

Praktikum 3, Python Socket

Hasil Praktikum anda Kumpul Ke [dropbox](#) :

1. Ketik Script dibawah dan Lakukan testing, usahakan hasilnya tidak error kemudian kumpulkan/upload
2. **TUGAS:**
Jalankan script praktikum 3 anda, kemudian ambil beberapa hasil SCREENSHOT dan simpan dalam bentuk file PDF, setelah itu upload bersama file scriptnya ke dropbox.
3. Kumpulkan atau Upload file praktikum anda [praktikum3_sc_npm.py](#) dan [praktikum3_sc_npm_hasil.PDF](#) ke folder **PJaringan** yang ada di akun [dropbox](#) anda.
4. Harap untuk memperhatikan **Batas Waktu Kumpul Hasil Praktikumnya.**

Server dan Client Sebagai Fungsi

1. Jika praktikum sebelumnya file server dan file client dipisah masing-masing file maka, praktikum 3 kali ini file server dan client dibuat menjadi satu file saja, server dan client diubah menjadi fungsi yang akan dipanggil
2. Saat menjalankan file script, diperlukan **3 buah argument**, yang terdiri dari argument **server/client**, argument kedua **IP address** dan, argument ketiga yaitu **nomor PORT, (Perhatikan CONTOH PERINTAH dibagian bawah)**
3. Pada sisi server hanya akan menerima dan membaca pesan yang dikirim oleh client dan server akan mengirim balik secara otomatis pesan yang dikirim oleh client,
4. Sebaliknya pada sisi client pesan dikirim dengan cara manual yang ditulis lewat keyboard oleh user client.
5. Lakukanlah praktikum dengan dua buah mesin berbeda, bisa kombinasi mesin real dan virtual, atau ajak rekan belajar anda dan gunakan mesin real dari masing-masing laptop anda, yang satu berperan sebagai server, maka dia lebih dahulu menjalankan file scriptnya, setelah itu diikuti dengan menjalankan file script clientnya.

Code Server-Client :

1. Tuliskan code berikut dan simpan dengan nama **praktikum3_sc_npm.py** :

```
1  #!/usr/bin/python
2  import socket, sys
3
4  def server(host, port):
5      s = socket.socket()
6      s.bind((host, port))
7      s.listen(1)
8      print('Menunggu Client....')
9      print('{:<12}: {}'.format('SERVER IP',s.getsockname()[0]))
10     print('{:<12}: {}'.format('SERVER PORT',s.getsockname()[1]))
11     while True:
12         c, addr = s.accept()
13         msg='{}'.format(c.recv(1024).decode())
14         if msg:
15             print ('dari CLIENT: ',msg)
16             c.send('{}'.format(msg).encode())
17         c.close()
18
19
20
21 def client(server_ip, server_port):
22     while True:
23         s = socket.socket()
24         s.connect((server_ip, server_port))
25         pesan=input(">>: ")
26         if pesan == 'q':
27             break
28         s.send(pesan.encode())
29         msg='dari SERVER: {}'.format(s.recv(1024).decode())
30         print (msg)
31         s.close()
32
33
34 # petunjuk :
35 help='''
36 # python3 praktikum3_npm_cs.py [server/client] [IP] [Port]
37 # contoh perintah : sebagai SERVER
38 #     python3 praktikum3_npm_cs.py server 192.168.2.24 3344
39 #     python3 praktikum3_npm_cs.py server - 3344
40
41 # contoh perintah : sebagai CLIENT
42 #     python3 praktikum3_npm_cs.py client 192.168.2.24 3344
43 '''
44
45 # default host and port
46 def_host = '0.0.0.0'
47
48 args = sys.argv[1:]
```

```
49
50 if len(args) == 3:
51     if args[0] == 'server':
52         if args[1] == "-":
53             server(def_host, int(args[2]))
54         if args[1] != "-":
55             server(args[1], int(args[2]))
56
57     if args[0] == 'client':
58         client(args[1], int(args[2]))
59     if args[0] != 'client' and args[0] != 'server':
60         print(help)
61 else:
62     print(help)
```

2. Terdapat total 62 baris code dihitung dengan baris kosong, perhatikan indentasinya (jarak spasi/tab masuk dan keluar)

HOW TO ?

1. Setelah anda selesai menulis Script diatas,
2. Tahap selanjutnya adalah, membuka terminal dan arahkan alamat di terminal ke direktori tempat anda menyimpan scripnya,
3. Script dapat dijalankan dalam satu mesin atau dua mesin, anda dapat menggunakan mesin virtual (virtualbox atau vmware atau container) untuk praktikum atau latihan
4. Copy file script **praktikum3_pjar_npm.py** ke mesin satunya yang akan digunakan sebagai client/server
5. **Di mesin Server:**
 1. Buka Terminal atau CMD, lalu masuk ke direktori tempat anda menyimpan file scripnya
 2. Terdapat contoh perintah dalam script, untuk menampilkan contoh perintah dapat ketikkan perintah tanpa argument :
\$ python3 praktikum3_pjar_npm.py
Hasilnya seperti gambar dibawah :

```

1: ubuntu\":praktikum ▾
ubuntu:praktikum$ python3 praktikum3_sc_npm.py

# python3 praktikum3_cs_npm.py [server/client] [IP] [Port]
# contoh perintah : sebagai SERVER
#     python3 praktikum3_cs_npm.py server 192.168.2.24 3344
#     python3 praktikum3_cs_npm.py server - 3344

# contoh perintah : sebagai CLIENT
#     python3 praktikum3_cs_npm.py client 192.168.2.24 3344

ubuntu:praktikum$ █

