

PEMROGRAMAN JARINGAN KOMPUTER

TUGAS 13

DOSEN PENGAJAR : NOPRIANTO, S. Kom., M.Eng.



OLEH:

NABILAH ROISUL AMINI (1731710116)

MI – 3A

D3 MANAJEMEN INFORMATIKA

TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2019

TUGAS

1. Modifikasi program di atas sehingga tidak hanya mengirimkan data text, tetapi juga bisa mengirimkan sebuah file.

Jawaban :

Pada praktikum ini file yang dikirim adalah “tugas.txt”

AH (D:) > KULIAH > SEMESTER 5 > PEMROGRAMAN JARINGAN > PERTEMUAN 13 > New folder > pemrograman-jaringan-tugas13				
Name	Date modified	Type	Size	
build	11/25/2019 2:43 PM	File folder		
nbproject	11/25/2019 12:28 PM	File folder		
src	11/25/2019 2:43 PM	File folder		
test	11/25/2019 12:32 PM	File folder		
.gitignore	11/25/2019 12:27 PM	Text Document	1 KB	
build	11/25/2019 12:27 PM	XML Document	4 KB	
manifest.mf	11/25/2019 12:27 PM	MF File	1 KB	
README	11/25/2019 12:27 PM	Markdown Source...	1 KB	
tugas	11/25/2019 12:46 PM	Text Document	1 KB	

Code Tugas1

Server

```
6 package Tugas1;
7 import java.io.BufferedReader;
8 import java.io.File;
9 import java.io.FileReader;
10 import java.io.InputStreamReader;
11 import java.net.DatagramPacket;
12 import java.net.DatagramSocket;
13 import java.net.InetAddress;
14 import java.util.Scanner;
15 /**
16  *
17  * @author asus
18  */
19 public class Server {
20     public static DatagramSocket ds;
21     public static int clientport = 800, serverport = 900;
22
23     public static void main(String args[]) throws Exception{
24         byte buffer[] = new byte[1024];
25         ds = new DatagramSocket(serverport);
26
27         File f = new File("tugas.txt");
28         BufferedReader dis = new BufferedReader(new FileReader(f));
29         System.out.println("Server menunggu input");
30         InetAddress i = InetAddress.getByName("localhost");
```

```

31         while(true){
32             System.out.print("Server Mengirim file " +f+"\n");
33             String str = dis.readLine();
34             if((str==null || str.equals("end"))){
35                 break;
36             }
37             buffer = str.getBytes();
38             ds.send(new DatagramPacket(buffer,str.length(),i,clientport));
39         }
40     }
41 }

```

Client

```

6     package Tugas1;
7
8     import java.net.DatagramPacket;
9     import java.net.DatagramSocket;
10
11     /**
12      * @author asus
13      */
14     public class Client {
15         public static DatagramSocket d;
16         public static byte buffer[] = new byte[1024];
17         public static int clientport = 800, serverport = 900;
18
19         public static void main(String[] args) throws Exception {
20             d = new DatagramSocket(clientport);
21             System.out.println("Client sedang menunggu server mengirimkan data");
22             System.out.println("tekan Ctrl+C untuk mengakhiri");
23             while(true){
24                 DatagramPacket p = new DatagramPacket(buffer,buffer.length);
25                 d.receive(p);
26                 String ps = new String(p.getData(),0,p.getLength());
27                 System.out.println("From Server : " + ps);
28             }
29         }
30     }

```

Output


```

Output x
pemrograman-jaringan-tugas13 (run-single) x pemrograman-jaringan-tugas13 (run-single) #2 x
ant -f "D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13" -Djavac.includes=Tugas1/Server.java -Dnb.inte
init:
Deleting: D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\build-jar.properties
deps-jar:
Updating property file: D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\build-jar.properties
Compiling 1 source file to D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\classes
compile-single:
run-single:
Server menunggu input
Server Mengirim file tugas.txt
Server Mengirim file tugas.txt
Server Mengirim file tugas.txt
Server Mengirim file tugas.txt
Server Mengirim file tugas.txt
BUILD SUCCESSFUL (total time: 1 second)

