUM:41%
Mengirim -> TEMP:36C|HUM:43%
Mengirim -> TEMP:31C|HUM:54%
Mengirim -> TEMP:27C|HUM:88%
Mengirim -> TEMP:30C|HUM:54%
Mengirim -> TEMP:34C|HUM:45%
Mengirim -> TEMP:28C|HUM:49%
Mengirim -> TEMP:33C|HUM:60%
Mengirim -> TEMP:29C|HUM:42%
Mengirim -> TEMP:29C|HUM:46%
Mengirim -> TEMP:34C|HUM:58%
Mengirim -> TEMP:33C|HUM:84%
Mengirim -> TEMP:21C|HUM:87%
Mengirim -> TEMP:31C|HUM:53%
Mengirim -> TEMP:26C|HUM:60%
Mengirim -> TEMP:36C|HUM:41%
Mengirim -> TEMP:32C|HUM:86%
Mengirim -> TEMP:23C|HUM:87%
Mengirim -> TEMP:29C|HUM:40%
Mengirim -> TEMP:24C|HUM:78%
Mengirim -> TEMP:25C|HUM:70%
```

BAB 5 ERROR HANDLING & FRAMING DAT

Latihan ini menunjukkan cara menangani sticky packet pada TCP, yaitu kondisi di mana beberapa pesan yang dikirim cepat menyatu menjadi satu blok sehingga server sulit membedakan pesan asli. Solusinya adalah menggunakan delimiter ('\n') dan fitur `makefile()` pada socket, sehingga server bisa membaca pesan **per baris dengan `readline()` atau iterasi `for line in stream`**. Dengan cara ini, setiap pesan tetap utuh meski client mengirim banyak pesan sekaligus. Selain framing, latihan ini juga menekankan pentingnya error handling untuk membuat aplikasi jaringan lebih tahan banting, misalnya dengan `timeout('sock.settimeout')` dan menangani `ConnectionResetError` atau `BrokenPipeError`, agar server dan client tetap stabil meski terjadi gangguan koneksi. Pendekatan ini menjadi praktik penting dalam membangun aplikasi TCP yang robust, reliable, dan mudah di-maintain.

Hasil:



The screenshot shows the Microsoft Visual Studio Code interface. The Explorer sidebar on the left lists several Python files under 'OPEN EDITORS' and a message 'NO FOLDER OPENED'. The terminal window on the right displays the output of a Python script named 'latihan_5_sticky_client.py'. The script uses the 'ms-python.debugpy' extension to connect to a server at port 53302 and send a message. The terminal output is as follows:

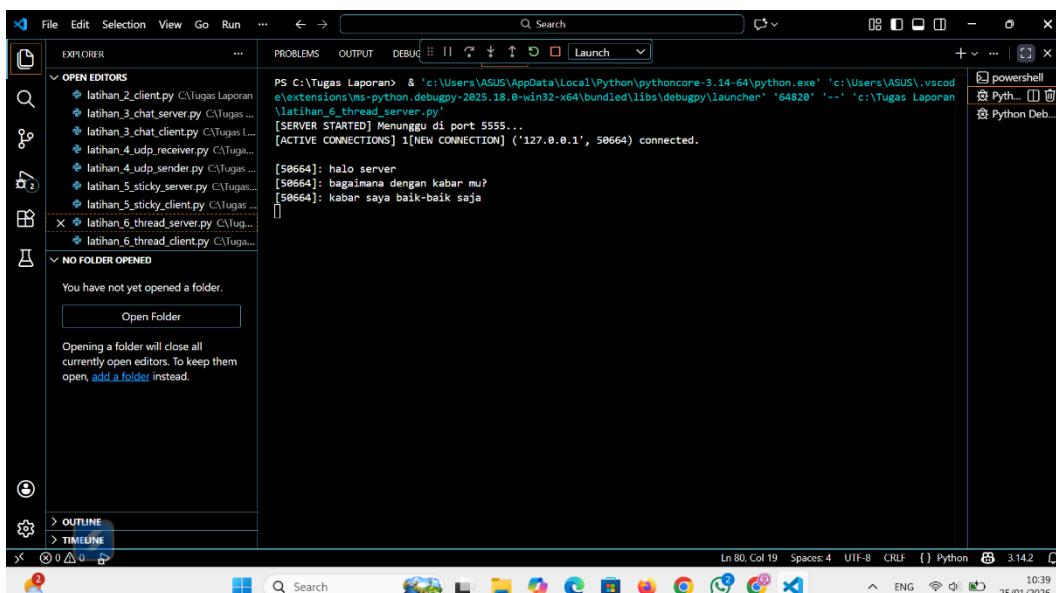
```
e:\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '53302' '--' 'c:\Tugas Laporan\\latihan_5_sticky_client.py'
*** Uji Coba Sticky Packet ***
Mengirim 10 pesan secepat kilat...
Selesai mengirim. Cek terminal Server!
PS C:\Tugas Laporan>
```

The status bar at the bottom indicates the current file is 'Python' and the date and time are '25/01/2026'.

BAB 6 CONCURRENCY PART I-THREADING

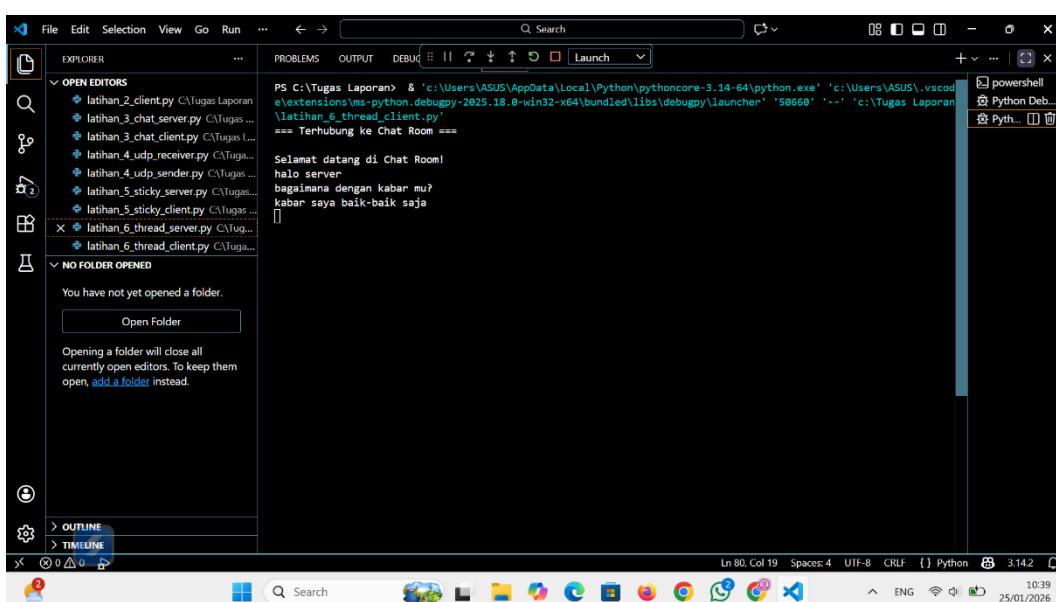
Latihan ini menunjukkan cara membuat chat room multi-client di mana pesan dari satu client disebarluaskan ke semua client lain. Setiap client dijalankan di thread terpisah, dan daftar koneksi client dijaga dengan lock agar tidak terjadi konflik saat menambah atau menghapus client. Pendekatan ini memungkinkan komunikasi real-time dan simultan antara banyak client secara aman dan stabil.

Hasil:



```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '58660' '--' 'c:\Tugas Laporan\latihan_6_thread_server.py'
[SERVER STARTED] Menunggu di port 5555...
[ACTIVE CONNECTIONS] 1[NEW CONNECTION] ('127.0.0.1', 50664) connected.

