Nama : La Ode Muhammad Gazali
NIM : 222212696
Kelas : 2KS2

MODUL 13 PEMROGRAMAN BERORIENTASI OBJEK

(Networking Bagian 1)

Penugasan

Laporkan hasil praktikum berikut dengan hasil penugasan dan penjelasannya ke Dosen dalam bentuk file pdf dengan format nama <<nim>>_modul13.

1. Lengkapi kode semua objek
2. Tangkapan layar hasil kode yang dilengkapi
3. Tangkapan layar hasil running

Penyelesaian

Buatlah project baru pada Netbeans dengan nama Latihan Server. Pada LatihanServer.java ikutilah kode berikut ini

LatihanServer.java

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this
template
 */
package latihan.server;

/**
 *
 * @author U53R
 */

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.net.ServerSocket;
import java.net.Socket;
import java.util.logging.Level;
import java.util.logging.Logger;
```

```
public class LatihanServer {
    /**
    * @param args the command line arguments
    */
    public static void main(String[] args) {
        // TODO code application logic here
        int portNumber = 4444;
        try (
            ServerSocket serverSocket = new ServerSocket(portNumber);
            Socket clientSocket = serverSocket.accept();
            PrintWriter out = new PrintWriter(clientSocket.getOutputStream(),
true);
            BufferedReader in = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));) {

            System.out.println("Server Receive: " + in.readLine());
            out.println("Pesan Diterima");

        } catch (IOException ex) {
            Logger.getLogger(LatihanServer.class.getName()).log(Level.SEVERE,null
, ex);
        }
    }

}
```

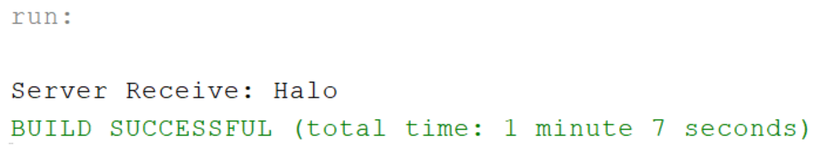
Kemudian buatlah satu kelas baru bernama LatihanClient.java kemudian sesuaikan dengan kode di bawah ini

LatihanClient.java

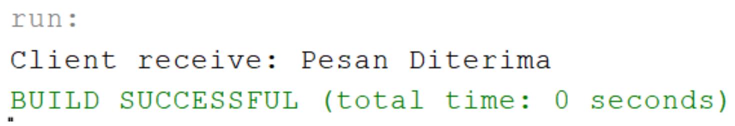
```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this
template
 */
package latihan.server;

/**
 *
 * @author U53R
 */
```

```java
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.net.Socket;
import java.util.logging.Level;
import java.util.logging.Logger;

public class LatihanClient {
    public static void main(String args[]){
        String hostName = "localhost";
        int portNumber = 4444;
        try (
            Socket echoSocket = new Socket(hostName, portNumber);
            PrintWriter out = new PrintWriter(echoSocket.getOutputStream(),true);
            BufferedReader in = new BufferedReader(new
InputStreamReader(echoSocket.getInputStream()));
            BufferedReader stdIn = new BufferedReader(new
InputStreamReader(System.in));){

            out.println("Halo");
            System.out.println("Client receive: "+ in.readLine());

        }catch (IOException ex) {
            Logger.getLogger(LatihanClient.class.getName()).log(Level.SEVERE,null
, ex);
        }
    }
}
```

Maka ketika dijalankan, outputnya sebagai berikut:



```
run:

Server Receive: Halo
BUILD SUCCESSFUL (total time: 1 minute 7 seconds)
```



```
run:
Client receive: Pesan Diterima
BUILD SUCCESSFUL (total time: 0 seconds)
```

Kemudian kita modifikasi sedikit pada LatihanServer.java supaya server tetap menerima pesan hingga client mengetikkan teks tertentu:

LatihanServer.java (Setelah modifikasi)

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this
template
 */
package latihan.server;

/**
 *
 * @author U53R
 */

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.net.ServerSocket;
import java.net.Socket;
import java.util.logging.Level;
import java.util.logging.Logger;

public class LatihanServer {
    /**
    * @param args the command line arguments
    */
    public static void main(String[] args) {
        // TODO code application logic here
        int portNumber = 4444;
        try (
            ServerSocket serverSocket = new ServerSocket(portNumber);
            Socket clientSocket = serverSocket.accept();
            PrintWriter out = new PrintWriter(clientSocket.getOutputStream(),
true);
            BufferedReader in = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));) {

            String msg;
            while(!(msg = in.readLine()).equalsIgnoreCase("exit")){
                System.out.println("Server Receive: " + msg);
                out.println("Pesan Diterima");
            }
        } catch (IOException ex) {
            Logger.getLogger(LatihanServer.class.getName()).log(Level.SEVERE,null
, ex);
        }
    }
}
```

Untuk LatihanClient.java dapat kita modifikasi menjadi berikut ini

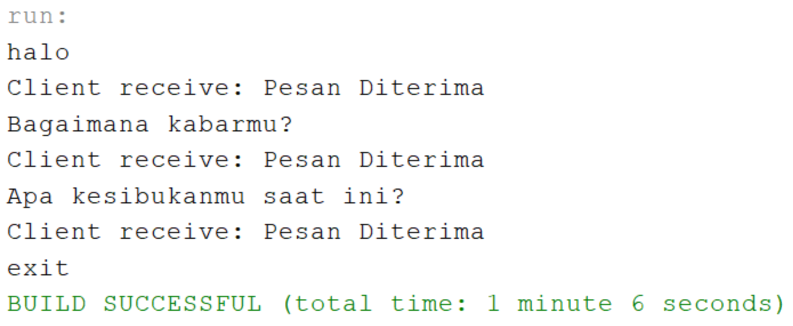
LatihanClient.java (Setelah modifikasi)

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this
template
 */
package latihan.server;

/**
 *
 * @author U53R
 */

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.net.Socket;
import java.util.logging.Level;
import java.util.logging.Logger;

public class LatihanClient {
    public static void main(String args[]){
        String hostName = "localhost";
        int portNumber = 4444;
        try (
            Socket echoSocket = new Socket(hostName, portNumber);
            PrintWriter out = new PrintWriter(echoSocket.getOutputStream(),true);
            BufferedReader in = new BufferedReader(new
InputStreamReader(echoSocket.getInputStream()));
            BufferedReader stdIn = new BufferedReader(new
InputStreamReader(System.in));){
```

```
            String msg;
            while((msg = stdIn.readLine()) != null){
                out.println(msg);
                if(msg.equalsIgnoreCase("exit")) break;
                System.out.println("Client receive: "+ in.readLine());
            }

        }catch (IOException ex) {
            Logger.getLogger(LatihanClient.class.getName()).log(Level.SEVERE,null
, ex);
        }
    }
}
```

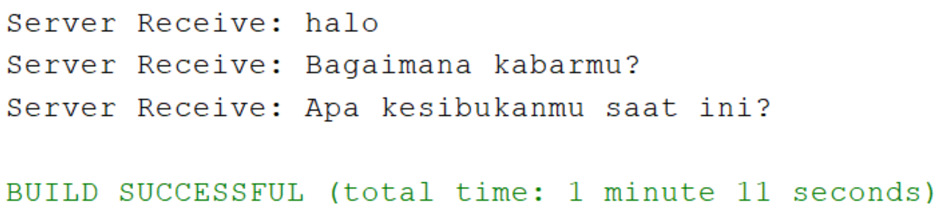
Jalankan LatihanServer.java kemudian jalankan juga LatihanClient.java. Kemudian ketikkan pesan dari Client ke server. Selama client idak menuliskan ”exit” maka client dapat terus mengirimkan pesan apapun ke server

Client mengrim pesan ke server



```
run:
halo
Client receive: Pesan Diterima
Bagaimana kabarmu?
Client receive: Pesan Diterima
Apa kesibukanmu saat ini?
Client receive: Pesan Diterima
exit
BUILD SUCCESSFUL (total time: 1 minute 6 seconds)
```

Server menerima pesan dari clienr



```
Server Receive: halo
Server Receive: Bagaimana kabarmu?
Server Receive: Apa kesibukanmu saat ini?

BUILD SUCCESSFUL (total time: 1 minute 11 seconds)
```