```

```
Output X
pemrograman-jaringan-tugas13 (run-single) X pemrograman-jaringan-tugas13 (run-single) #2 X
ant -f "D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13" -Djavac.includes=Tugas1/Client.java -Dnb.inte
init:
Deleting: D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\build-jar.properties
deps-jar:
Updating property file: D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\build-jar.properties
Compiling 1 source file to D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\classes
compile-single:
run-single:
Client sedang menunggu server mengirimkan data
tekan Ctrl+C untuk mengakhiri
From Server : Nabilah Roisul Amini
From Server : 1731710116
From Server : Manajemen Informatika - 3A
From Server : Politeknik Negeri Malang
```

File tugas.txt

 tugas.txt - Notepad

File Edit Format View Help

Nabilah Roisul Amini
1731710116
Manajemen Informatika - 3A
Politeknik Negeri Malang

2. Modifikasilah program diatas agar menjadi program yang dapat digunakan untuk saling mengirim pesan menggunakan 2 komputer.

Jawaban :

Code Tugas2

Server

```
6 package Tugas2;
7
8 import java.io.BufferedReader;
9 import java.io.IOException;
10 import java.io.InputStreamReader;
11 import java.io.PrintStream;
12 import java.net.ServerSocket;
13 import java.net.Socket;
14
15 /**
16  *
17  * @author asus
18  */
19 public class Server {
20
21     public static void main(String[] args) throws IOException {
22         ServerSocket sk = new ServerSocket(8888);
23         Socket ss = sk.accept();
24         BufferedReader in = new BufferedReader(new InputStreamReader(ss.getInputStream()));
25         PrintStream out = new PrintStream(ss.getOutputStream());
26         BufferedReader cin = new BufferedReader(new InputStreamReader(System.in));
27         String inputan;
```

```

28         while (true) {
29             inputan = in.readLine();
30             if (inputan.equalsIgnoreCase("END")) {
31                 out.println("BYE");
32                 break;
33             }
34             System.out.print("Client : " + inputan + "\n");
35             System.out.print("Server : ");
36             inputan = cin.readLine();
37             out.println(inputan);
38         }
39         sk.close();
40         ss.close();
41         in.close();
42         out.close();
43         cin.close();
44     }
45 }

```

Client

```

6  package Tugas2;
7  import java.io.BufferedReader;
8  import java.io.InputStreamReader;
9  import java.io.PrintStream;
10 import java.net.Socket;
11 import java.util.Scanner;
12 /**
13  *
14  * @author asus
15  */
16 public class Client {
17     public static void main(String[] args) throws Exception{
18         try {
19             Scanner sc = new Scanner(System.in);
20             System.out.print("Masukkan IP Address : ");
21             String ip = sc.nextLine();
22             System.out.print("Masukkan Socket Server : ");
23             int socket = sc.nextInt();
24             Socket sk = new Socket(ip, socket);
25             BufferedReader sin = new BufferedReader(new InputStreamReader(sk.getInputStream()));
26             PrintStream sout = new PrintStream(sk.getOutputStream());
27             BufferedReader stdin = new BufferedReader(new InputStreamReader(System.in));
28             String s;
29
30             while (true){
31                 System.out.print("Client : ");
32                 s = stdin.readLine();
33                 sout.println(s);
34                 s = sin.readLine();
35                 System.out.print("Server : "+s+"\n");
36                 if(s.equalsIgnoreCase("BYE")){
37                     break;
38                 }
39             }
40             sk.close();
41             sin.close();
42             sout.close();
43             stdin.close();
44         } catch (Exception e){
45             System.out.println("Unable to connect to server \nMaybe your ip or socket is not correct");
46         }
47     }

```

Output

Jalankan class Server dan class Client. Pada class Client masukkan ip address dari server. Kemudian masukkan pula socket server (8888).