[50664]: halo server
[50664]: bagaimana dengan kabar mu?
[50664]: kabar saya baik-baik saja
```



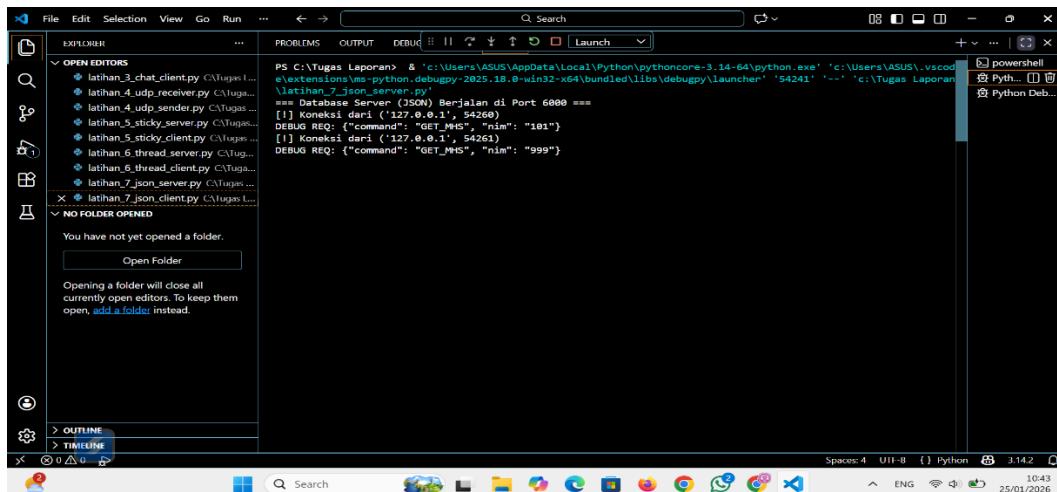
```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '58660' '--' 'c:\Tugas Laporan\latihan_6_thread_client.py'
*** Terhubung ke Chat Room ===

Selamat datang di Chat Room!
halo server
bagaimana dengan kabar mu?
kabar saya baik-baik saja
```

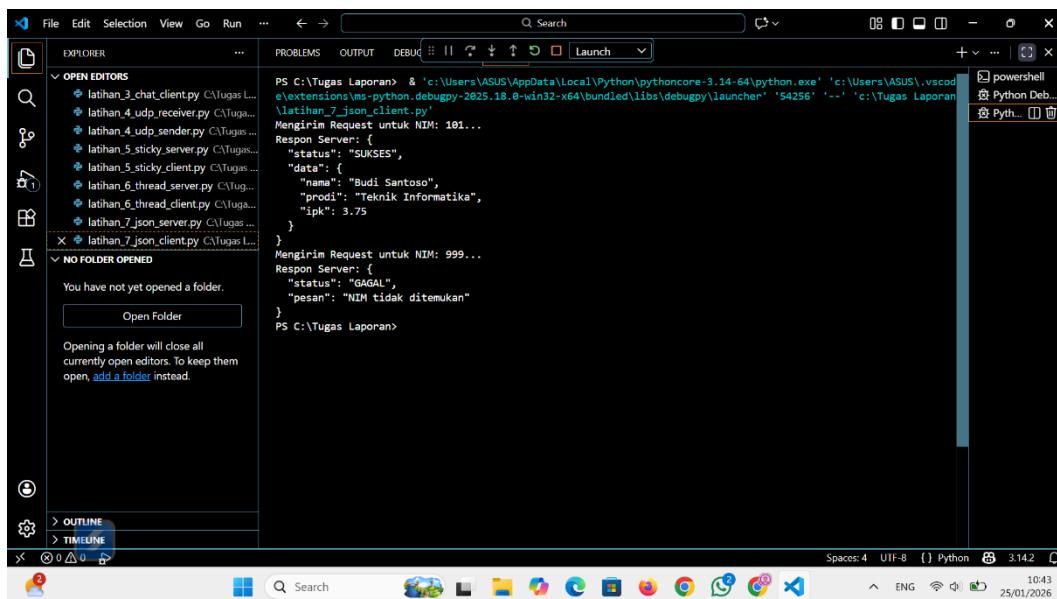
BAB 7 SERIALISASI DATA (JSON & PICKEL)

Latihan ini memperlihatkan cara membangun server database sederhana menggunakan TCP dan format JSON. Client mengirim perintah dalam bentuk JSON, misalnya `{"command": "GET_MHS", "nim": "101"}`, dan server menanggapi dengan JSON berisi data mahasiswa atau pesan error jika NIM tidak ditemukan atau format salah. Pendekatan ini mengajarkan konsep serialisasi/deserialisasi data (dict JSON bytes), protokol request-response sederhana, serta bagaimana membuat komunikasi client-server terstruktur, mudah dipahami, dan robust terhadap kesalahan format data. Latihan ini menjadi dasar penting untuk membuat API atau layanan database ringan berbasis TCP.

Hasil:



```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '54241' '--' 'c:\Tugas Laporan\latihan_7_json_server.py'
[!] Database Server (JSON) Berjalan di Port 6000 ====
[!] Koneksi dari ('127.0.0.1', 54240)
DEBUG REQ: {"command": "GET_MHS", "nim": "101"}
[!] Koneksi dari ('127.0.0.1', 54241)
DEBUG REQ: {"command": "GET_MHS", "nim": "999"}
```

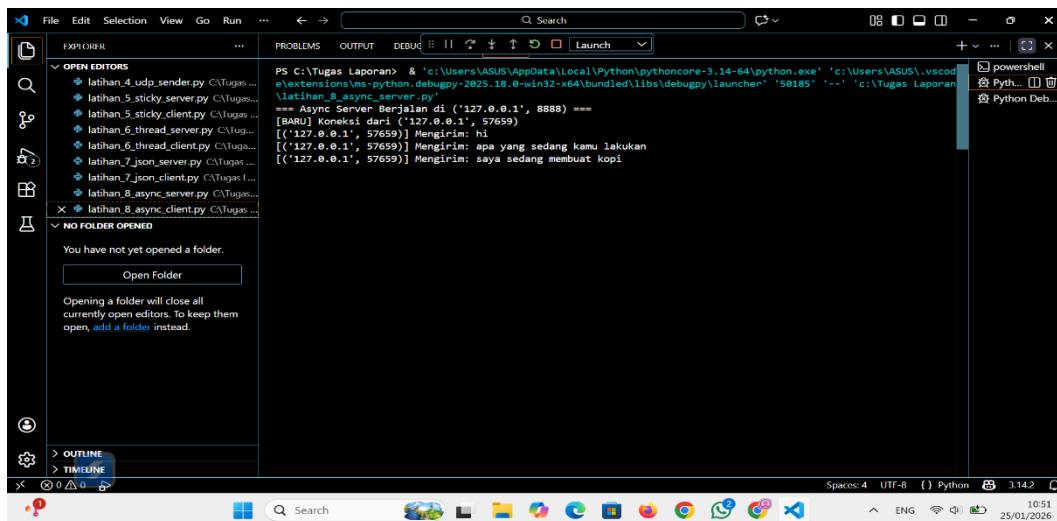


