

## Tugas 2 - FTP Socket Programming On python

1. Buat sebuah program file transfer protocol menggunakan socket programming dengan beberapa perintah dari client seperti berikut

- a. ls : ketika client menginputkan command tersebut, maka server akan memberikan daftar file dan folder
  - b. rm {nama file} : ketika client menginputkan command tersebut, maka server akan menghapus file dengan acuan nama file yang diberikan pada parameter pertama
  - c. Download {nama file} : ketika client menginputkan command tersebut, maka server akan memberikan file dengan acuan nama file yang diberikan pada parameter pertama
  - d. Upload {nama file} : ketika client menginputkan command tersebut, maka server akan menerima dan menyimpan file dengan acuan nama file yang diberikan pada parameter pertama
  - e. Size {nama file} : ketika client menginputkan command tersebut, maka server akan memberikan informasi file dalam satuan MB (Mega bytes) dengan acuan nama file yang diberikan pada parameter pertama
  - f. byebye : ketika client menginputkan command tersebut, maka hubungan socket client akan diputus
  - g. connme : ketika client menginputkan command tersebut, maka hubungan socket client akan terhubung.
2. buat readme.md dengan memberikan Nama dan nim serta penjelasan dan cara menggunakan setiap command yang tersedia

Jawaban:

1. Server:

```
import socket

def hitung_jumlah_karakter(pesan):
    return str(len(pesan))

server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.bind(('localhost', 12345))
server_socket.listen(5)

print("Server berjalan. Menunggu koneksi...")

while True:
    client_socket, client_address = server_socket.accept()
    print("Terhubung dengan", client_address)

    pesan_dari_klien = client_socket.recv(1024).decode()

    if pesan_dari_klien:
        print("Pesan dari klien:", pesan_dari_klien)
```

```
    jumlah_karakter = hitung_jumlah_karakter(pesan_dari_klien)

    client_socket.send(jumlah_karakter.encode())
    print("Jumlah karakter pesan:", jumlah_karakter)

client_socket.close()
```

Client:

```
import socket

def main():
    client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    client_socket.connect(('localhost', 12345))

    print("Koneksi berhasil. Ketik 'connme' untuk masuk ke dalam server.")

    while True:
        command = input("Command: ")

        client_socket.sendall(command.encode())
        response = client_socket.recv(1024).decode()

        if response == "disconnect":
            print("Server memutus koneksi.")
            break
        elif response.startswith("File"):
            print(response)
            continue
        elif response.startswith("Ukuran"):
            print(response)
            continue
        elif response.startswith("Koneksi"):
            print(response)
            continue

        print(response)

    client_socket.close()

if __name__ == "__main__":
    main()
```

## 2. Readme.md

```
# File Transfer Protocol menggunakan Socket Programming

## Nama: [Anggita Rachmadinda Putri]
## NIM: [Nim 1203220086]

### Penjelasan
Program ini adalah implementasi sederhana dari File Transfer Protocol (FTP) menggunakan socket programming dengan Python. FTP ini memungkinkan klien untuk berinteraksi dengan server menggunakan beberapa perintah, seperti:

- ls: Untuk menampilkan daftar file dan folder yang ada di server.
- rm {nama_file}: Untuk menghapus file dengan nama tertentu dari server.
- Download {nama_file}: Untuk mengunduh file dengan nama tertentu dari server.
- Upload {nama_file}: Untuk mengunggah file dengan nama tertentu ke server.
- Size {nama_file}: Untuk menampilkan informasi ukuran file dalam satuan MB dari file tertentu di server.
- byebye: Untuk memutuskan koneksi dengan server.
- connme: Untuk terhubung ke server.

### Cara Menggunakan
1. Jalankan server.py terlebih dahulu
2. Jalankan client.py untuk memulai klien dengan menjalankan perintah
3. Setelah klien terhubung, kita dapat memasukkan perintah-perintah berikut:

- ls: Menampilkan daftar file dan folder di server.
- rm {nama_file}: Menghapus file dengan nama tertentu dari server.
- Download {nama_file}: Mengunduh file dengan nama tertentu dari server.
- Upload {nama_file}: Mengunggah file dengan nama tertentu ke server.
- Size {nama_file}: Menampilkan informasi ukuran file dalam satuan MB dari file tertentu di server.
- byebye: Memutuskan koneksi dengan server.
- connme: Terhubung kembali ke server.
```