

Implementasi program server sesuai contoh

Semua tugas berikut ini harus dijalankan di simulator

1. Jalankan program server seperti dalam pembahasan
2. Buatlah program client yang dapat melakukan 100 request get pada dalam satu saat untuk operasi get file "pokijan.jpg"
3. Capture dan submitlah poin 1 dan 2 dalam satu dokumen pdf. Berikan deskripsi dan penjelasan

Jawab:

1. Mengganti IP Address pada file_server.py menjadi IP Address Alpine1

- Cek IP Address Alpine 1 dengan command **ifconfig**

```
GNS3 console  alpine-1  alpine-2  -  X
/home/work/Pemrograman_Jaringan_D/progjar4a/Jawab # ifconfig
eth0      Link encap:Ethernet  HWaddr 9A:EB:78:9A:4A:34
          inet addr:192.168.122.248  Bcast:192.168.122.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:7689 errors:0 dropped:1 overruns:0 frame:0
          TX packets:4456 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:10868727 (10.3 MiB)  TX bytes:244914 (239.1 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

/home/work/Pemrograman_Jaringan_D/progjar4a/Jawab #
```

- Ganti IP Address pada file_server.py

```
GNS3 console  alpine-1  alpine-2  -  X
GNU nano 4.6  file_server.py
class Server(threading.Thread):
    def __init__(self,ipaddress='192.168.122.248',port=5005):
        self.ipinfo=(ipaddress,port)
        self.the_clients = []
        self.my_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        self.my_socket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
        threading.Thread.__init__(self)

def main():
    svr = Server(ipaddress='192.168.122.248',port=5005)
    svr.start()

if __name__ == "__main__":
    main()

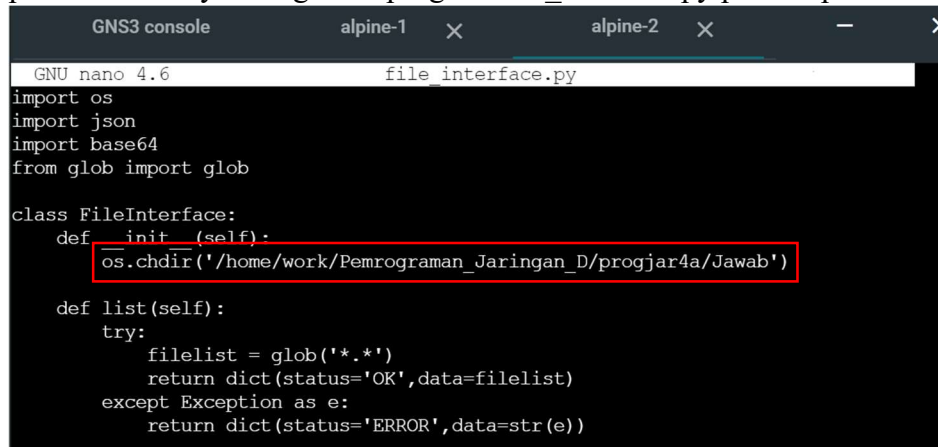
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit      ^R Read File  ^N Replace   ^U Paste Text  ^T To Spell  ^_ Go To Line
```

- Jalankan file_server.py pada Alpine 1 dengan command **python3 file_server.py**

```
GNS3 console  alpine-1  alpine-2  -  X
/home/work/Pemrograman_Jaringan_D/progjar4a/Jawab # python3 file_server.py
WARNING:root:server berjalan di ip address ('192.168.122.248', 5005)
█
```

2. Membuat Program Client 100 Request

- Mengganti path pada `os.chdir('/files')` pada `file_interface.py` menjadi `os.chdir('/home/work/Pemrograman_Jaringan_D/progjar4a/Jawab')` sesuai path dimana saya mengclone program `file_interface.py` pada Alpine2

