NP Summary

Chapter (1)

Network questions:

- * Most common network card? Ethernet Card

- * Physical Layer devices? Repeaters, Hubs] → Single Collision domain.
- * Data link layer device? Switch
- * Network layer device? Router
- * Data link layer protocols: → LLC (Flow & error Control)
  (point to point layer). → MAC (media access Control) _ MAC Addressing

- * Network layer protocol: Routing algorithms → Routing table.

- * Transport Layer protocols: TCP (ch:6), UDP (ch:5)
  (end to end layer)

- * OSI model ?! (7-layers) → What?!
  TCP/IP model ?! (5-layers) → what?!

- * Information Unit (bits → frame → packet → Segment → msg)
  (physical, data link, Network, Transport, App.)

protocol that Convert from mac to IP Address? ARP.
or same meaning word.

Chapter 2

* AWT → Components & containers, Layout manager, Event-handling, Graphics.

-Building GUI:

① Frame or Applet
↳ application, but have GUI like applet.

- by extend applet or extend frame in main method.

② Components

- Creating Objects for (buttons, checkbox, panel, ---)

③ Arrange Components

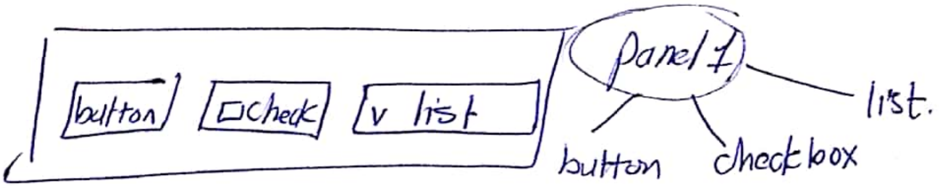
- by Creating layout → flow layout (default for applet)
→ grid layout
→ Border layout. (default for frame).

* For Now, no active On Components.

④ listener (listen events) : interface not class.

- button, list (when double clicked), text field → Action listener (action performed)
- choice, Checkbox, list → Item listener (Item State changed)
- Scroll bar → Adjusment listener (adjustment ~~sta~~ Value changed)
- Text field, Text area → Text listener. (Text value changed)

* Panel is Sub from Applet, we can put inside it Components.



* get Code base
URL of director
---- /classes

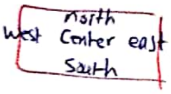
* get document base.
URL of document
--- /My applet.html

- AWT example :

1. في البدايه عمل (Object) من (button) وال (label) (Object from each Component) --- والباقي (Choice) و (Checkbox) وال

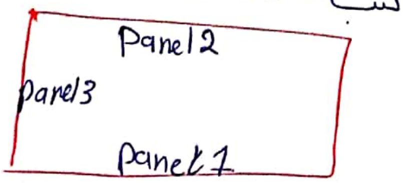
2.
- list (1) → true in Object → multi mode.
- list (2) → false in Object → Single mode.
- Text field → tf = --- (30) // 30 means number of col.
- Text area → ta = --- (30,40) // 30 & 40 → cols and rows.
- Check box group → cb2, cb3, cb4
  ☑ONE □Two □Three
  true — False.

3. ضفنا لكل وحدة (listener) مناسبة الها حسب Component
ممكن هون نحط فراغ ونجيب listener مناسب . listeners كتير .

4. 

← Border layout ← (layout) حددنا
ضفنا عليها (choice)

5. Create panels.
panel 1 → button, check box, --
panel 2 → lis1, lis2.
panel 3 → tf, ta

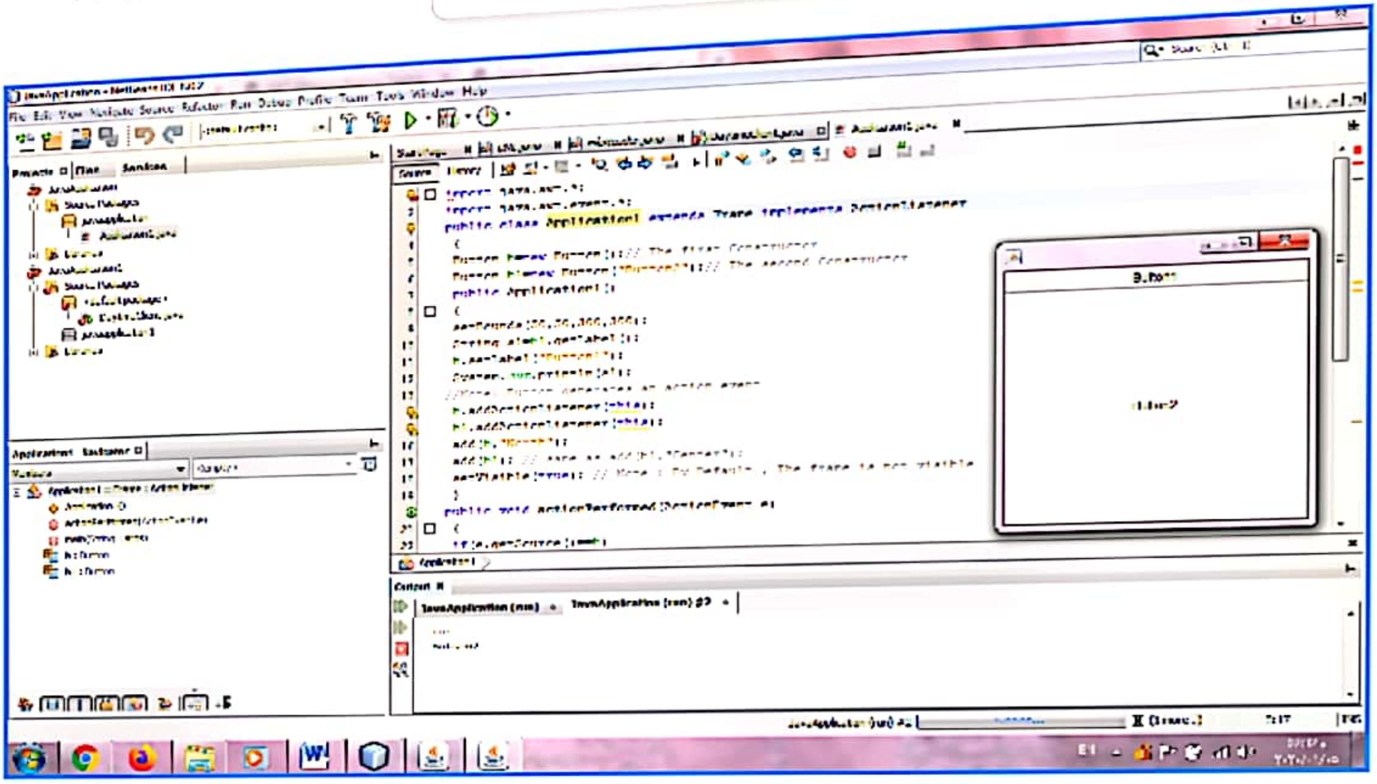
6. انتبه مكان ال panel وين موجودة :
panel 1 → South
panel 2 → North
panel 3 → West