```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '54256' '--' 'c:\Tugas Laporan\latihan_7_json_client.py'
Mengirim Request untuk NIM: 101...
Respon Server: {
    "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"
}
PS C:\Tugas Laporan>
```

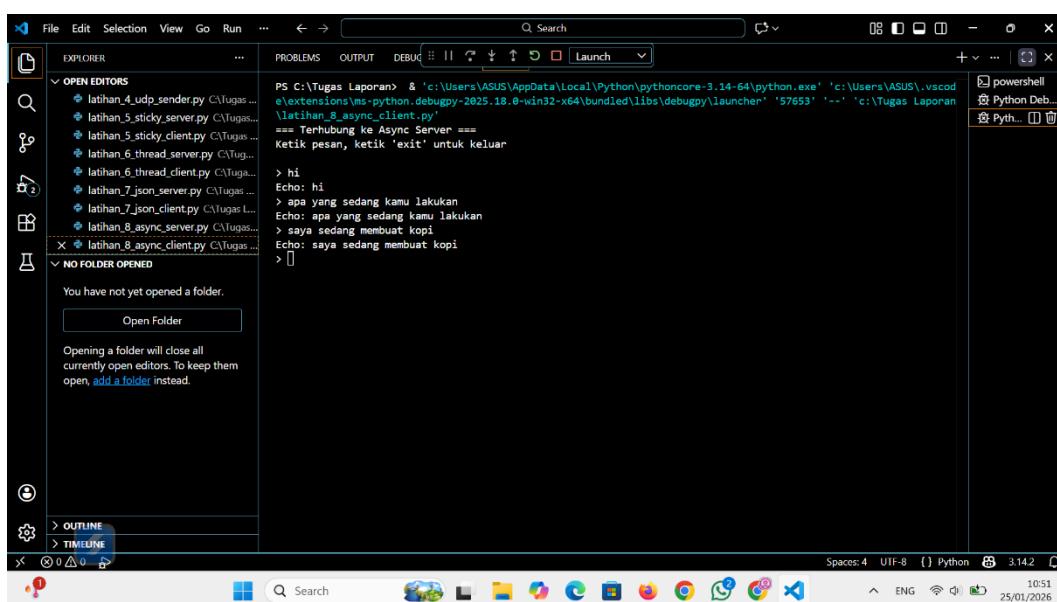
BAB 8 ASYNCHRONOUS I/O (CONCURRENCY PART II)

Latihan ini memperkenalkan server TCP high-performance menggunakan `asyncio`, di mana setiap client ditangani oleh coroutine secara non-blocking di satu thread. Dengan kata lain, server bisa melayani ribuan koneksi secara bersamaan tanpa membuat thread baru untuk tiap client. Konsep penting yang dipelajari adalah event loop dan penggunaan `await` untuk membaca dan menulis data tanpa menghentikan server, serta bahaya blocking code seperti `time.sleep()` yang bisa menghentikan seluruh loop. Pendekatan ini sangat efisien untuk aplikasi real-time berskala besar, menggunakan RAM minimal namun tetap responsif.

Hasil:



```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '50185' '--' 'c:\Tugas Laporan\latihan_8_async_server.py'
*** Async Server Berjalan di ('127.0.0.1', 8888)
[BARU] Koneksi dari ('127.0.0.1', 57659)
[('127.0.0.1', 57659)] Mengirim: hi
[('127.0.0.1', 57659)] Mengirim: apa yang sedang kamu lakukan
[('127.0.0.1', 57659)] Mengirim: saya sedang membuat kopi
```



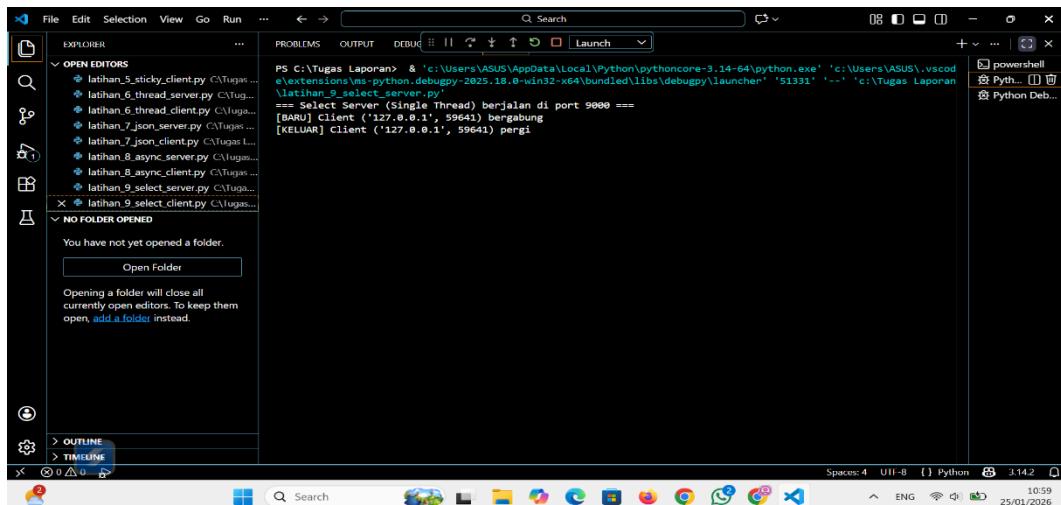
```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '57653' '--' 'c:\Tugas Laporan\latihan_8_async_client.py'
*** Terhubung ke Async Server ===
Ketik pesan, ketik 'exit' untuk keluar