Client

```
Output X
pemrograman-jaringan-tugas13 (run-single) X pemrograman-jaringan-tugas13 (run-single) #2 X
ant -f "D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13" -Djavac.includes=Tugas2\Client.java -Dnb.int
init:
Deleting: D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\build-jar.properties
deps-jar:
Updating property file: D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\build-jar.properties
Compiling 1 source file to D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\classes
compile-single:
run-single:
Masukkan IP Address : 192.168.1.7
Masukkan Socket Server : 8888
Client : Good Night, Server
Server : Good Night, Client
Client : How are you ?
Server : I'm fine, thank you. And you?
Client : I'm fine too. Thank you
```

Server

```
Output X
pemrograman-jaringan-tugas13 (run-single) X pemrograman-jaringan-tugas13 (run-single) #2 X
ant -f "D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13" -Djavac.includes=Tugas2/Server.java -Dnb.inte
init:
Deleting: D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\build-jar.properties
deps-jar:
Updating property file: D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\build-jar.properties
Compiling 1 source file to D:\KULIAH\SEMESTER 5\PEMROGRAMAN JARINGAN\PERTEMUAN 13\New folder\pemrograman-jaringan-tugas13\build\classes
compile-single:
run-single:
Client : Good Night, Server
Server : Good Night, Client
Client : How are you ?
Server : I'm fine, thank you. And you?
Client : I'm fine too. Thank you
Server :
```

3. Buatlah program untuk meremote Cursor Mouse computer lain menggunakan UDP!

Jawaban :

Code Tugas3

Class Robots

```
6 package Tugas3;
7 import java.awt.AWTException;
8 import java.awt.Robot;
9 /**
10  *
11  * @author asus
12  */
13 public class Robots {
14     public static void main(String[] args) {
15         try{
16             Robot robo = new Robot();
17         } catch (AWTException ex){
18
19         }
20     }
21
22 }
```

Class Canvas

```
6 package Tugas3;
7 import java.awt.Dimension;
8 import java.awt.Graphics;
9 import java.awt.image.BufferedImage;
10 import javax.swing.JPanel;
11 /**
12  *
13  * @author asus
14  */
15 public class Canvas extends JPanel{
16     BufferedImage capture;
17
18     public Canvas(BufferedImage capture){
19         this.capture = capture;
20         this.setPreferredSize(new Dimension(capture.getWidth(), capture.getHeight()));
21     }
22
23     @Override
24     protected void paintComponent(Graphics g){
25         super.paintComponent(g);
26         g.drawImage(capture, 0, 0, null);
27         g.dispose();
28     }
29 }
```

Main Class

```
12 package Tugas3;
13 import java.awt.AWTException;
14 import java.awt.Rectangle;
15 import java.awt.Robot;
16 import java.awt.image.BufferedImage;
17 import javax.swing.JFrame;
18 /**
19  *
20  * @author od3ng
21  */
22 public class PemrogramanJaringanTugas13 {
23
24     /**
25      * @param args the command line arguments
26      */
27     public static void main(String[] args) {
28         BufferedImage capture = null;
29         int width = 400;
30         int height = 400;
31         try{
32             Robot robo = new Robot();
33             //delay untuk mengatur layar
34             robo.delay(1000);
```

```

36         capture = robo.createScreenCapture(new Rectangle(100,100,width,height));
37         if(capture != null){
38             Canvas panel = new Canvas(capture);
39             JFrame frame = new JFrame("Test Capture");
40             frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
41             frame.add(panel);
42             frame.pack();
43             frame.setLocationRelativeTo(null);
44             frame.setVisible(true);
45         }
46     } catch (AWTException ex) {
47
48     }
49
50 }
51
52 }

```

Output

The screenshot shows an IDE window titled "Test Capture". The left sidebar displays a project structure with packages and files. The main editor shows the source code of a Java program. The code defines a `main` method that creates a `Robot` object, delays execution for 1000ms, and then attempts to create a screen capture. The output window at the bottom shows the execution of an Ant build script, which includes steps for deleting old files, updating property files, compiling source files, and running the program.

Source Code:

```

26  /*
27  */
28
29  public static void main()
30  {
31      BufferedImage capture;
32      int width = 400;
33      int height = 400;
34      try{
35          Robot robo = new Robot();
36          //delay untuk me
37          robo.delay(1000)
38
39          capture = robo.c
40          if(capture != nu
41          Canvas panel

```

Output - pemrograman-jaringan-tugas13 (run-singl

```

ant -f "D:\\KULIAH\\SEMESTER 5\\PEM
init:
Deleting: D:\\KULIAH\\SEMESTER 5\\PEM
deps-jar:
Updating property file: D:\\KULIAH\\
Compiling 1 source file to D:\\KULI
compile-single:

```