7. In Action performed → if I pressed button What will happen and where will print !
See the Output ----

Component lec example 1

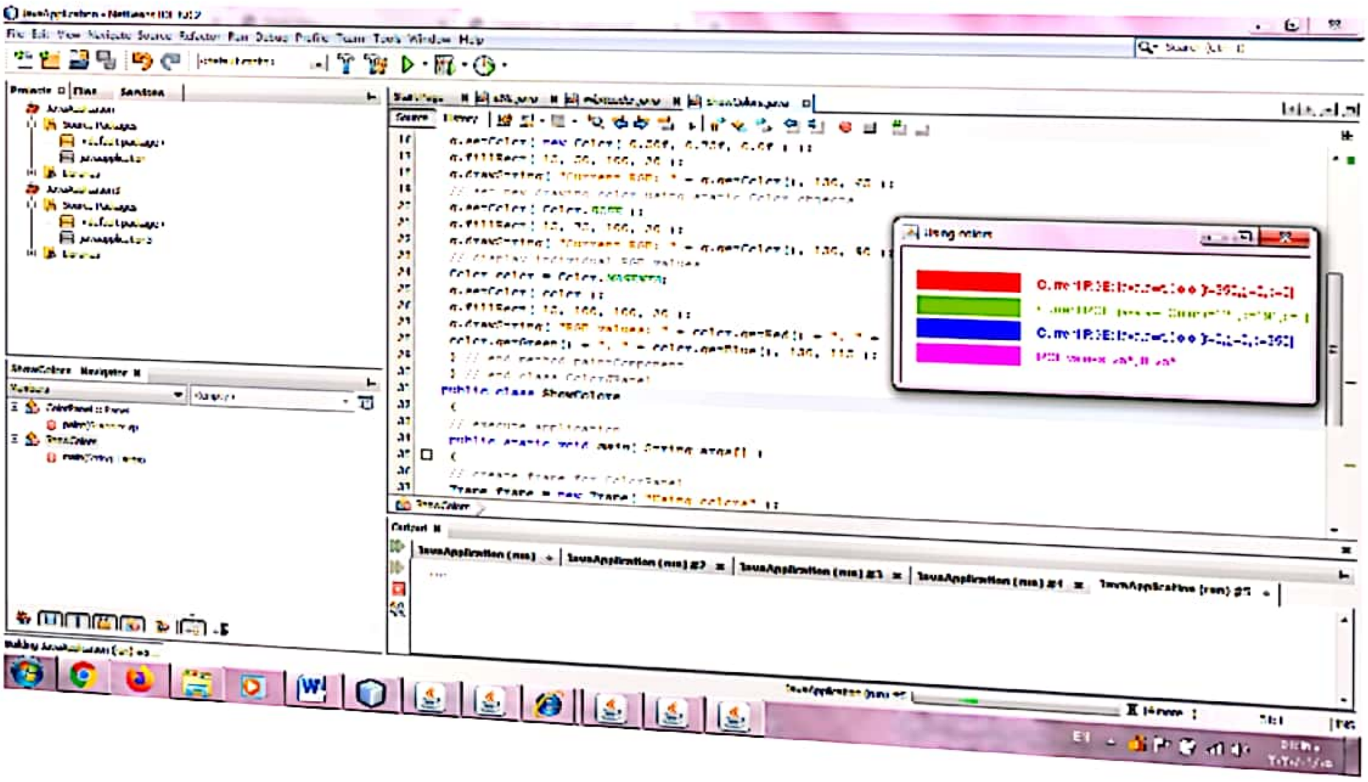
If button 1 pressd will print button 1 . the same for< >button 2

Select all Copy Add Comment



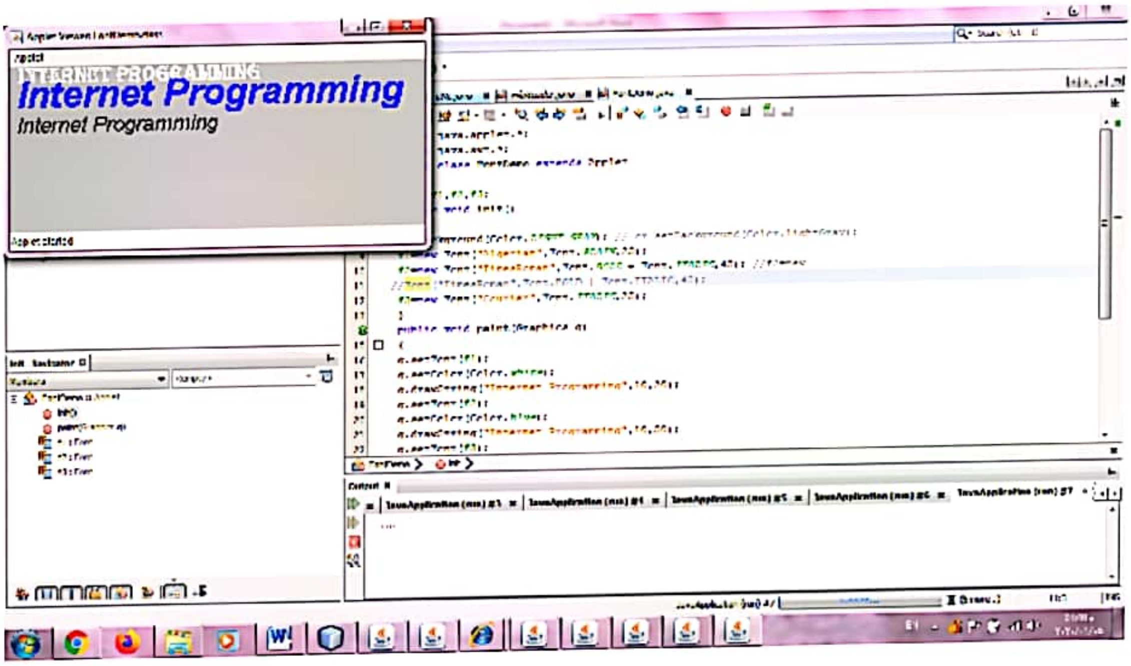
```
import java.awt.*;
import java.awt.event.*;
public class Application1 extends Frame implements ActionListener
 {
 Button b=new Button();// The first Constructor
 Button b1=new Button("Button2");// The second Constructor
 public Application1()
 {
 setBounds(50,50,300,300);
 String s1=b1.getLabel();
 b.setLabel("Button1");
 System.out.println(s1);
//Note: Button generates an action event
 b.addActionListener(this);
 b1.addActionListener(this);
 add(b,"North");
 add(b1); // same as add(b1,"Center");
 setVisible(true); // Note : by Default , The frame is not visible
 }
public void actionPerformed(ActionEvent e)
 {
 if(e.getSource()==b)
```

:Ghraphics lec example 1



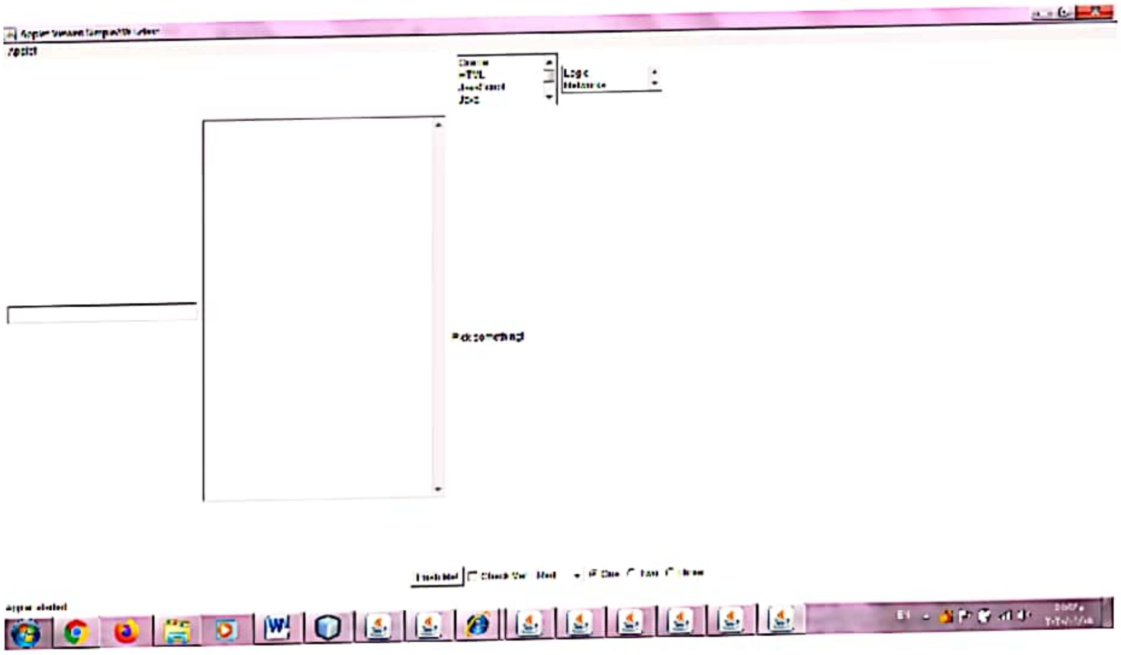
```
 g.setColor( new Color( 0.50f, 0.75f, 0.0f ) );
 g.fillRect( 15, 50, 100, 20 );
 g.drawString( "Current RGB: " + g.getColor(), 130, 65 );
 // set new drawing color using static Color objects
 g.setColor( Color.BLUE );
 g.fillRect( 15, 75, 100, 20 );
 g.drawString( "Current RGB: " + g.getColor(), 130, 90 );
 // display individual RGB values
 Color color = Color.MAGENTA;
 g.setColor( color );
 g.fillRect( 15, 100, 100, 20 );
 g.drawString( "RGB values: " + color.getRed() + ", " +
 color.getGreen() + ", " + color.getBlue(), 130, 115 );
 } // end method paintComponent
 } // end class ColorPanel
public class ShowColors
 {
 // execute application
 public static void main( String args[] )
 {
 // create frame for ColorPanel
 Frame frame = new Frame( "Using colors" );
```