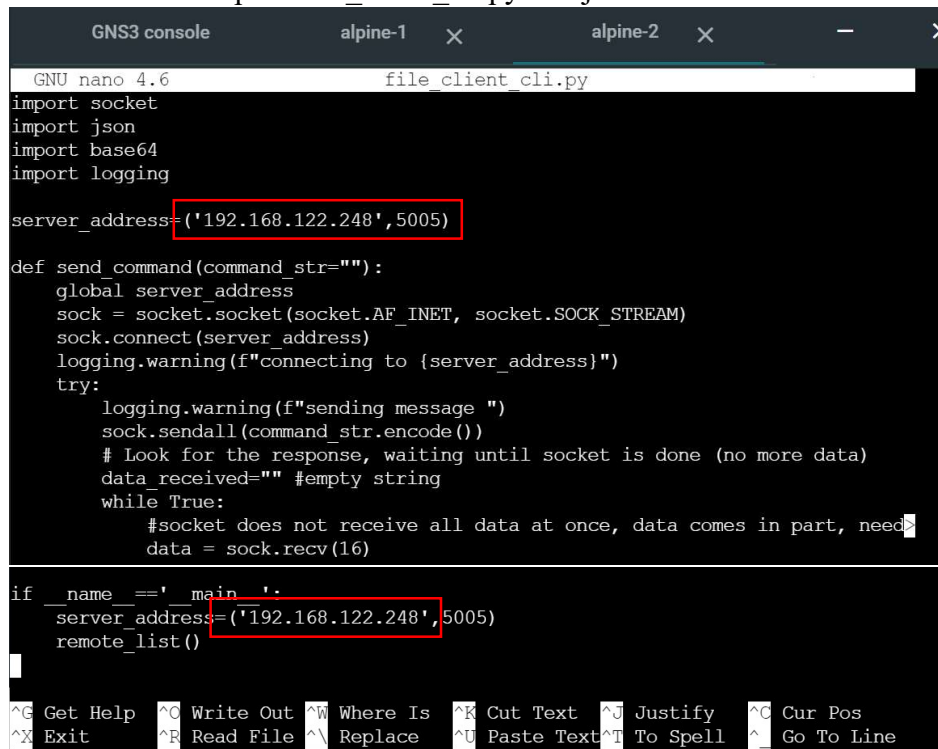


```
GNU nano 4.6 file_interface.py
import os
import json
import base64
from glob import glob

class FileInterface:
    def __init__(self):
        os.chdir('/home/work/Pemrograman_Jaringan_D/progjar4a/Jawab')

    def list(self):
        try:
            filelist = glob('*.*)
            return dict(status='OK',data=filelist)
        except Exception as e:
            return dict(status='ERROR',data=str(e))
```

- Ubah IP Address pada `file_client_cli.py` menjadi IP Address Server



```
GNU nano 4.6 file_client_cli.py
import socket
import json
import base64
import logging

server_address= ('192.168.122.248',5005)

def send_command(command_str=""):
    global server_address
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    sock.connect(server_address)
    logging.warning(f"connecting to {server_address}")
    try:
        logging.warning(f"sending message ")
        sock.sendall(command_str.encode())
        # Look for the response, waiting until socket is done (no more data)
        data_received="" #empty string
        while True:
            #socket does not receive all data at once, data comes in part, need
            data = sock.recv(16)

if __name__ == '__main__':
    server_address= ('192.168.122.248',5005)
    remote_list()

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line
```

- Pada tugas ini, saya menggunakan multithread udp dari tugas sebelumnya yaitu tugas progjar3 untuk client yang melakukan 100 request get (Kode program `udp_multi_thread.py` pada tugas progjar3 dengan sedikit modifikasi). Modifikasi akan dijelaskan pada gambar dibawah.

```
GNS3 console      alpine-1  X      alpine-2  X      -      X
GNU nano 4.6      udp_multi_thread.py
from file_client_cli import remote_get
import time
import datetime
import threading
import socket

#sama seperti kodingan pada folder concurrency, hanya ganti fungsi dan menambah
def kirim_semua():
    texec = dict()
    daftar = 'pokijan.jpg'

    catat_awal = datetime.datetime.now()
    for k in range(100):
        print(f"mengirim {k}")
        #bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi kirim
        texec[k] = threading.Thread(target=remote_get, args=(daftar,))
        texec[k].start()

    #setelah menyelesaikan tugasnya, dikembalikan ke main thread dengan join
    for k in range(100):

    catat_akhir = datetime.datetime.now()
    selesai = catat_akhir - catat_awal
    print(f"Waktu TOTAL yang dibutuhkan {selesai} detik {catat_awal} s/d {catat_akhir}")

#fungsi kirim_semua akan dijalankan secara multithread
if __name__ == '__main__':
    kirim_semua()
```