> hi
Echo: hi
> apa yang sedang kamu lakukan
Echo: apa yang sedang kamu lakukan
> saya sedang membuat kopi
Echo: saya sedang membuat kopi
> []
```

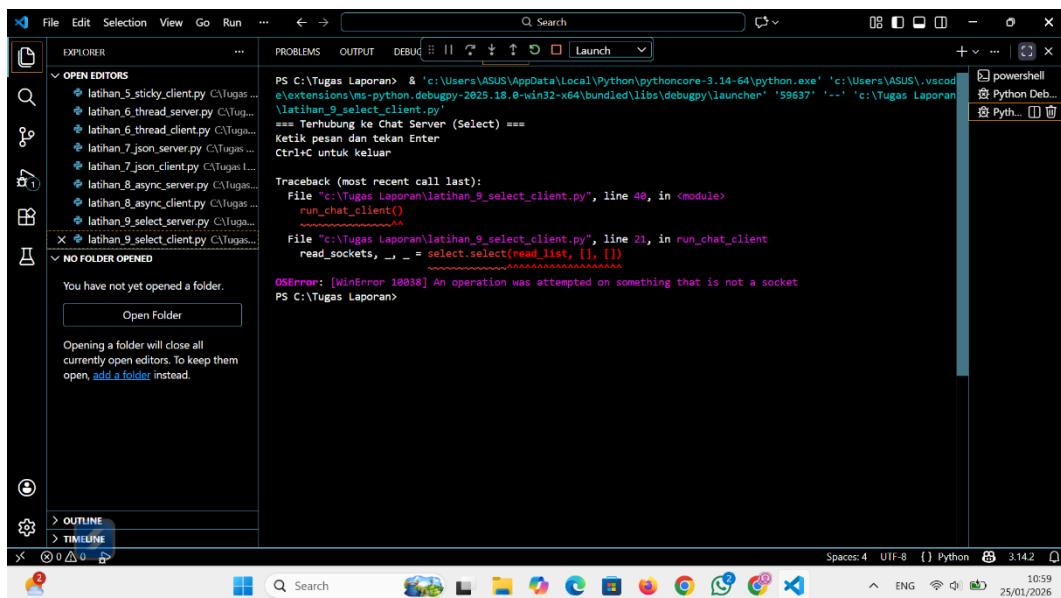
BAB 9 I/O MULTIPLEXING (SELECT & POLL)

Latihan ini menunjukkan cara membuat chat server efisien tanpa thread menggunakan select untuk memantau banyak socket secara bersamaan. Server menggunakan satu thread untuk menangani semua client, memeriksa socket mana yang aktif (baru connect atau mengirim pesan), kemudian menambahkan client baru atau melakukan broadcast pesan. Pendekatan ini hemat memori karena tidak ada overhead thread per client, tetapi responsif, dan mampu melayani banyak koneksi secara simultan. Namun, semua operasi tetap blocking per 'select()', sehingga desain ini cocok untuk server skala menengah sebelum menggunakan 'asyncio' untuk skala besar.

Hasil:



A screenshot of the Visual Studio Code (VS Code) interface. The Explorer sidebar shows several Python files: latihan_5_sticky_client.py, latihan_6_thread_server.py, latihan_7_thread_client.py, latihan_7_json_server.py, latihan_7_json_client.py, latihan_8_async_server.py, latihan_8_async_client.py, latihan_9_select_server.py, and latihan_9_select_client.py. The terminal window displays the output of a Python script named latihan_9_select_server.py. The output shows the server listening on port 9000 and accepting a connection from a client at 127.0.0.1:59641. The client sends a message, and the server responds. The status bar at the bottom right indicates the date as 25/01/2026 and the time as 10:59.

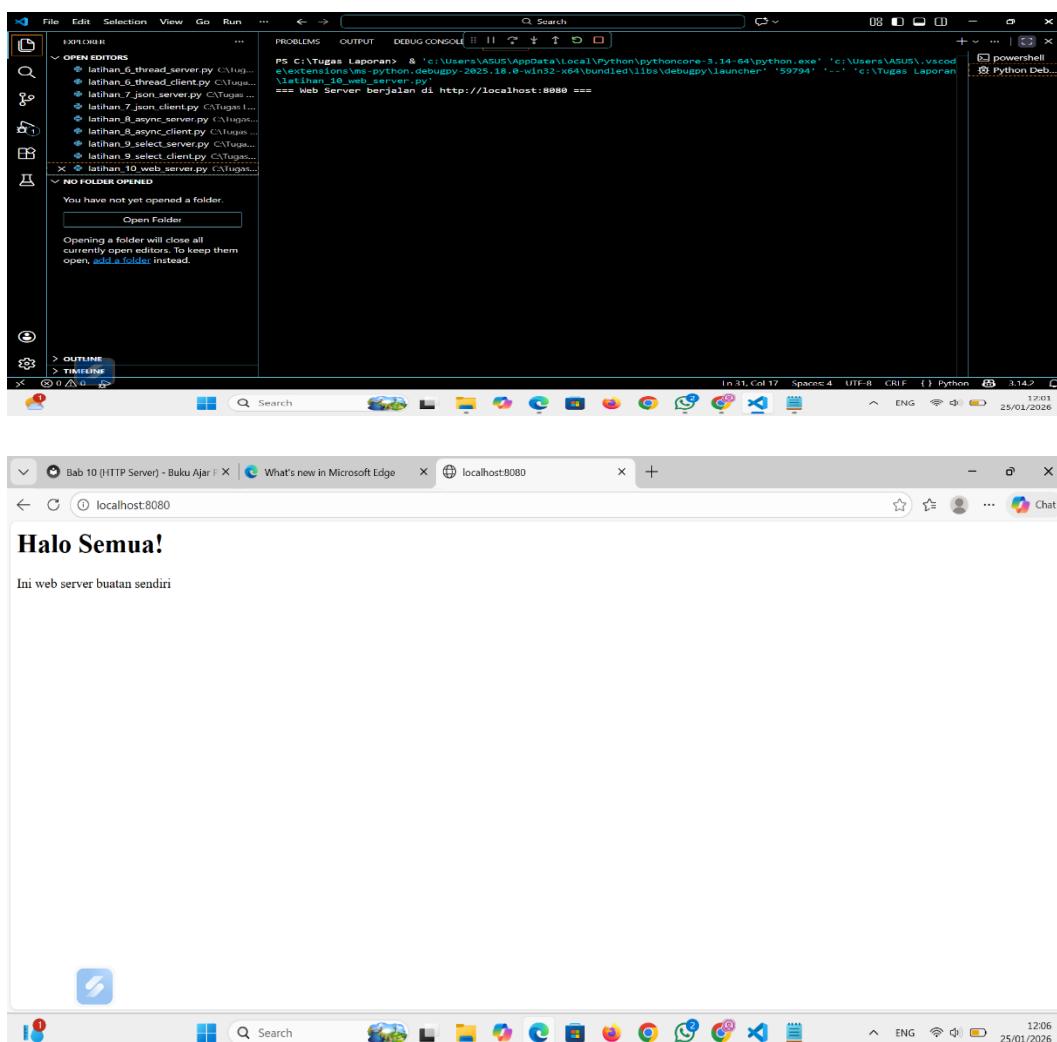


A screenshot of the Visual Studio Code (VS Code) interface, similar to the previous one but showing a failed execution. The terminal window displays the output of the same Python script, but it ends with an error message: "OSError: [WinError 10038] An operation was attempted on something that is not a socket". This indicates a problem with the socket operations, likely related to the 'select()' call. The status bar at the bottom right indicates the date as 25/01/2026 and the time as 10:59.

BAB 10 PROTOKOL HTTP & WEB SERVER

Latihan ini menunjukkan bagaimana membangun web server sederhana dengan Python murni menggunakan socket TCP. Server menerima request HTTP dari browser, mem-parsing request line untuk mendapatkan nama file, lalu mengirimkan response berisi header (status, content-type, content-length) dan isi file. Jika file tidak ada, server mengirim 404 Not Found. Pendekatan ini bersifat single-threaded dan blocking, sehingga mudah dipahami dan cocok untuk belajar konsep HTTP, tetapi tidak cocok untuk produksi karena satu client lambat bisa menahan server. Di industri, server modern menggunakan asyncio, thread pool, atau reverse proxy agar bisa melayani ribuan request secara simultan, sementara logika parsing HTTP tetap sama.

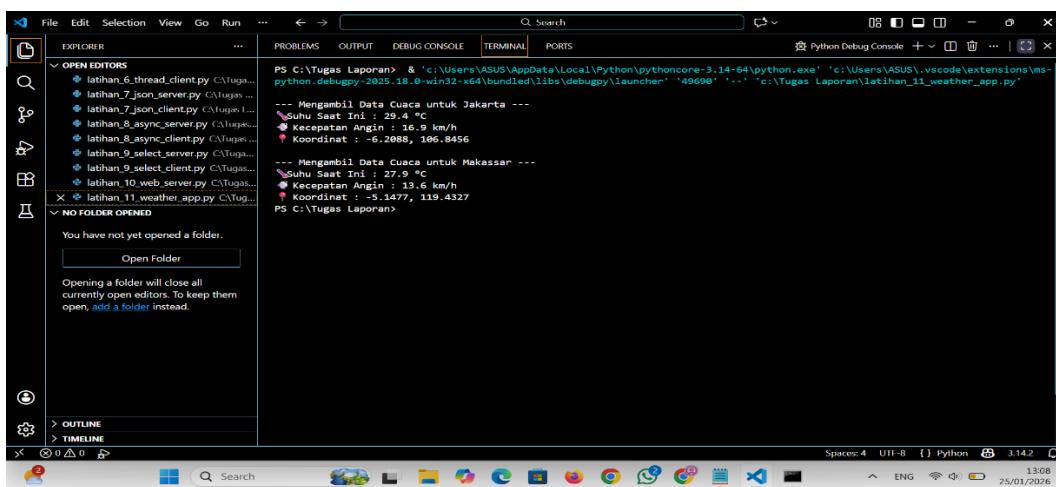
Hasil:



BAB 11 REST API & WEB SERVER

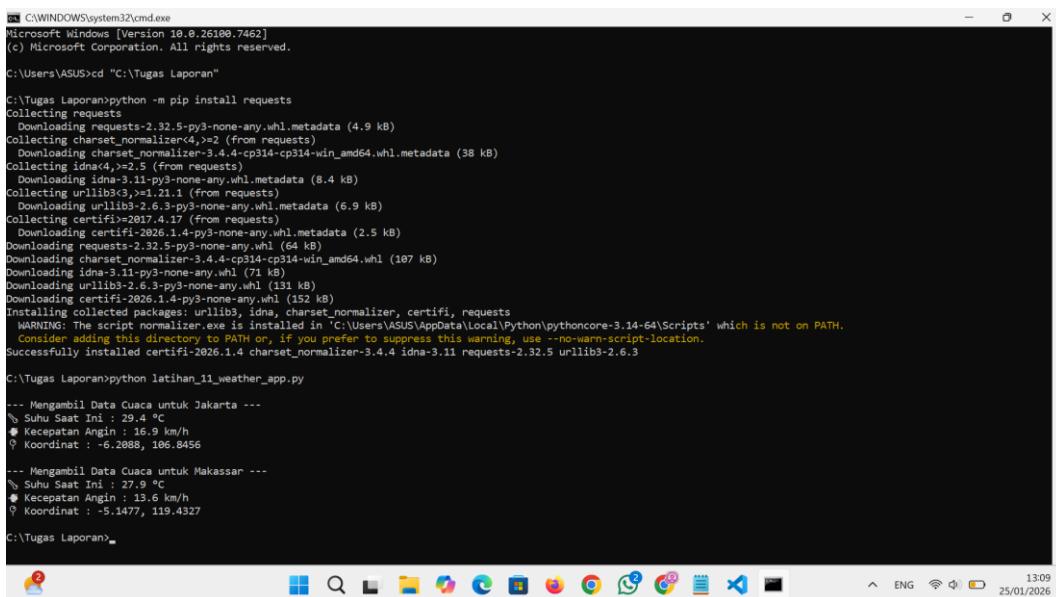
Latihan ini memperlihatkan bagaimana membuat aplikasi console untuk menampilkan cuaca dengan memanfaatkan API publik Open-Meteo. Program mengirim request HTTP GET dengan parameter koordinat ('latitude' dan 'longitude'), menerima respons JSON, lalu menampilkan informasi penting seperti suhu saat ini dan kecepatan angin. Pendekatan ini mengajarkan konsep HTTP request, query string otomatis menggunakan dictionary Python, dan parsing JSON menjadi dictionary, sehingga data API bisa langsung diolah. Latihan ini juga menekankan pentingnya penanganan error seperti koneksi internet gagal atau respons tidak valid, menjadikan aplikasi lebih robust dan user-friendly.

Hasil:



```
P5 C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\vscode\extensions\ms-python.debugpy-2025.16.0-win32-x64\bundled\libs\debugpy\launcher' '49690' '<-->' 'c:\Tugas Laporan\latihan_11_weather_app.py'
--- Mengambil Data Cuaca untuk Jakarta ---
• Suhu Saat Ini : 29.4 °C
• Kecepatan Angin : 16.9 km/h
• Koordinat : -6.2088, 106.8456

--- Mengambil Data Cuaca untuk Makassar ---
• Suhu Saat Ini : 27.9 °C
• Kecepatan Angin : 13.6 km/h
• Koordinat : -5.1477, 119.4327
P5 C:\Tugas Laporan>
```



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.26100.7462]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS>cd "C:\Tugas Laporan"

C:\Tugas Laporan>python -m pip install requests
Collecting requests
  Downloading requests-2.32.5-py3-none-any.whl.metadata (4.9 kB)
Collecting charset_normalizer<4,>=2 (from requests)
  Downloading charset_normalizer-3.4.4-cp314-cp314-win_amd64.whl.metadata (38 kB)
Collecting idna<4,>=2.5 (from requests)
  Downloading idna-3.11-py3-none-any.whl.metadata (8.4 kB)
Collecting urllib3<3,>=1.21.1 (from requests)
  Downloading urllib3-2.6.3-py3-none-any.whl.metadata (6.9 kB)
Collecting certifi<2026.1.4,>=2023.12.7 (from requests)
  Downloading certifi-2026.1.4-py3-none-any.whl.metadata (2.5 kB)
Collecting requests-2.32.5-py3-none-any.whl (64 kB)
  Downloading charset_normalizer-3.4.4-cp314-cp314-win_amd64.whl (107 kB)
  Downloading idna-3.11-py3-none-any.whl (71 kB)
  Downloading urllib3-2.6.3-py3-none-any.whl (131 kB)
  Downloading certifi-2026.1.4-py3-none-any.whl (152 kB)
Installing collected packages: urllib3, idna, charset_normalizer, certifi, requests
WARNING: The script normalize.exe is installed in 'C:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\Scripts' which is not on PATH.
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed certifi-2026.1.4 charset_normalizer-3.4.4 idna-3.11 requests-2.32.5 urllib3-2.6.3

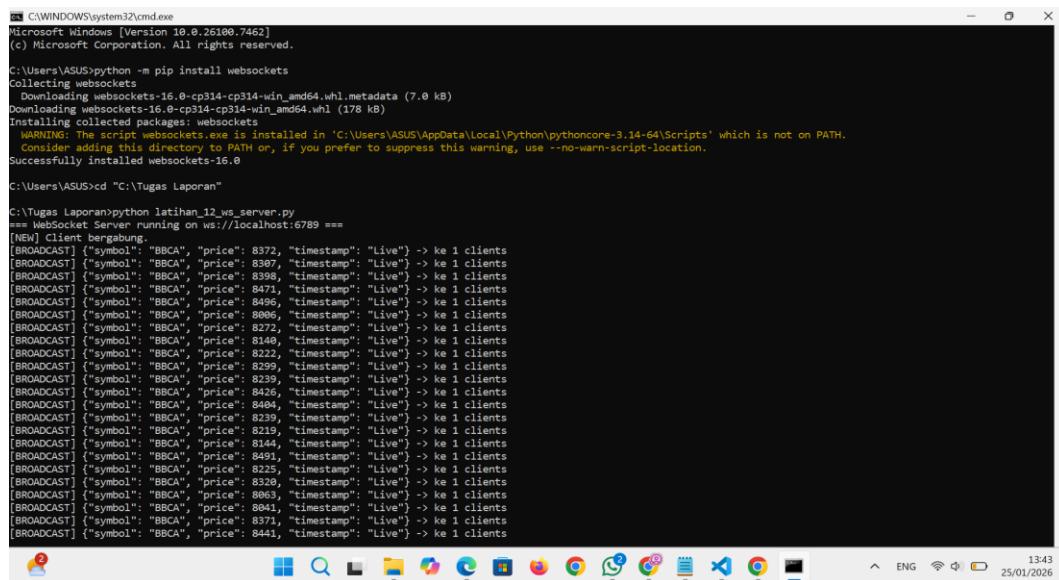
C:\Tugas Laporan>python latihan_11_weather_app.py
--- Mengambil Data Cuaca untuk Jakarta ---
• Suhu Saat Ini : 29.4 °C
• Kecepatan Angin : 16.9 km/h
• Koordinat : -6.2088, 106.8456

--- Mengambil Data Cuaca untuk Makassar ---
• Suhu Saat Ini : 27.9 °C
• Kecepatan Angin : 13.6 km/h
• Koordinat : -5.1477, 119.4327
C:\Tugas Laporan>
```

BAB 12 REAL-TIME COMMUNICATION (WEBSOCKET)

Latihan ini memperlihatkan cara membuat sistem pemantau data real-time menggunakan WebSocket. Server Python mengirimkan harga saham palsu setiap detik ke semua client yang terhubung, sementara browser menerima data tersebut dan langsung menampilkan update secara instan tanpa perlu reload halaman. Konsep penting yang dipelajari termasuk full-duplex communication, penggunaan `zasyncio` untuk menangani banyak client secara efisien, dan parsing JSON di sisi client untuk menampilkan informasi secara dinamis. Pendekatan ini mirip dengan aplikasi dashboard real-time, monitoring IoT, atau aplikasi finansial yang membutuhkan update data terus-menerus dan responsif.