:Ghraphics lec example 1

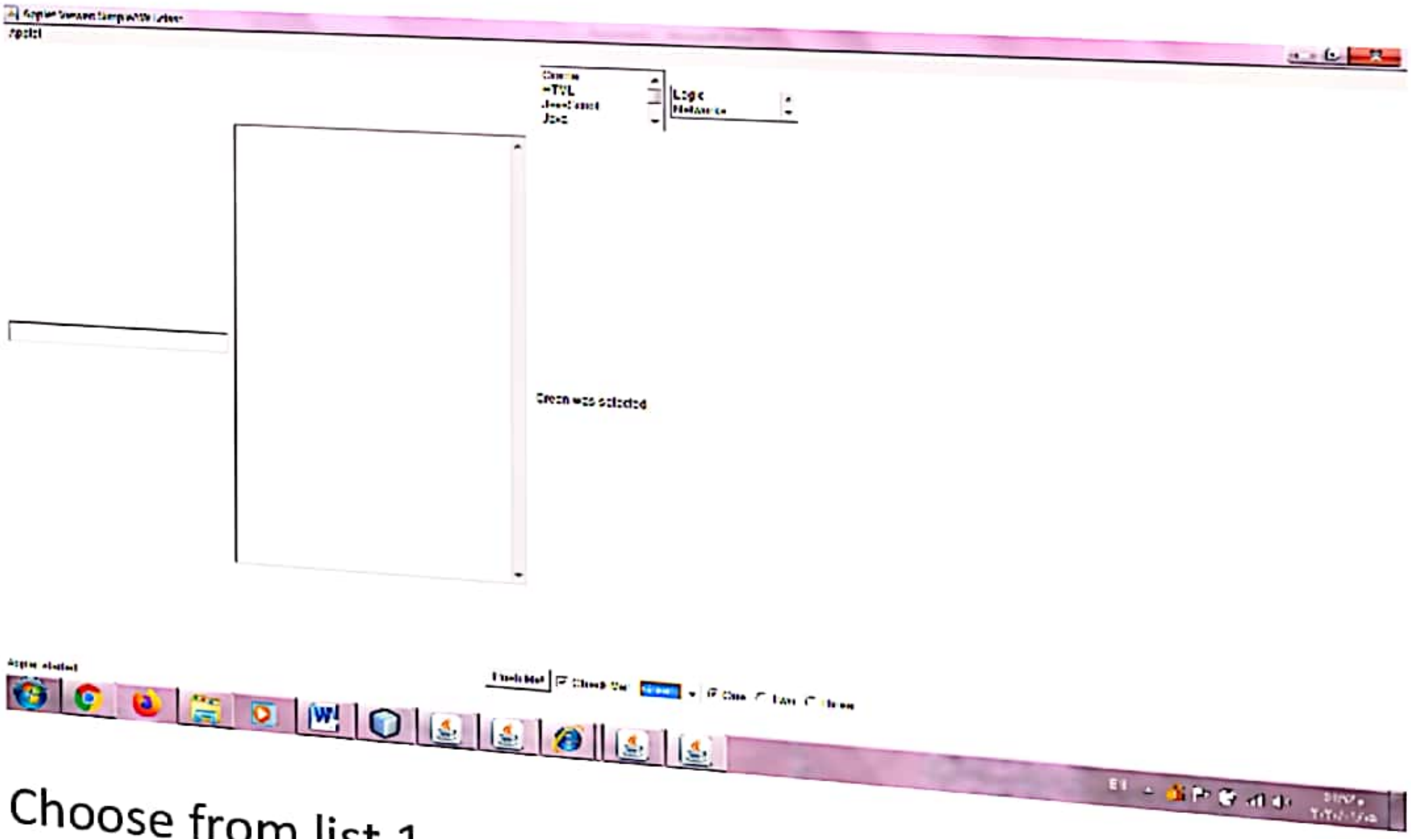


```
java.applet.*;
java.awt.*;
class FontDemo extends Applet
f1,f2,f3;
void init()
[illegible]
public void paint(Graphics g)
{
g.setFont(f1);
g.setColor(Color.white);
g.drawString("Internet Programming",10,20);
g.setFont(f2);
g.setColor(Color.blue);
g.drawString("Internet Programming",10,20);
g.setFont(f3);
```