```

Gambar 1: Perintah tanpa argument

3. Seperti di contoh perintah yang tampil, jalankan perintah sebagai server terlebih dahulu:

```
$ python3 praktikum3_pjar_npm.py server 192.168.2.24 3344
```

nomor **IP sesuaikan** dengan ip yang ada di mesin anda masing-masing, untuk **PORT (3344)** silahkan ganti dan gunakan **NPM** (lima digit terakhir) anda, sehingga perintahnya menjadi seperti dibawah:

```
$ python3 praktikum3_pjar_npm.py server 10.226.73.109 17000
```

Hasilnya akan tampak seperti gambar 2 dibawah:

```

1: ubuntu\":praktikum ▾
ubuntu:praktikum$ python3 praktikum3_sc_npm.py server 10.226.73.109 17000 # ubah 17000 menjadi npm anda
Menunggu Client....
SERVER IP   : 10.226.73.109
SERVER PORT : 17000
█

```

Gambar 2: Perintah di Terminal pada sisi server

4. **Selesai di sisi Server, sekarang pindah ke sisi Client**

6. Di mesin Client:

1. Buka Terminal atau CMD, lalu masuk ke direktori tempat anda menyimpan file scripnya
2. Perintah tanpa argument, contoh perintah:

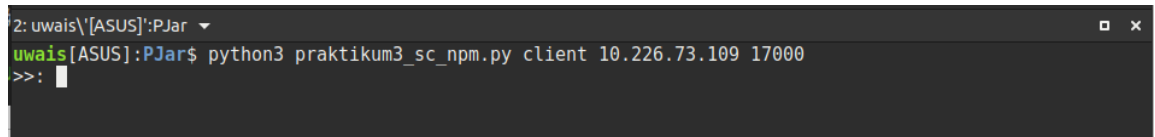
```
$ python3 praktikum3_pjar_npm.py
```

perintah ini akan menghasilkan output sebagaimana yang tampak pada gambar 1 sebelumnya, selanjutnya...

3. Pada sisi client yang perlu diubah hanya argument pertama, yaitu pada argument **server** diubah **menjadi client**, sehingga perintahnya menjadi:

```
$ python3 praktikum3_pjar_npm.py client 10.226.73.109 17000
```

IP diisi dengan ip server tujuan, sedangkan PORT diisi dengan port server tujuan.



```
2: uwais@[ASUS]:PJar  
uwais@[ASUS]:PJar$ python3 praktikum3_sc_npm.py client 10.226.73.109 17000  
>>: 
```

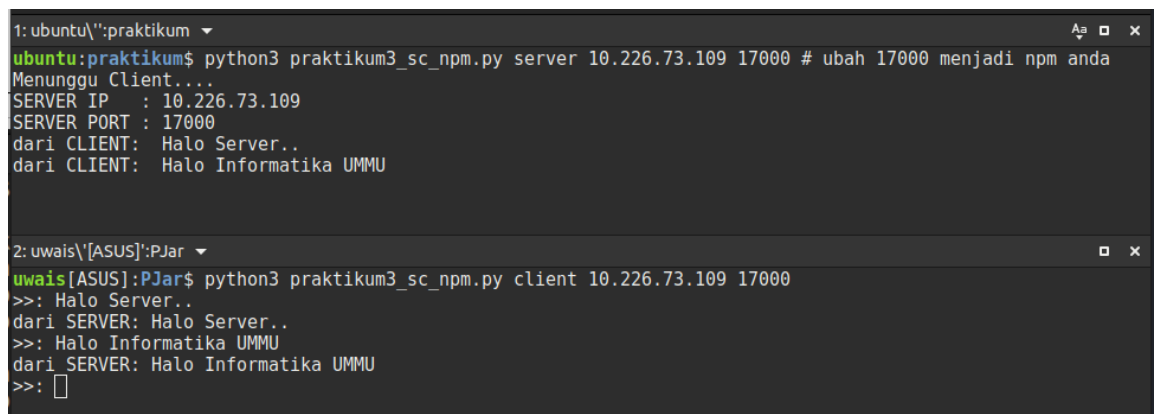
Gambar 3: Sisi Client, siap kirim teks

7. Hasil screenshot Server-Client:

1. Pada gambar 4 dibawah, tampak client mengirim dua pesan:

pesan pertama : **Halo Server..**

pesan kedua : **Halo Informatika UMMU**



```
1: ubuntu\praktikum  
ubuntu:praktikum$ python3 praktikum3_sc_npm.py server 10.226.73.109 17000 # ubah 17000 menjadi npm anda  
Menunggu Client....  
SERVER IP : 10.226.73.109  
SERVER PORT : 17000  
dari CLIENT: Halo Server..  
dari CLIENT: Halo Informatika UMMU  
  
2: uwais@[ASUS]:PJar  
uwais@[ASUS]:PJar$ python3 praktikum3_sc_npm.py client 10.226.73.109 17000  
>>: Halo Server..  
dari SERVER: Halo Server..  
>>: Halo Informatika UMMU  
dari SERVER: Halo Informatika UMMU  
>>: 
```

Gambar 4: Komunikasi Server client

2. Pada server, pesan yang dikirim oleh client secara otomatis akan dikirim kembali ke client, sehingga client akan tau pesannya telah sampai ke server.

selesai