Hasil:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.26100.7462]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS>python -m pip install websockets
Collecting websockets
  Downloading websockets-16.0-cp314-cp314-win_amd64.whl.metadata (7.0 kB)
  Downloading websockets-16.0-cp314-cp314-win_amd64.whl (178 kB)
Installing collected packages: websockets
  WARNING: The script websockets.exe is installed in 'C:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\Scripts' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed websockets-16.0

C:\Users\ASUS>cd "C:\Tugas Laporan"

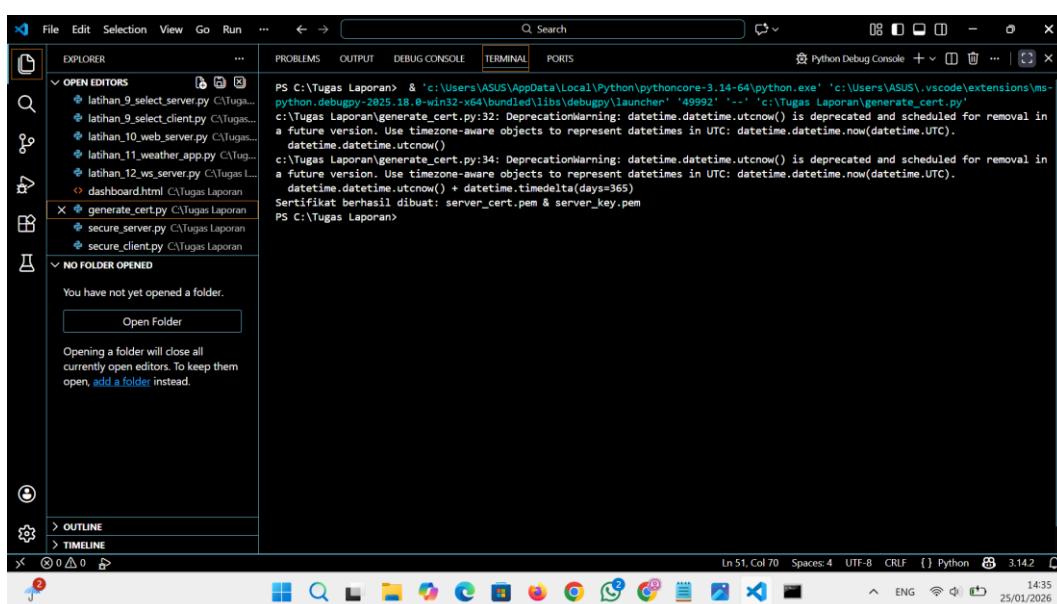
C:\Tugas Laporan\python latihan_12_ws_server.py
*** WebSocket Server running on ws://localhost:6789 ===
[NEW] Client bergabung
[BROADCAST] {"symbol": "BBCA", "price": 8372, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8307, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8398, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8424, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8495, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8006, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8272, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8140, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8222, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8299, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8239, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8426, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8404, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8239, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8191, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8491, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8225, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8320, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8063, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8841, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8371, "timestamp": "Live"} -> ke 1 clients
[BROADCAST] {"symbol": "BBCA", "price": 8441, "timestamp": "Live"} -> ke 1 clients
```



BAB 13 KEAMANAN JARINGAN (NETWORK SECURITY)

Latihan ini memperlihatkan cara membuat server dan client chat aman menggunakan SSL/TLS. Server menggunakan sertifikat self-signed untuk mengenkripsi semua komunikasi, sehingga data yang dikirim dan diterima client terenkripsi otomatis. Client harus menggunakan konteks SSL untuk membungkus socket sebelum connect, sehingga handshake TLS terjadi dan pesan bisa dikirim dengan aman. Pendekatan ini menjadi dasar untuk membangun aplikasi chat atau layanan jaringan aman yang tahan penyadapan, seperti WhatsApp atau aplikasi perbankan.

Hasil:



The screenshot shows the Visual Studio Code interface with the Terminal tab selected. The terminal window displays the following command and its execution:

```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '49992' '--' 'c:\Tugas Laporan\generate_cert.py'  
c:\Tugas Laporan\generate_cert.py:32: DeprecationWarning: datetime.datetime.utcnow() is deprecated and scheduled for removal in  
a future version. Use timezone-aware objects to represent datetimes in UTC: datetime.datetime.now(datetime.UTC).  
    datetime.datetime.utcnow()  
c:\Tugas Laporan\generate_cert.py:34: DeprecationWarning: datetime.datetime.utcnow() is deprecated and scheduled for removal in  
a future version. Use timezone-aware objects to represent datetimes in UTC: datetime.datetime.now(datetime.UTC).  
    datetime.datetime.utcnow() + datetime.timedelta(days=365)  
Sertifikat berhasil dibuat: server_cert.pem & server_key.pem
```

The terminal shows the command used to run the Python script, which generates a self-signed SSL certificate named `server_cert.pem` and `server_key.pem`. The output also includes several deprecation warnings from the Python standard library regarding the use of `datetime.datetime.utcnow()`.

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.26100.7462]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS>cd "C:\Tugas Laporan"

C:\Tugas Laporan>python 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.

C:\Tugas Laporan>
```

```
14:36
25/01/2026

C:\WINDOWS\system32\cmd.exe - python secure_server.py
25/01/2026 10:58      1,202 latihan_9_select_client.py
25/01/2026 10:53      2,867 latihan_9_select_server.py
25/01/2026 14:22      995 secure_client.py
25/01/2026 14:21      1,344 secure_server.py
25 File(s)           31,454 bytes
1 Dir(s)            12,619,657,216 bytes free

C:\Tugas Laporan>python -m pip install cryptography
Requirement already satisfied: cryptography in c:\users\asus\appdata\local\python\pythoncore-3.14-64\lib\site-packages (46.0.3)
Requirement already satisfied: cffi>=2.0.0 in c:\users\asus\appdata\local\python\pythoncore-3.14-64\lib\site-packages (from cryptography) (2.0.0)
Requirement already satisfied: pycparser in c:\users\asus\appdata\local\python\pythoncore-3.14-64\lib\site-packages (from cffi>=2.0.0->cryptography) (3.0)

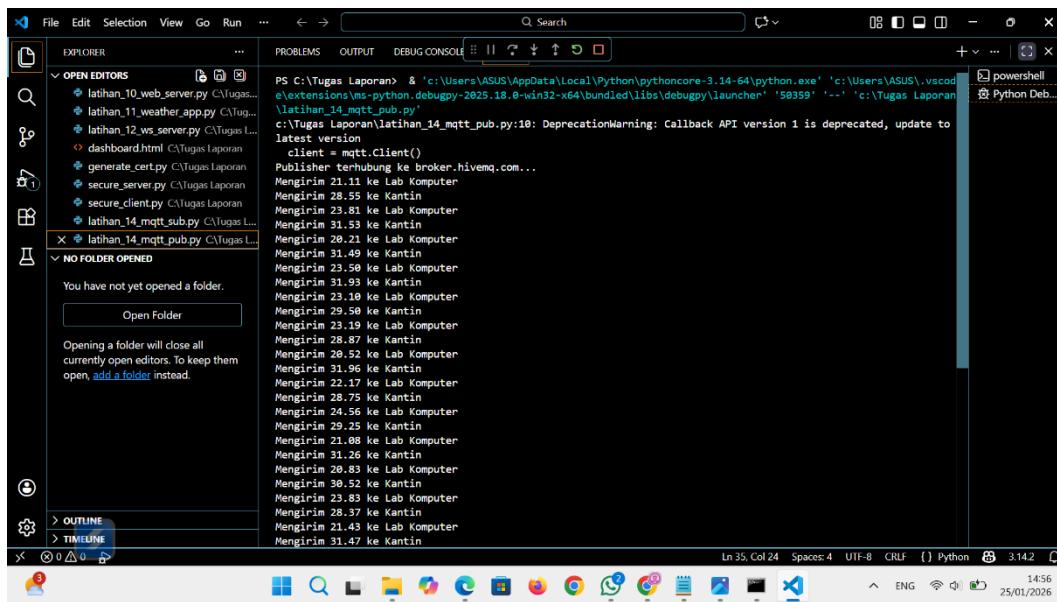
C:\Tugas Laporan>python generate_cert.py
C:\Tugas Laporan>generate_cert.py:32: DeprecationWarning: datetime.datetime.utcnow() is deprecated and scheduled for removal in a future version. Use timezone-aware objects
to represent datetimes in UTC: datetime.datetime.now(datetime.UTC).
  datetime.datetime.utcnow()
C:\Tugas Laporan>generate_cert.py:34: DeprecationWarning: datetime.datetime.utcnow() is deprecated and scheduled for removal in a future version. Use timezone-aware objects
to represent datetimes in UTC: datetime.datetime.now(datetime.UTC).
  datetime.datetime.utcnow() + datetime.timedelta(days=365)
Sertifikat berhasil dibuat: server.cert.pem & server.key.pem

C:\Tugas Laporan>python secure_server.py
Secure Server listening on port 10023...
[NEW] Koneksi TCP dari ('127.0.0.1', 62956)
[SECURE] SSL Handshake sukses dengan ('127.0.0.1', 62956)
Pesan (Decrypted): Halo, ini pesan rahasia CIA.
```