Example in chapter 2 lec 6



Choose green from choice


Applet

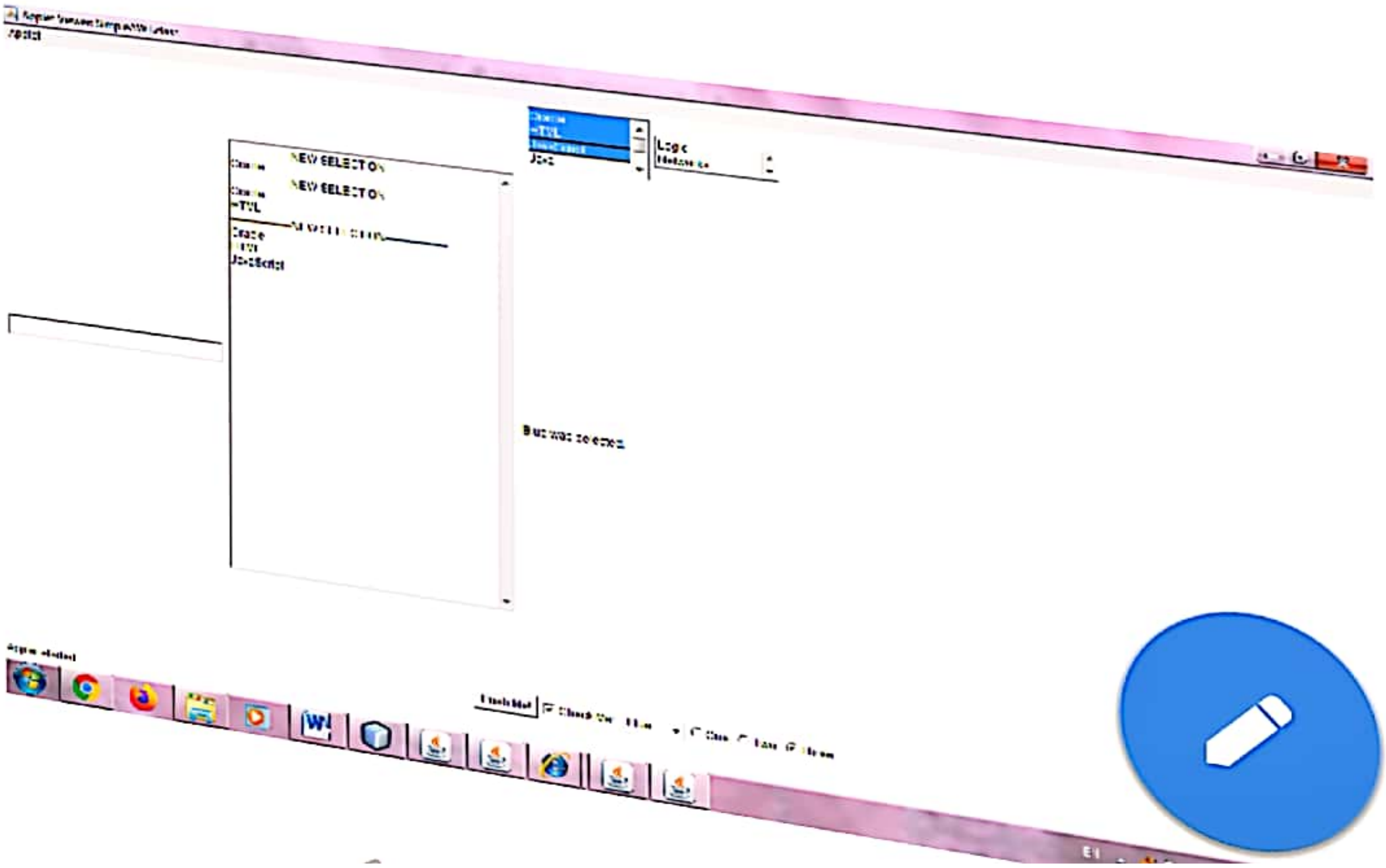
Oracle
HTML
JavaScript
Java

Logic

Green was selected

Choose from list 1

Write in text area


Applet

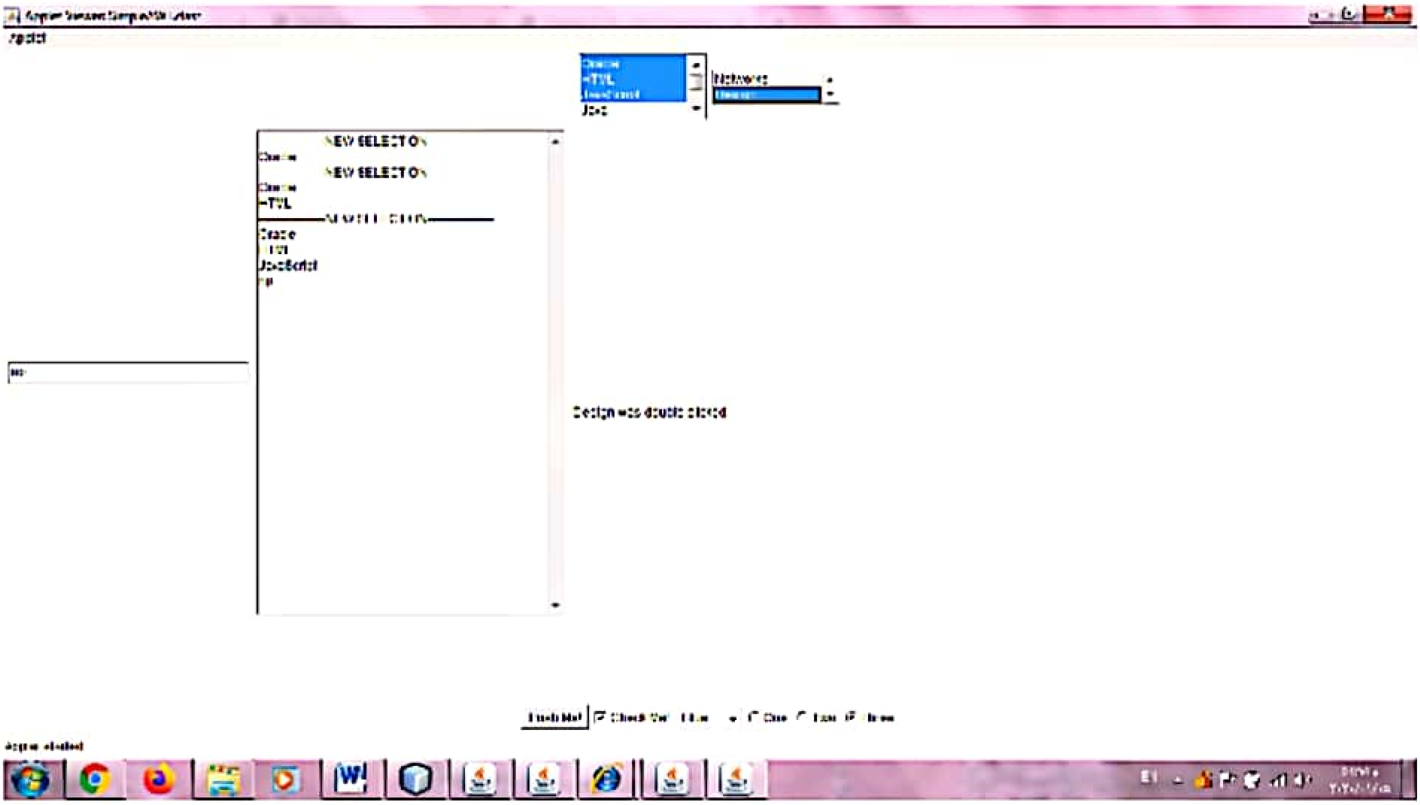
Oracle
HTML
JavaScript
Java

Logic

Oracle NEW SELECTION
Oracle
HTML NEW SELECTION
Oracle
HTML
JavaScript NEW SELECTION

Blue was selected

Choose design from list 2



```
NEW SELECTION
Oracle
NEW SELECTION
Oracle
HTML
----------NEW SELECTION----------
Oracle
HTML
JavaScript
```

Design was double clicked

(حسب شو صار اخر action)

المفروض ال ta , tf , on label

معبيين حسب شو اختار اخر اشي من components وكل component على مين بتأثر .

Chapter 3 Data Streams.

-Byte level Communication ⟶ Streams ⟶ Input (read)
⟶ Output (write).

* Input Stream:-

- Read from array /file or user.
- Difference between mark, skip
  ⟶ not Supported
  ⟶ not supported For all input Streams. [Use skip or mark, pushback]

→ Example <1>:

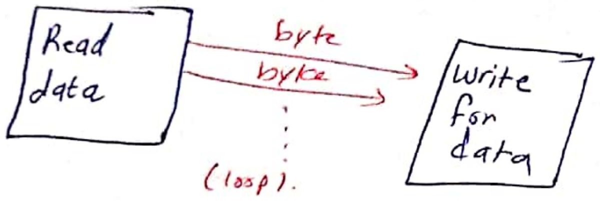
(Syntax, - - -) ← اشي بيطبع
بندخل File معين ازا مافي (File) لكون مكان (args[0]) بيلش يقرأ
وازا فيه File جبنا اسم
close ← بيكتبها data طول ماقايتنا (int data) ويخزن بال File ←

*Output Stream.

-Write on File ,array, monitor.

→ Example <2>.

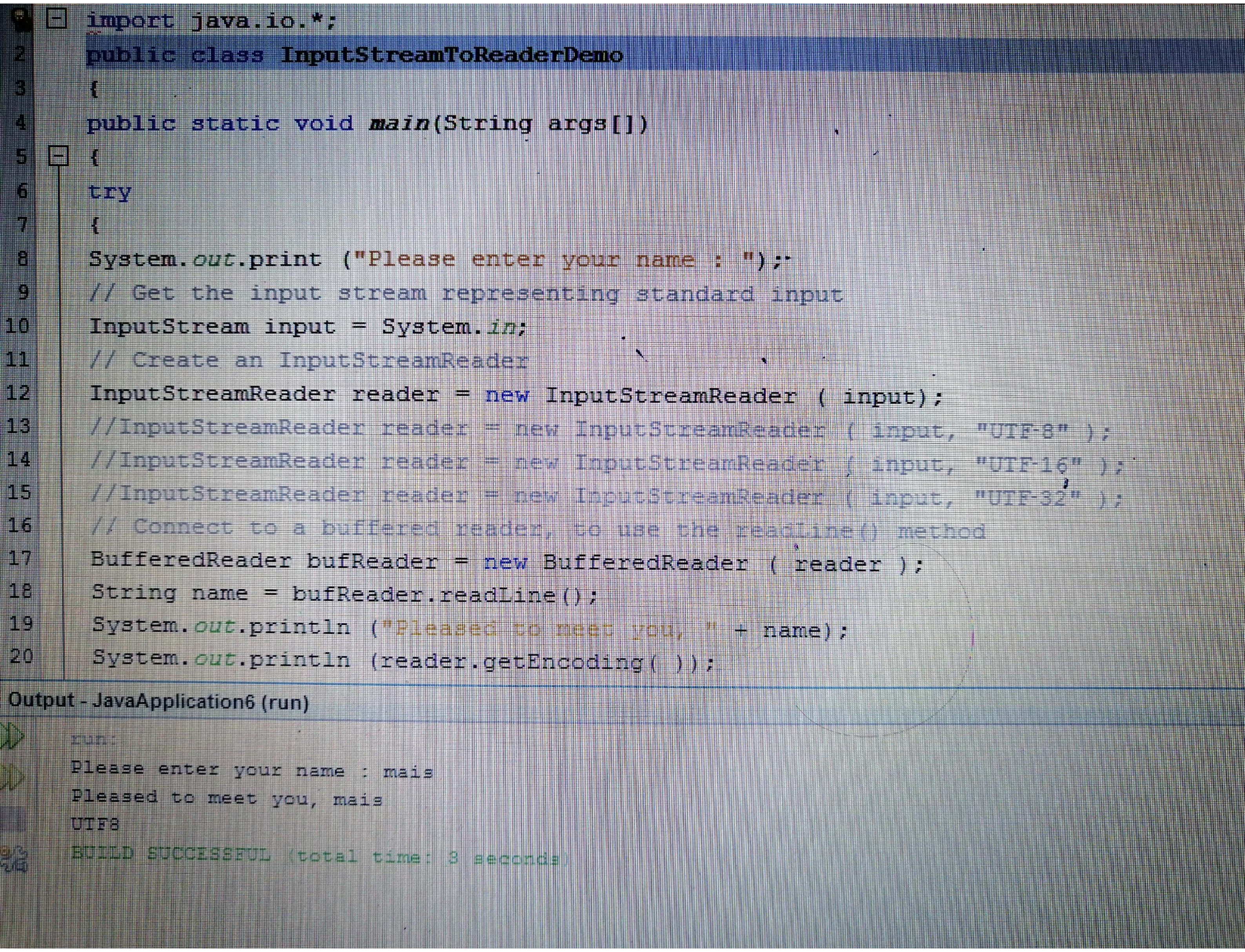
-read from src and write to destination.



* To read int, char, string (line by line) not byte byte ⟶ Use filter

(Read, write) ⟹ like Input and Output Stream but support unicode char
↓
Same function with Input & Output Stream but:
available() ⟶ Ready()

⟶ Input Stream Reader (byte to char)
⟶ Input Stream Writer (char to byte).

