

LAPORAN PRAKTIKUM PEMROGRAMAN JARINGAN

“Client – Server (Single Thread)”



Dibuat oleh :

Reyza Syaifullah Sully - 1203220045

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS INFORMATIKA

UNIVERSITAS TELKOM UNIVERSITY

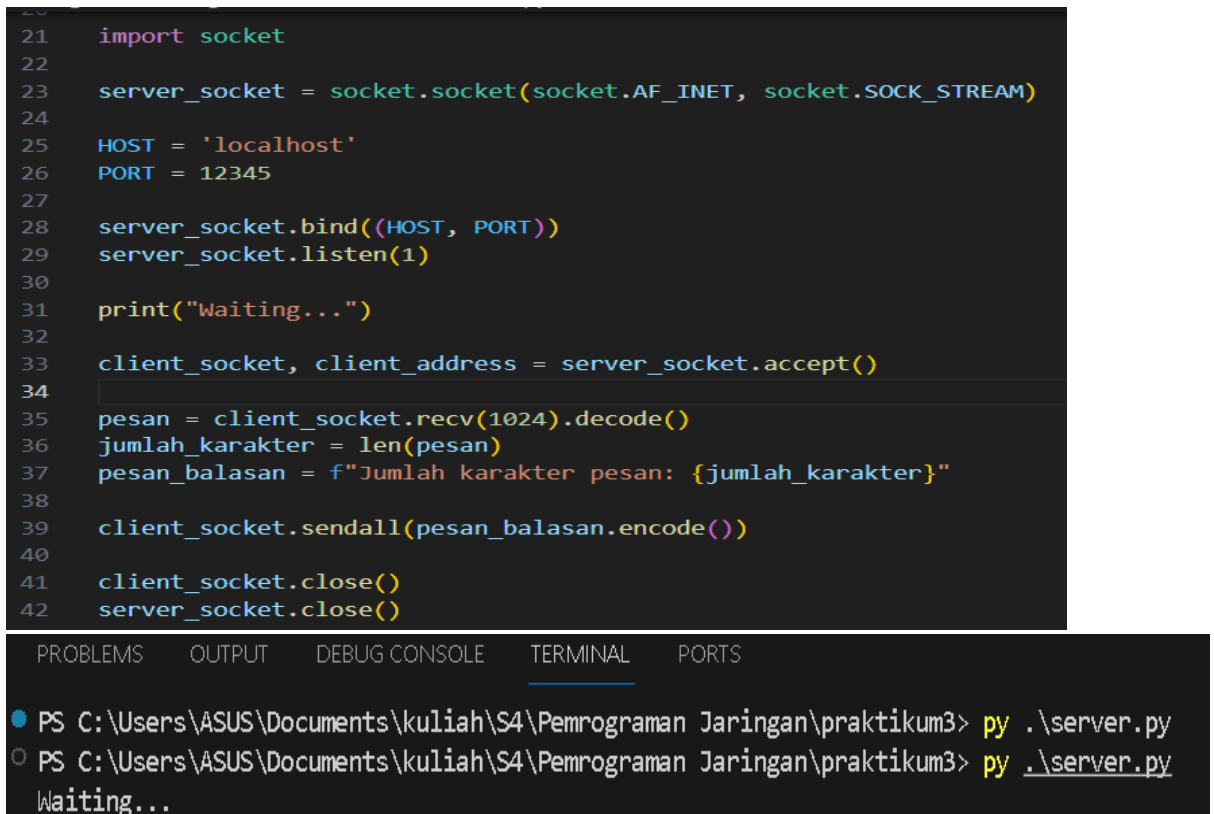
SURABAYA

2024

TUGAS DAN LATIHAN PRAKTIKUM

1. Membuat sebuah program server yang dapat menerima koneksi dari klien menggunakan protokol TCP. Server ini akan menerima pesan dari klien dan mengirimkan pesan balasan berisi jumlah karakter pada pesan tersebut. Gunakan port 12345 untuk server. Membuat analisa dari hasil program tersebut

Screenshot :



```
21 import socket
22
23 server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
24
25 HOST = 'localhost'
26 PORT = 12345
27
28 server_socket.bind((HOST, PORT))
29 server_socket.listen(1)
30
31 print("Waiting...")
32
33 client_socket, client_address = server_socket.accept()
34
35 pesan = client_socket.recv(1024).decode()
36 jumlah_karakter = len(pesan)
37 pesan_balasan = f"Jumlah karakter pesan: {jumlah_karakter}"
38
39 client_socket.sendall(pesan_balasan.encode())
40
41 client_socket.close()
42 server_socket.close()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\ASUS\Documents\kuliah\S4\Pemrograman Jaringan\praktikum3> py .\server.py
PS C:\Users\ASUS\Documents\kuliah\S4\Pemrograman Jaringan\praktikum3> py .\server.py
Waiting...
```

Membuat sebuah server menggunakan modul socket pada bahasa pemrograman Python. Server akan mengikat socket ke alamat IP 'localhost' dengan port '12345' dan kemudian akan mulai mendengarkan koneksi. Saat ada koneksi masuk dari klien, server akan menerima koneksi tersebut dan menerima pesan yang dikirim oleh klien. Pesan yang diterima oleh server berupa data dalam format string yang akan diubah menjadi sebuah bilangan bulat dengan proses dekode.

Setelah menerima pesan dari klien, server akan menghitung jumlah karakter dalam pesan tersebut. Selanjutnya, server akan membuat sebuah pesan balasan yang berisi jumlah karakter dari pesan yang diterima, dan mengirimkannya kembali kepada klien. Setelah mengirim pesan balasan, server akan menutup koneksi dengan klien dan kemudian menutup socket.

- Langkah kedua adalah membuat program klien yang dapat terhubung ke server yang telah dibuat dalam soal nomor 1. Klien ini akan mengirimkan pesan ke server berupa inputan yang diberikan oleh pengguna, dan kemudian menampilkan pesan balasan yang berisi jumlah karakter yang diterima dari server. Selanjutnya, kita akan melakukan analisis terhadap hasil dari program tersebut.

Screenshot :

```
12
13  import socket
14
15  client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
16
17  HOST = 'localhost'
18  PORT = 12345
19
20  client_socket.connect((HOST, PORT))
21  pesan = input("Masukkan pesan: ")
22
23  client_socket.sendall(pesan.encode())
24  pesan_balasan = client_socket.recv(1024).decode()
25
26  print("Pesan balasan dari server:", pesan_balasan)
27
28  client_socket.close()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
● PS C:\Users\ASUS\Documents\kuliah\S4\Pemrograman Jaringan\praktikum3> py .\client.py
Masukkan pesan: Reyza Syaifullah Sully
Pesan balasan dari server: Jumlah karakter pesan: 22
```

Membuat sebuah objek klien socket untuk berkomunikasi dengan server menggunakan modul 'socket'. Klien akan terhubung ke server menggunakan alamat 'localhost' dan port '12345'. Setelah itu, klien akan meminta input dari pengguna berupa pesan yang akan dikirimkan ke server menggunakan metode 'sendall()'.

Ketika klien menerima respons dari server menggunakan metode 'recv()', respons tersebut akan berisi pesan balasan yang mencakup jumlah karakter dari pesan yang dikirim oleh klien. Klien akan menampilkan pesan balasan dari server. Setelah selesai berkomunikasi, klien akan menutup koneksi dengan server.