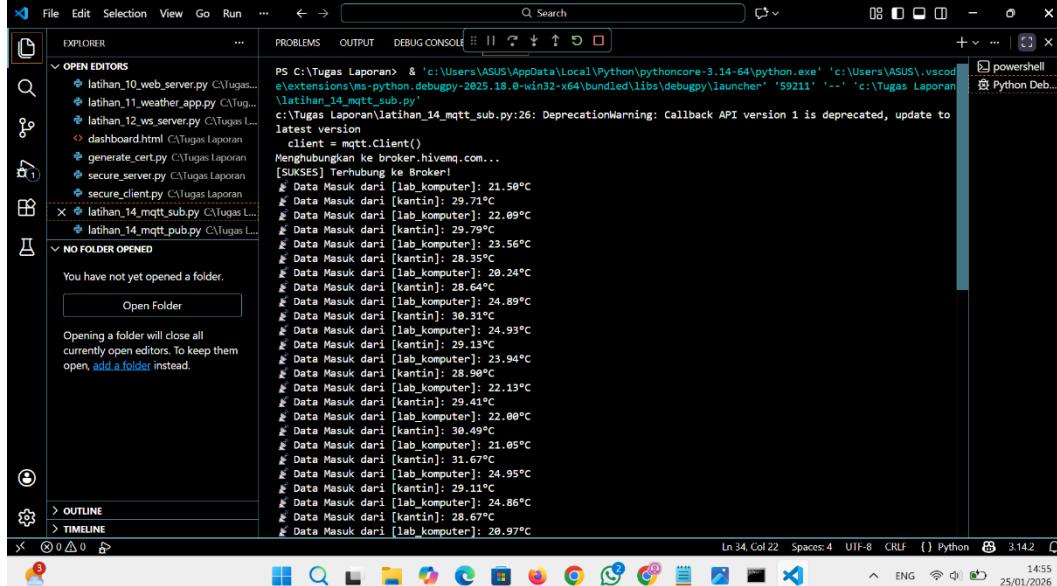
BAB 14 ARSITEKTUR SISTEM TERDISTRIBUSI & IoT (MQTT)

Latihan ini menunjukkan cara membuat sistem rumah cerdas sederhana menggunakan MQTT, di mana sensor mengirim data ke broker publik dan pusat kontrol menerima data secara real-time. Publisher (sensor) mengirimkan suhu ke topik tertentu, sedangkan subscriber (monitor) menggunakan wildcard ('+') untuk menerima data dari semua ruangan sekaligus. Pendekatan ini meniru sistem IoT nyata, memungkinkan monitoring dan kontrol berbagai perangkat secara efisien dan real-time.

Hasil:



```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\vscodeextensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '58359' '--' 'c:\Tugas Laporan\latihan_14_mqtt_pub.py'
c:\Tugas Laporan\latihan_14_mqtt_pub.py:10: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
  client = mqtt.Client()
Publisher terhubung ke broker.hivemq.com...
Mengirim 21.11 ke Lab Komputer
Mengirim 28.55 ke Kantin
Mengirim 23.81 ke Lab Komputer
Mengirim 31.53 ke Kantin
Mengirim 28.21 ke Lab Komputer
Mengirim 31.49 ke Kantin
Mengirim 23.50 ke Lab Komputer
Mengirim 31.93 ke Kantin
Mengirim 23.10 ke Lab Komputer
Mengirim 29.50 ke Kantin
Mengirim 23.19 ke Lab Komputer
Mengirim 28.87 ke Kantin
Mengirim 20.52 ke Lab Komputer
Mengirim 31.96 ke Kantin
Mengirim 22.17 ke Lab Komputer
Mengirim 28.75 ke Kantin
Mengirim 24.56 ke Lab Komputer
Mengirim 29.25 ke Kantin
Mengirim 21.08 ke Lab Komputer
Mengirim 31.26 ke Kantin
Mengirim 20.83 ke Lab Komputer
Mengirim 30.52 ke Kantin
Mengirim 23.83 ke Lab Komputer
Mengirim 28.37 ke Kantin
Mengirim 21.43 ke Lab Komputer
Mengirim 31.47 ke Kantin
```