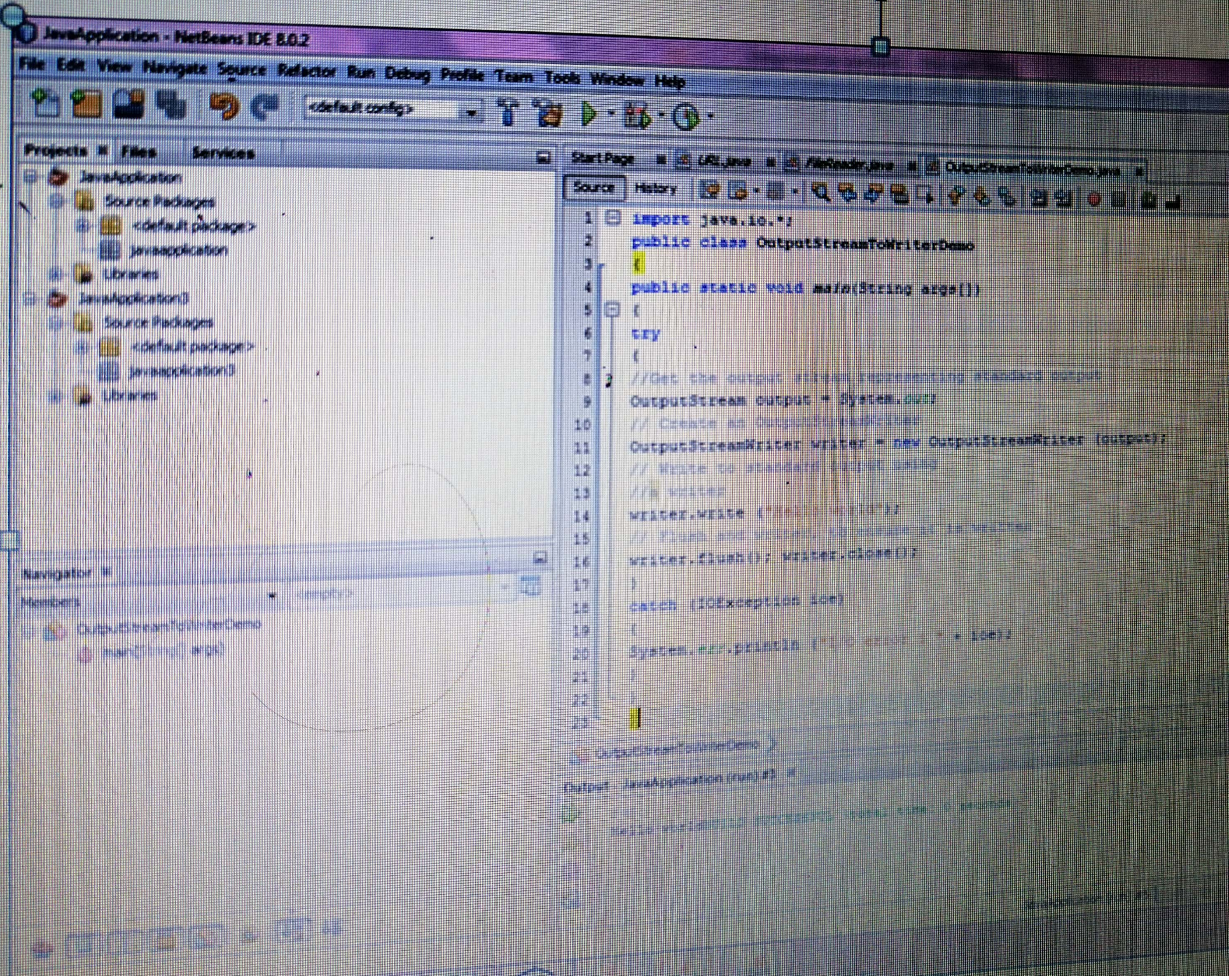


```java
import java.io.*;
public class InputStreamToReaderDemo
{
public static void main(String args[])
{
try
{
System.out.print ("Please enter your name : ");
// Get the input stream representing standard input
InputStream input = System.in;
// Create an InputStreamReader
InputStreamReader reader = new InputStreamReader ( input);
//InputStreamReader reader = new InputStreamReader ( input, "UTF-8" );
//InputStreamReader reader = new InputStreamReader ( input, "UTF-16" );
//InputStreamReader reader = new InputStreamReader ( input, "UTF-32" );
// Connect to a buffered reader, to use the readLine() method
BufferedReader bufReader = new BufferedReader ( reader );
String name = bufReader.readLine();
System.out.println ("Pleased to meet you, " + name);
System.out.println (reader.getEncoding( ));
```

Output - JavaApplication6 (run)

```
run:
Please enter your name : mais
Pleased to meet you, mais
UTF8
BUILD SUCCESSFUL (total time: 3 seconds)
```

OutputStreamWriter Example

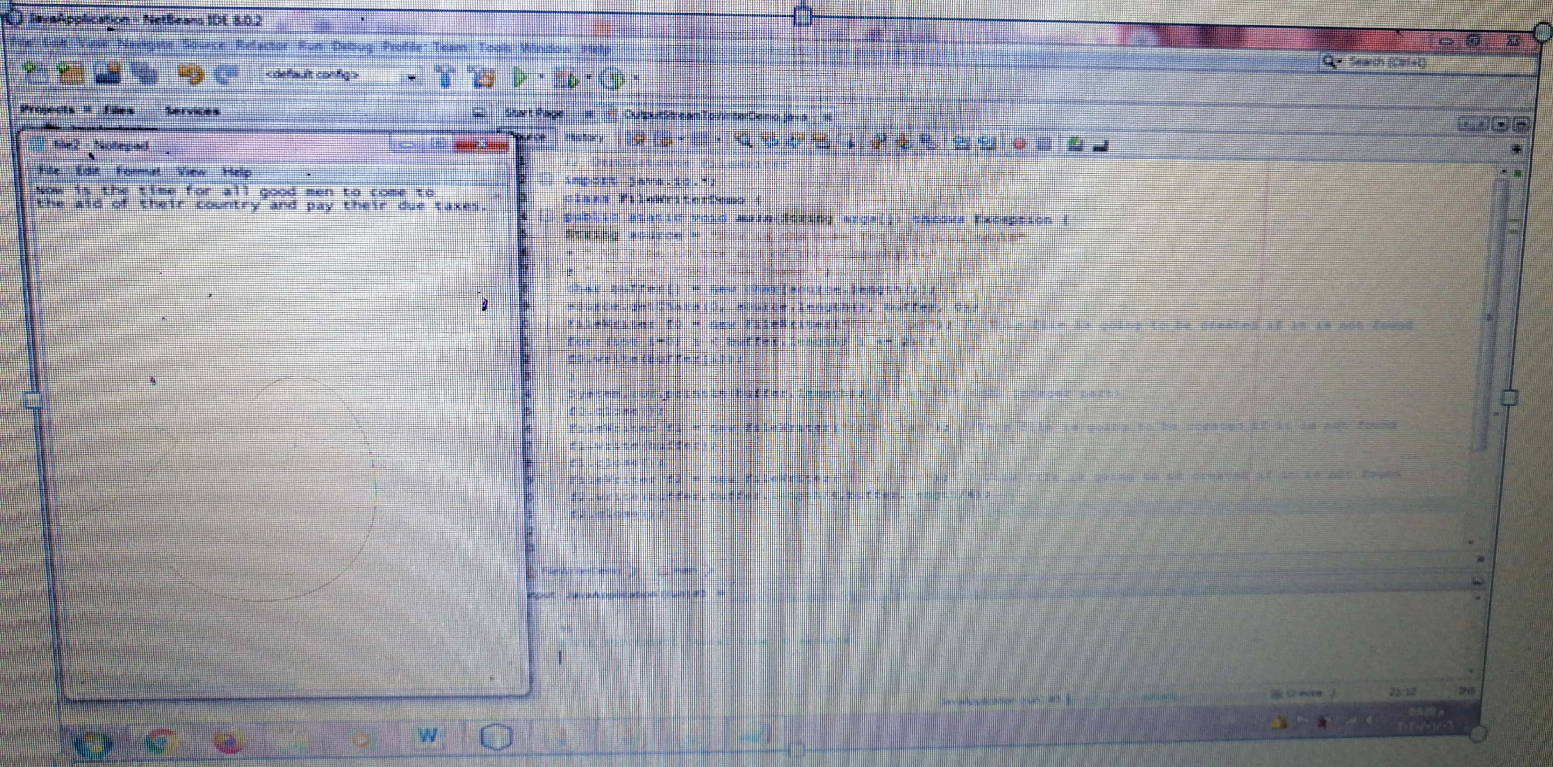


```
import java.io.*;
public class OutputStreamToWriterDemo
{
public static void main(String args[])
{
try
{
//Get the output stream representing standard output
OutputStream output = System.out;
// Create an OutputStreamWriter
OutputStreamWriter writer = new OutputStreamWriter (output);
// Write to standard output using
//a writer
writer.write ("Hello World");
// Flush and writer, to ensure it is written
writer.flush(); writer.close();
}
catch (IOException ioe)
{
System.err.println ("I/O error : " + ioe);
}
}
```

```
Hello WorldBUILD SUCCESSFUL (total time: 0 seconds)
```

