

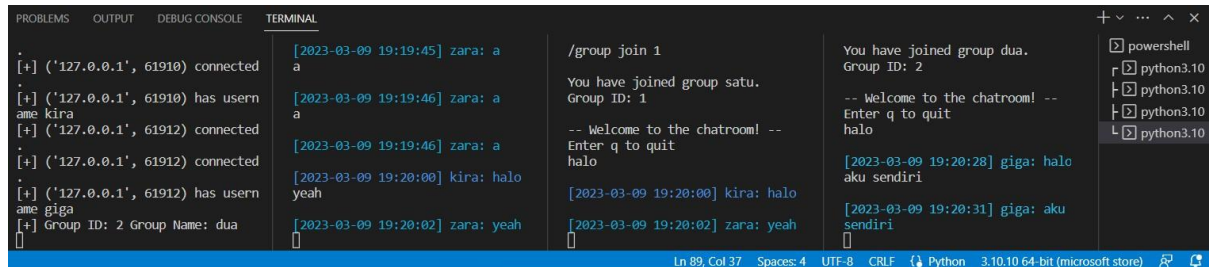
## KELOMPOK 20

Anggota :

1. Zahra Fayyadiyah - 5025201133
2. Khariza Azmi Alfajira Hisyam - 5025201044

## Screenshot eksekusi program

Client dapat membuat grup dan chat di dalam grup yang diinginkan sesuai ID grup



## Penjelasan algoritma Program

```
print("\n-- Welcome to the program, "+name+"! --")
print("Please enter commands below to start:")
print("/list - show all online users with their IP and port")
print("/group create <group name> - create a new group chat")
print("/group list - show all available group")
print("/group join <group ID> - join group based on ID")
print("/private <receiver ID> <message> - to send a private message")
print("\n")
```

Apabila ingin melakukan sesuatu, program dapat menerima command seperti yang tertera di atas

```
class Group:
    def __init__(self):
        self.id = ''
        self.name = ''
        self.member = []
```

Dibuat struktur data bernama Group baik di client ataupun server.

```
def create_group(cs, name):
    global group_id
    g = Group() # kurang args
    g.name = name
    g.id = group_id
    groups.add(g)
    group_id += 1

    str = f"[+] Group ID: {g.id} Group Name: {g.name}"
    print(str)

    to_send = f"A new group named {name} is successfully created."
    cs.client.send(to_send.encode())
```

```
group_id = 1
```

Apabila /group create dijalankan, akan dijalankan fungsi di atas yang menginisiasi instansi Group() baru.

```
def join_group(cs, id):
    for group in groups:
        if group.id == int(id):
            group.member.append(cs)
            str = f"You have joined group {group.name}.~{group.id}"
            cs.client.send(str.encode())
            break
```

Apabila user masuk ke suatu group, server akan mengembalikan string dengan id group yang nanti disimpan oleh client. ID Group akan dikirim kembali oleh klien dalam paket pesan sehingga server dapat mengidentifikasi harus mengirimkan chat tersebut kepada client yang mana saja.

```
elif cmd.startswith("/group join"):
    s.send(cmd.encode())
    message = s.recv(1024).decode()
    tok = message.split('~')
    print("\n" + tok[0])
    print("Group ID: " + str(tok[1]))
    chat(int(tok[1]))
```

```

while True:
    # input message we want to send to the server
    to_send = input()
    # a way to exit the program
    if to_send.lower() == 'q':
        t.stop()
        welcome()
    # add the datetime, name & the color of the sender
    date_now = datetime.now().strftime('%Y-%m-%d %H:%M:%S')
    to_send = f"{group_id}~{client_color}[{date_now}] {name}{separator_token}{to_send}{Fore.RESET}"
    # finally, send the message
    s.send(to_send.encode())

```

Di bawah ini adalah fungsi server ketika klien sudah masuk ke grup dan mengirimkan pesan di dalamnya. Server akan mencari grup yang sesuai dengan ID yang ada di dalam pesan. Apabila ditemukan, akan melakukan iterasi dari semua member yang ada di grup dan memberikan pesan kepada setiap member.

```

else:
    msg = msg.replace(separator_token, ": ")
    # iterate over all connected sockets
    identifier = msg.split('~')
    for group in groups:
        if group.id == int(identifier[0]):
            for client_socket in group.member:
                # and send the message
                client_socket.client.send(identifier[1].encode())

```