

Praktikum Pemrograman Berbasis Objek

Tanggal Praktikum: Tuesday, 07 June 2022

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

/**
 *
 * @author User
 */
public class ServerThread extends Thread {
    private Socket clientSocket = null;
    public ServerThread(Socket clientSocket) {
        super();
        this.clientSocket = clientSocket;
    }

    @Override
    public void run() {
        try {
            PrintWriter out = new PrintWriter(clientSocket.getOutputStream(), true);
            BufferedReader bis = new BufferedReader(clientSocket.getInputStream());
            ObjectInputStream ois = new ObjectInputStream(bis); {

                Pesan pesan;
                while ((pesan = (Pesan) ois.readObject()) != null) {
                    System.out.println("Server Receive: " +
                        pesan.toString());
                    out.println("Pesan Diterima");
                    if (pesan.getPesan().equalsIgnoreCase("exit"))
                        break;
                }
            } catch (IOException ex) {

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

                Logger.getLogger(ServerThread.class.getName()).log(Level.SEVERE,
                    null, ex);
            } finally {
                if (clientSocket != null) {
                    try {
                        clientSocket.close();
                    } catch (IOException ex) {

                        Logger.getLogger(ServerThread.class.getName()).log(Level.SEVERE,
                            null, ex);
                    }
                }
            }
        }
    }
}
```

```

import java.io.Serializable;

/**
 * @author User
 */
public class Pesan implements Serializable {
    private String nama;
    private String pesan;
    public Pesan(String nama, String pesan) {
        this.nama = nama;
        this.pesan = pesan;
    }

    @Override
    public String toString() {
        return "[" + nama + "]" + " " + pesan;
    }
    /**
     * @return the nama
     */
    public String getNama() {
        return nama;
    }
    /**
     * @param nama the nama to set
     */
    public void setNama(String nama) {
        this.nama = nama;
    }
    /**
     * @return the pesan
     */
    public String getPesan() {
        return pesan;
    }
    /**
     * @param pesan the pesan to set
     */
    public void setPesan(String pesan) {
        this.pesan = pesan;
    }
}

```

The screenshot shows an IDE window for a project named 'praktikum14'. The main editor displays the 'LathanClient.java' file, which contains the following code:

```

30      BufferedReader in = new BufferedReader(new
31          InputStreamReader(echoSocket.getInputStream()));
32      BufferedReader stdIn = new BufferedReader(new
33          InputStreamReader(System.in));
34
35      String msg;
36      while((msg = stdIn.readLine()) != null){
37          oos.writeObject(new Pesan( nama: "Polis", msg));
38          oos.flush();

```

The left sidebar shows the project structure with files like 'LathanClient', 'LathanServer', 'Pesan', and 'ServerThread'. The bottom panel shows the 'Run' output, indicating that the server is receiving messages from the client:

```

Server Receive: [Indi] Halo
Server Receive: [Ana] Apa Kabs
Server Receive: [Polis] Y

```