Last example in data streams examples (example 8)

Note : File 2 created automatically because we write on it and write the string source , but if we want to read from file , exception will happen if file does not exist . print the length which is 95



JavaApplication - NetBeans IDE 8.0.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Projects Files Services
Start Page
OutputStreamToWriterDemo.java
Source History
file2 - Notepad
File Edit Format View Help
Search (Ctrl+I)

```
Now is the time for all good men to come to
the aid of their country and pay their due taxes.
```

```
// Demonstrate FileWriter.
import java.io.*;
class FileWriterDemo {
public static void main(String args[]) throws Exception {
String source = "Now is the time for all good men\n"
+ " to come to the aid of their country\n"
+ " and pay their due taxes.";
char buffer[] = new char[source.length()];
source.getChars(0, source.length(), buffer, 0);
FileWriter f0 = new FileWriter("file1.txt"); // This file is going to be created if it is not found
for (int i=0; i < buffer.length; i += 2) {
f0.write(buffer[i]);
}
System.out.println(buffer.length); // [illegible]
f0.close();
FileWriter f1 = new FileWriter("file2.txt"); // this file is going to be created if it is not found
f1.write(buffer);
f1.close();
FileWriter f2 = new FileWriter("file3.txt"); // this file is going to be created if it is not found
f2.write(buffer,buffer.length-buffer.length/4,buffer.length/4);
f2.close();
```

```
95
BUILD SUCCESSFUL (total time: 0 seconds)
```

CHapter "5" ((UDP))

* UDP →
- Connectionless (no flow and error Control)
- Faster than TCP
- Real time Application like Videos.

* To describe UDP ——→ (by Simulation)
- *packet (byets(data), Addressing info)
- *Socket (to Send & recieve packet)

- patkets are sent using datagram Socket.
- Note: port number determines in datagram packet → (Simulator) but in real, in Transport Layer.

* Datagram packet: →
- Send: must determine IP Address & port no. For destination
- Recieve: Just data (array of bytes).

بنحقق behavioar انت الذات

* Datagram Socket: →
- DatagramSocket() → Port. مارح توصل ماشي
- DatagramSocket(int port)
- DatagramSocket(int port ; InetAddres adress)

Slide 28 : example

Create a packet include array of bytes (256 bytes) initially Zero, Create Socket on port 2000, while true recieve packets then do Some processing on bytes, then Close

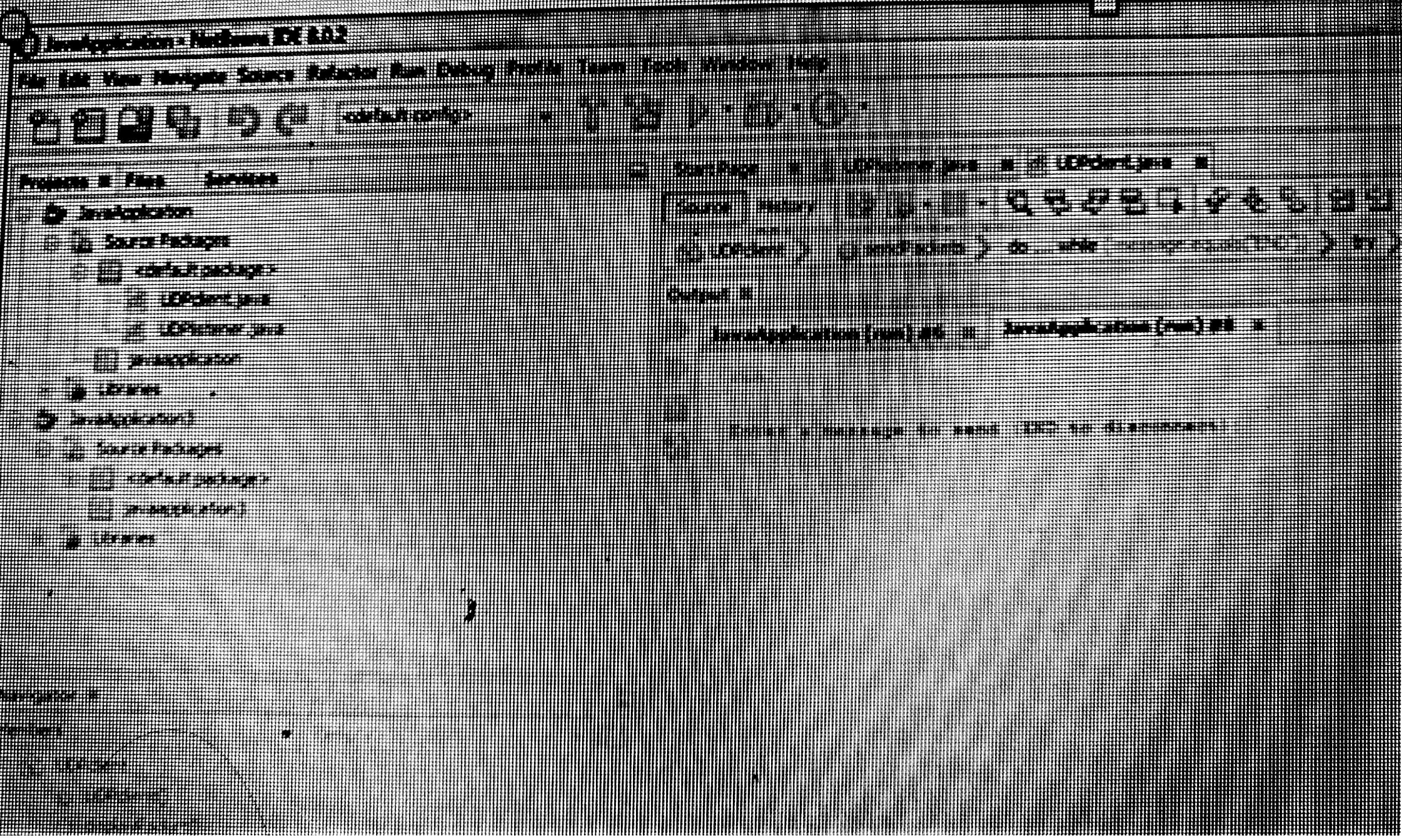
next examples.

* processing on bytes ——→ by byte array input Stream.

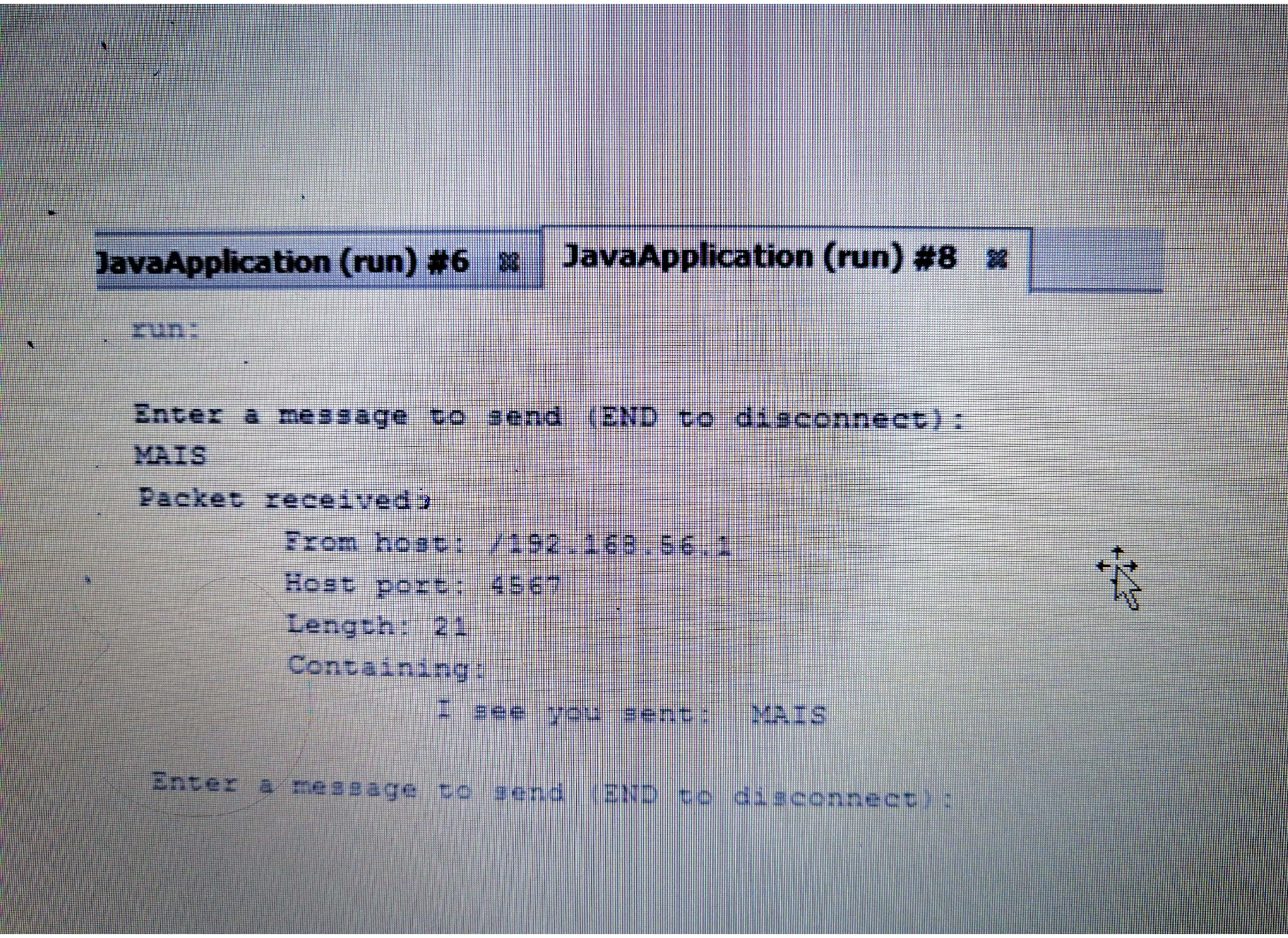
IF processing on String, int, .. ——→ bridge by data input Stream.