1

2

3

4

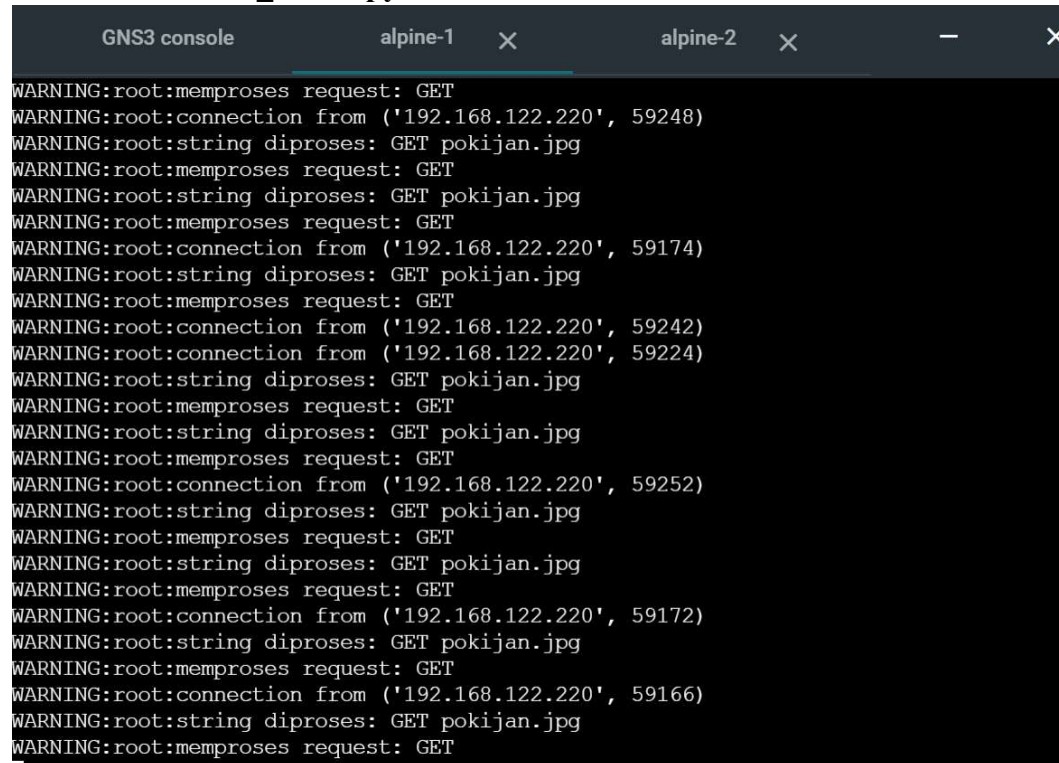
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line

1. Import fungsi `remote_get` pada `file_interface.py`. Fungsi `remote_get` berguna untuk mengirim command dan mengecek apakah command (dalam hal ini filename) yang dikirim berhasil atau tidak. Apabila berhasil, file akan di write dan fungsi mereturn `true`. Sebaliknya apabila tidak berhasil, akan mereturn `false`.

```
def remote_get(filename=""):
    command_str=f"GET {filename}"
    hasil = send_command(command_str)
    if (hasil['status']=='OK'):
        #proses file dalam bentuk base64 ke bentuk bytes
        namafile= hasil['data_namafile']
        isifile = base64.b64decode(hasil['data_file'])
        fp = open(namafile,'wb+')
        fp.write(isifile)
        fp.close()
        return True
    else:
        print("Gagal")
        return False
```

2. Daftar berisikan gambar pokijan.jpg, yaitu gambar yang akan dikirim 100 kali
 3. Looping menggunakan variable k sebanyak 100 kali dengan command `for k in range(100):`
 4. Targetkan fungsi `remote_get` yang diimport dari `file_interface.py` untuk dieksekusi menggunakan thread
- Jalankan program `udp_multi_thread.py` pada Alpine 2 dengan command `python3 udp_multi_thread.py`

Hasil eksekusi `file_server.py`



```
GNS3 console  alpine-1  alpine-2
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.220', 59248)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.220', 59174)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.220', 59242)
WARNING:root:connection from ('192.168.122.220', 59224)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.220', 59252)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.220', 59172)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.220', 59166)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
```

Hasil eksekusi udp_multi_thread.py

```
GNS3 console  alpine-1  alpine-2
WARNING:root:data received from server:
mengirim 89
WARNING:root:data received from server:
mengirim 90
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:connecting to ('192.168.122.248', 5005)
WARNING:root:sending message
WARNING:root:data received from server:
WARNING:root:connecting to ('192.168.122.248', 5005)
WARNING:root:sending message
WARNING:root:data received from server:
mengirim 91
mengirim 92
mengirim 93
mengirim 94
mengirim 95
mengirim 96
mengirim 97
mengirim 98
mengirim 99
WARNING:root:data received from server:

GNS3 console  alpine-1  alpine-2
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:connecting to ('192.168.122.248', 5005)
WARNING:root:sending message
WARNING:root:connecting to ('192.168.122.248', 5005)
WARNING:root:sending message
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:connecting to ('192.168.122.248', 5005)
WARNING:root:sending message
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
Waktu TOTAL yang dibutuhkan 0:00:02.551454 detik 2021-05-31 12:55:11.943845 s/d
2021-05-31 12:55:14.495299
/home/work/Pemrograman_Jaringan_D/progjar4a/Jawab #
```