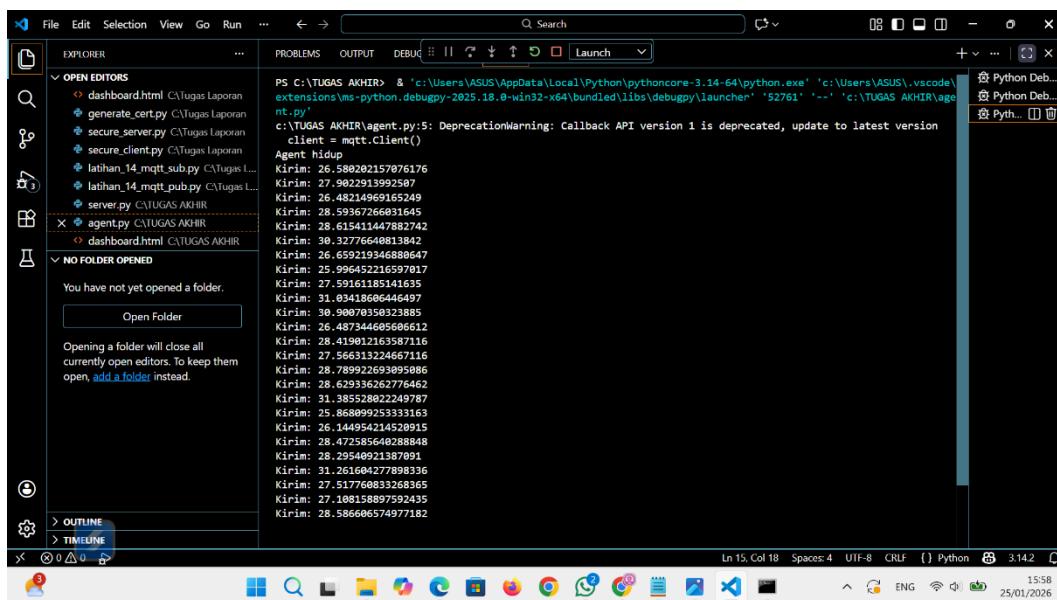


```
PS C:\Tugas Laporan> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\vscodeextensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '59211' '--' 'c:\Tugas Laporan\latihan_14_mqtt_sub.py'
c:\Tugas Laporan\latihan_14_mqtt_sub.py:26: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
  client = mqtt.Client()
Menghubungkan ke Broker!
[SUKSES] Ternhubung ke Broker!
< Data Masuk dari [lab_komputer]: 21.50°C
< Data Masuk dari [kantin]: 29.71°C
< Data Masuk dari [lab_komputer]: 22.09°C
< Data Masuk dari [kantin]: 29.79°C
< Data Masuk dari [lab_komputer]: 23.56°C
< Data Masuk dari [kantin]: 28.35°C
< Data Masuk dari [lab_komputer]: 20.24°C
< Data Masuk dari [kantin]: 28.64°C
< Data Masuk dari [lab_komputer]: 24.89°C
< Data Masuk dari [kantin]: 30.31°C
< Data Masuk dari [lab_komputer]: 24.93°C
< Data Masuk dari [kantin]: 29.13°C
< Data Masuk dari [lab_komputer]: 23.94°C
< Data Masuk dari [kantin]: 28.90°C
< Data Masuk dari [lab_komputer]: 22.13°C
< Data Masuk dari [kantin]: 29.41°C
< Data Masuk dari [lab_komputer]: 22.00°C
< Data Masuk dari [kantin]: 30.49°C
< Data Masuk dari [lab_komputer]: 21.05°C
< Data Masuk dari [kantin]: 31.67°C
< Data Masuk dari [lab_komputer]: 24.95°C
< Data Masuk dari [kantin]: 29.11°C
< Data Masuk dari [lab_komputer]: 24.86°C
< Data Masuk dari [kantin]: 28.67°C
< Data Masuk dari [lab_komputer]: 28.97°C
```

BAB 15 PROYEK AKHIR (CAPSTONE PROJECT)

Bab ini menutup perjalanan pembelajaran dari dasar hingga aplikasi modern, termasuk TCP/UDP, concurrency, JSON/HTTP, WebSocket, SSL, dan MQTT. Mahasiswa diwajibkan membuat Distributed Smart System dengan 3 komponen: Agent (kirim data), Core Server (simpan & API), dan Monitor (dashboard real-time). Fokus penilaian pada integrasi protokol, keandalan, keamanan, arsitektur kode, dan dokumentasi. Bab ini juga memberi peta jalan karier di DevOps, Backend, dan IoT.

Hasil:



The screenshot shows the Visual Studio Code interface. The left sidebar has an 'EXPLORER' view showing files like 'dashboard.html', 'generate_cert.py', 'secure_server.py', 'secure_client.py', 'latihan_14_mqtt_sub.py', 'latihan_14_mqtt_pub.py', 'server.py', and 'agent.py'. The right side has a 'PROBLEMS' tab showing a warning about a deprecated API version. The bottom status bar indicates the file is 'Python' and the version is '3.14.2'.

```
PS C:\TUGAS AKHIR> & "c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe" "c:\Users\ASUS\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher" "52761" '--' 'c:\TUGAS AKHIR\agent.py'
c:\TUGAS AKHIR\agent.py:5: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
  client = mqtt.Client()
Agent: hidup
Kirim: 26.580282157876176
Kirim: 27.9022913992587
Kirim: 26.48214969165249
Kirim: 28.59367266931645
Kirim: 28.615411447882742
Kirim: 30.32776640813842
Kirim: 26.659219346898647
Kirim: 25.996452216597807
Kirim: 27.59161185141635
Kirim: 31.0341866446497
Kirim: 30.96070350523885
Kirim: 26.487344695686612
Kirim: 28.419612163587116
Kirim: 27.566313224667116
Kirim: 28.789922693895086
Kirim: 28.629336262776462
Kirim: 31.38552802249787
Kirim: 25.868899253333163
Kirim: 26.144954214520915
Kirim: 28.472585640288848
Kirim: 28.29540921387091
Kirim: 31.261604277898336
Kirim: 27.517760833268365
Kirim: 27.108158897592435
Kirim: 28.586606574977182
```

```
PS C:\TUGAS AKHIR> & 'c:\Users\ASUS\AppData\Local\Python\pythoncore-3.14-64\python.exe' 'c:\Users\ASUS\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '59481' '--' 'c:\TUGAS AKHIR\server.py'
c:TUGAS AKHIR\server.py:23: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
    mqttc = mqtt.Client()
      * Serving Flask app 'server'
      * Debug mode: off
      * Running on http://127.0.0.1:5000
Press CTRL+C to quit
Diterima: {'suju': '26.68', 'waktu': '2026-01-25 15:57:28.846802'}
Diterima: {'suju': '31.96', 'waktu': '2026-01-25 15:57:21.425982'}
Diterima: {'suju': '29.19', 'waktu': '2026-01-25 15:57:22.852352'}
Diterima: {'suju': '29.28', 'waktu': '2026-01-25 15:57:23.427991'}
Diterima: {'suju': '28.43', 'waktu': '2026-01-25 15:57:24.859915'}
Diterima: {'suju': '28.43', 'waktu': '2026-01-25 15:57:25.431806'}
Diterima: {'suju': '28.97', 'waktu': '2026-01-25 15:57:26.854985'}
Diterima: {'suju': '29.93', 'waktu': '2026-01-25 15:57:27.434830'}
Diterima: {'suju': '31.01', 'waktu': '2026-01-25 15:57:28.859279'}
Diterima: {'suju': '30.99', 'waktu': '2026-01-25 15:57:29.442471'}
Diterima: {'suju': '26.58', 'waktu': '2026-01-25 15:57:29.997883'}
Diterima: {'suju': '27.90', 'waktu': '2026-01-25 15:57:30.001963'}
Diterima: {'suju': '26.48', 'waktu': '2026-01-25 15:57:30.003186'}
Diterima: {'suju': '28.14', 'waktu': '2026-01-25 15:57:30.858783'}
Diterima: {'suju': '28.59', 'waktu': '2026-01-25 15:57:31.003338'}
Diterima: {'suju': '28.44', 'waktu': '2026-01-25 15:57:31.436282'}
Diterima: {'suju': '29.63', 'waktu': '2026-01-25 15:57:32.858406'}
Diterima: {'suju': '28.62', 'waktu': '2026-01-25 15:57:33.002179'}
Diterima: {'suju': '26.69', 'waktu': '2026-01-25 15:57:33.443918'}
Diterima: {'suju': '26.26', 'waktu': '2026-01-25 15:57:34.893112'}
Diterima: {'suju': '30.53', 'waktu': '2026-01-25 15:57:35.000914'}
Diterima: {'suju': '30.83', 'waktu': '2026-01-25 15:57:35.475983'}
```