UDP EXAMPLE (IMPORTANT) : 5-2

FIRST :



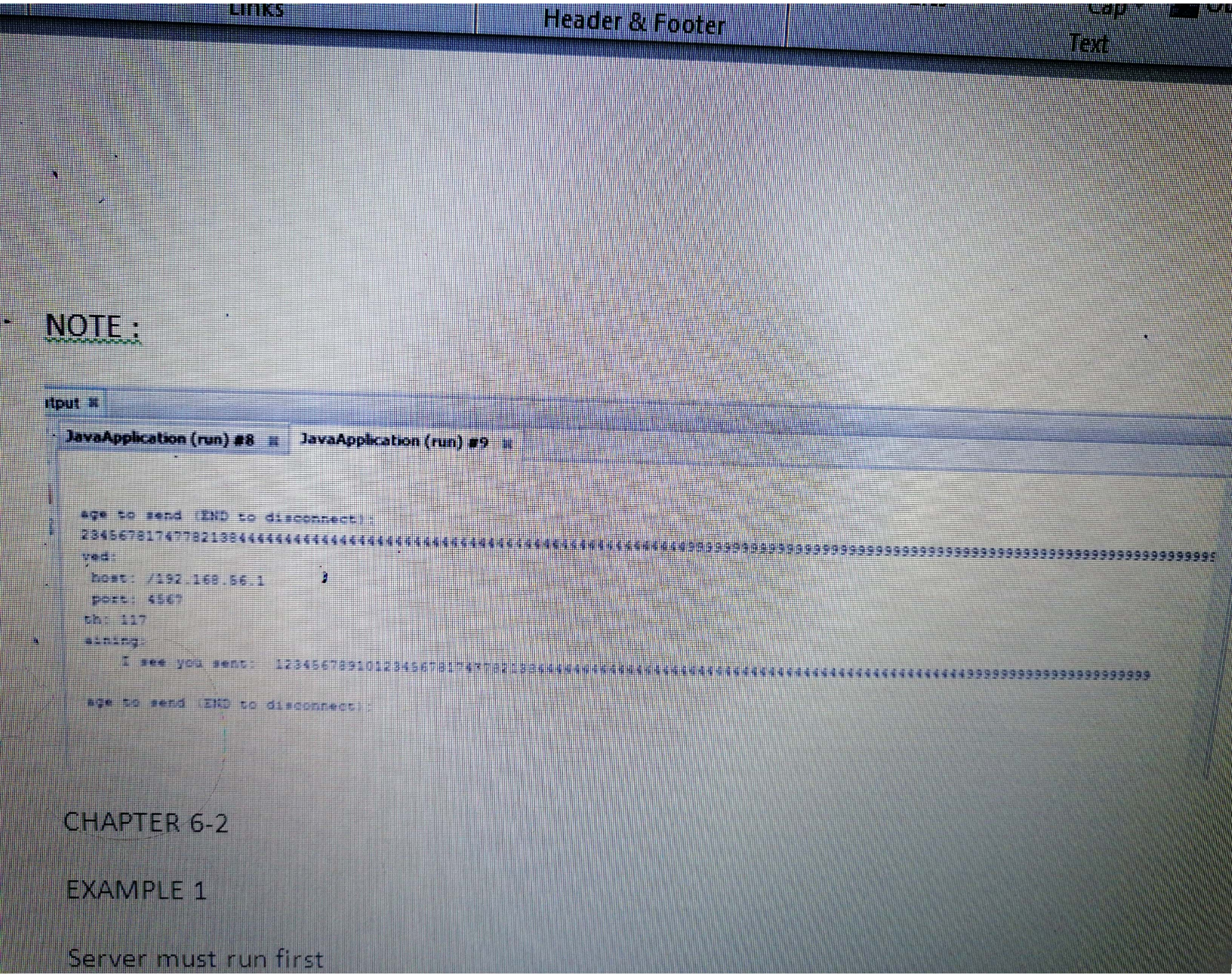
```
Enter a message to send. 'END' to disconnect.
```



```
run:

Enter a message to send (END to disconnect):
MAIS
Packet received:
        From host: /192.168.56.1
        Host port: 4567
        Length: 21
        Containing:
                I see you sent:  MAIS

Enter a message to send (END to disconnect):
```



NOTE :

```
age to send (END to disconnect):
23456781747782138444444444444444444444444444444444444444444444444999999999999999999999999999999999999999999999999999999999995
ved:
 host: /192.168.56.1
 port: 4567
th: 117
aining:
    I see you sent:  12345678910123456781747782138444444444444444444444444444444444444444444444444499999999999999999999

age to send (END to disconnect):
```

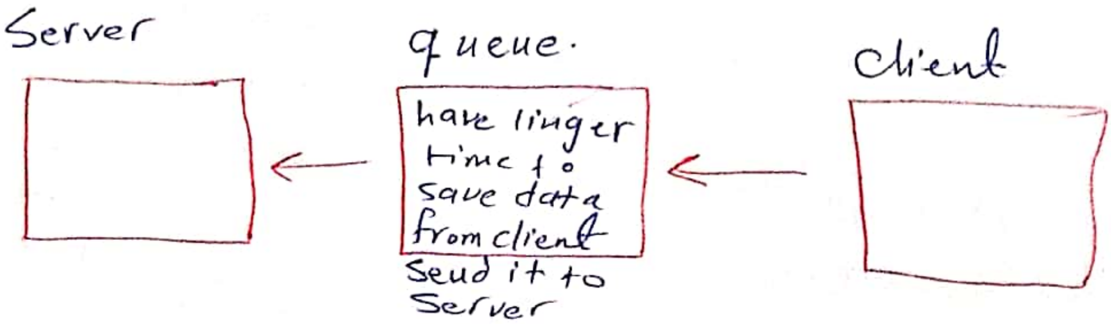
CHAPTER 6-2

EXAMPLE 1

Server must run first

CHapter <6> TCP → Reliable, flow & error Control.

