

Program Test 1, OOPL

Due Time: 2: 10 pm, April 6th.

Open Book, but No Internet!

Honesty and Independent Work Demanded! (각자 정직하게 답해 보세요.)

Please do not give me any question! Try as you understood.

(질문 받지 않습니다. 이해한 범위 내에서 하시길 바랍니다.)

Q1: Modify the given class socket example programs for exchanging messages between the server and client program as below:

The two program continues communication until both send “bye”.

Make a use of Scanner library function for making a string message.

Use the **same socket opening**, and do **not use any UTF** functions.

You are supposed to use and modify the given class example, EchoServer.java and EchoClient.java.

The image shows two terminal windows side-by-side, connected by red arrows indicating the flow of communication.

Left Window (EchoServer):

```
EchoServer started.  
Connected to: /127.0.0.1 at port: 8008  
Reading the Message from Client  
client says: Greeting from a client...  
Write the message for client.  
Greeting from Server  
Reading the Message from Client  
client says: Peace to you!  
Write the message for client.  
Good day!  
Reading the Message from Client  
client says: bye  
Write the message for client.  
bye
```

Right Window (EchoClient):

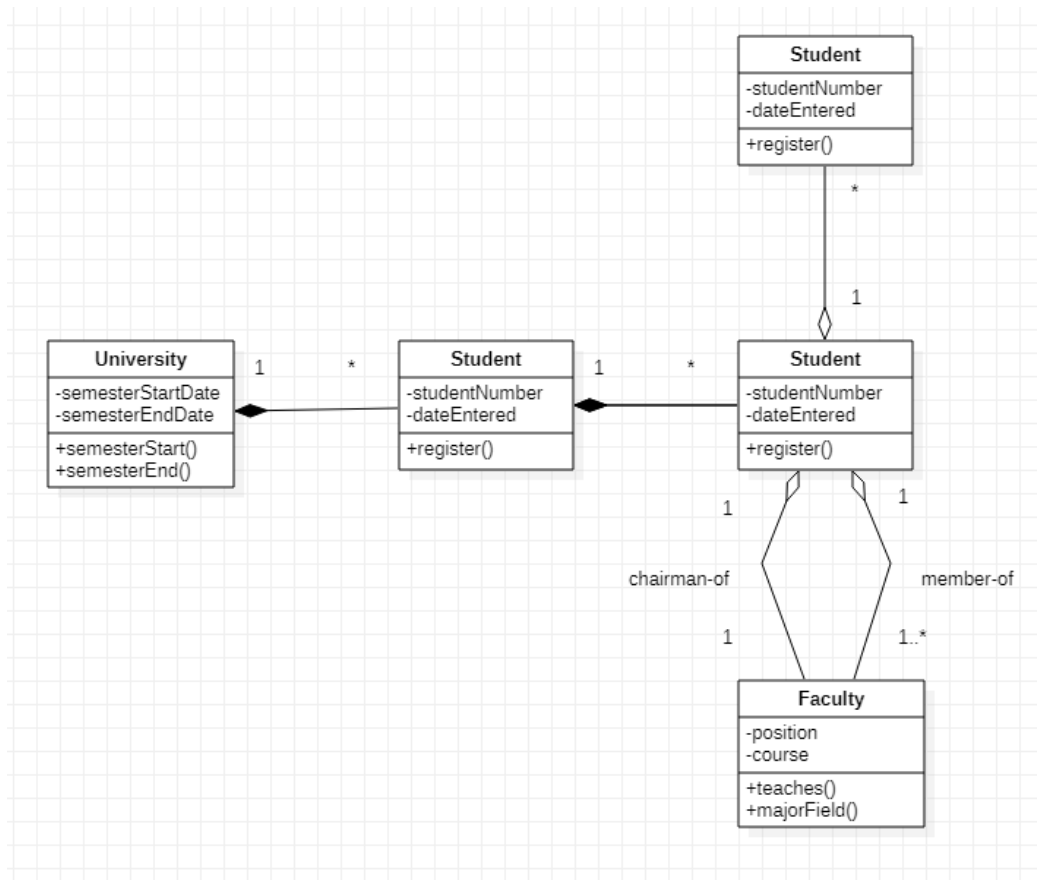
```
Write a Message to Server  
Greeting from a client...  
Reading the Message from Server  
Server says: Greeting from Server  
Write a Message to Server  
Peace to you!  
Reading the Message from Server  
Server says: Good day!  
Write a Message to Server  
bye  
Reading the Message from Server  
Server says: bye
```

Red arrows show the sequence: Client sends 'Greeting from a client...' to Server, Server responds 'Greeting from Server' to Client. Client sends 'Peace to you!' to Server, Server responds 'Good day!' to Client. Client sends 'bye' to Server, Server responds 'bye' to Client.

Submit your zip file that contains ServerSide.java and ClientSide.java under the name, **Your_Name_socketTest**.zip that also contains your starUML file, YourNameUML.mdj.

Submit your result into Hisnet.

Q2: Draw the given UML diagram with your starUML and store the result under the file name, YourNameUML.mdj, and include the mdj file into the file, **Your_Name_socketTest**.zip



Q3: If you have finished Q1, expand Q1 so that one server can communicate two clients with some communication sequence as given below:

```
EchoServer started.  
Connected to: /127.0.0.1 at port: 8008  
Connected to: /127.0.0.1 at port: 8009  
Reading the Message from Client 1  
client says: from c1  
Write the message for client 1.  
to c1  
Reading the Message from Client 2  
client says: from c2  
Write the message for client 2.  
to c2  
Reading the Message from Client 1  
client says: from c1 again  
Write the message for client 1.  
to c1 again  
Reading the Message from Client 2  
client says: from c2 again  
Write the message for client 2.  
to c2 again  
Reading the Message from Client 1  
client says: bye  
Write the message for client 1.  
bye  
Reading the Message from Client 2  
client says: bye  
Write the message for client 2.  
bye
```

The single server communicate client 1 first, then client 2, and client 1, then client 2, and so on until all three send 'bye'.

```
Write a Message to Server  
from c1  
Reading the Message from Server  
Server says: to c1  
Write a Message to Server  
from c1 again  
Reading the Message from Server  
Server says: to c1 again  
Write a Message to Server  
bye  
Reading the Message from Server  
Server says: bye
```

```
Write a Message to Server  
from c2  
Reading the Message from Server  
Server says: to c2  
Write a Message to Server  
from c2 again  
Reading the Message from Server  
Server says: to c2 again  
Write a Message to Server  
bye  
Reading the Message from Server  
Server says: bye
```

Submit your zip file that contains **ServerSide2.java** and **ClientSide2.java**, and **ClientSide3.java** under the same zip file, **Your_Name_socketTest.zip**