- in UDP we call packet → datagram
- in TCP we call packet → packet
- Socket (For Connection)
- every one Socket for each Connection between Two hosts.
- TCP must be listening for Connection before Sending Packets
- Send and Recieve data by: (get InputStream, get Output Stream)
  ↳ low Level → Filter

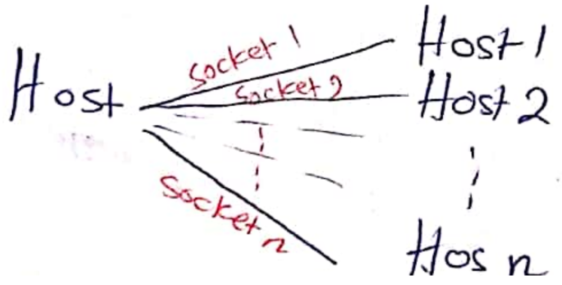


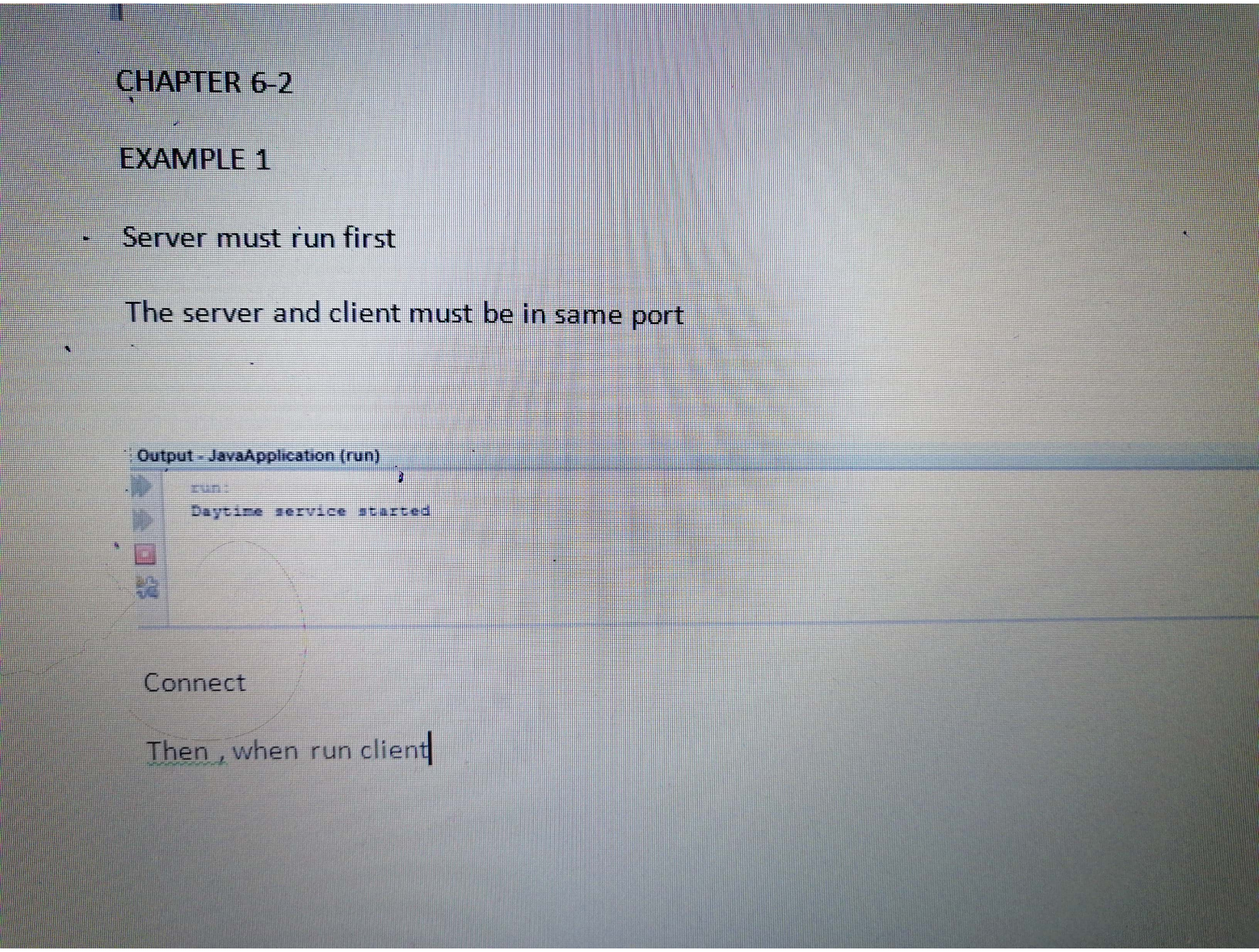
(5-50) Requests.

Notes: - Flush before Close (to make sure the queue is empty)
- In Server Socket, Don't determine IP Address, it is not importa
  ↳ Determine port, number of clients Just.
  (you Can determine IP address (optional))

- in TCP: Server program execute first (to listen)
- in UDP: client program execute first (no need to listen)

Operations in TCP: Connect → Send & Recieve → close.
Full duplex.





CHAPTER 6-2

EXAMPLE 1

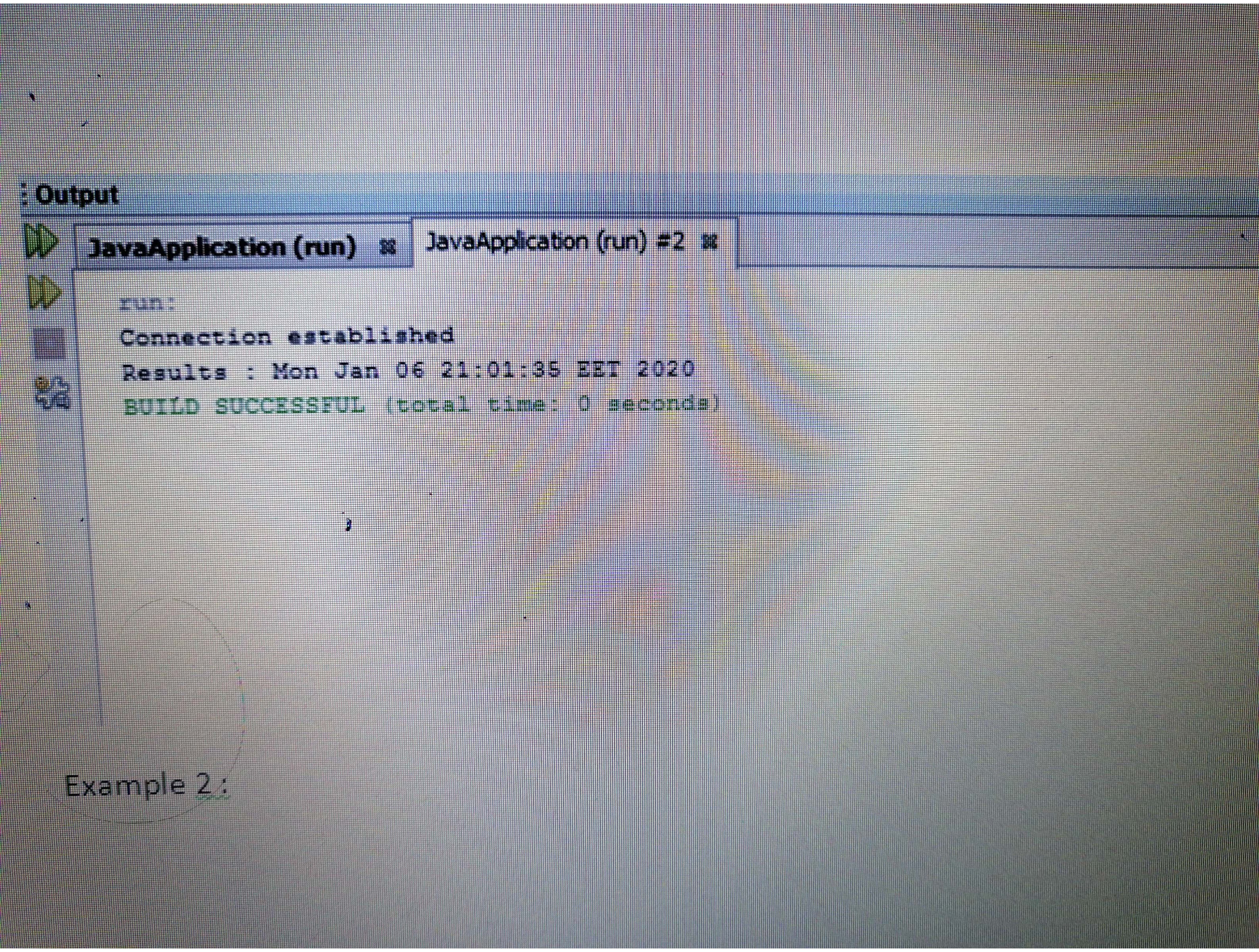
- Server must run first

The server and client must be in same port

```
run:
Daytime service started
```

Connect

Then , when run client



```
run:
Connection established
Results : Mon Jan 06 21:01:35 EET 2020
BUILD SUCCESSFUL (total time: 0 seconds)
```

Example 2 :

Chapter, 7-1

- Servlet vs. CGI

Servlet ↓ process for all ~~every~~ Requests. (good for performance).

CGI
→ not memory efficient
→ lots of slots.
→ Every Request take separate process.

- Servlet
  → Generic (any Server exclude HTTP)
  → Web (For http Requests)
  } must implements. (interfaces)

* When Request recieved → load servlet if it is not loaded or the expire time finished then process and send response.

* Process: when Request in HTTP Servlet → Send Request to service() method which include (doget(), dopost())

* What is the Functions execute automatically? after Creation Obj

(Init, Service, Destroy)

[* GET: Client needs Something from Server.
* POST: ~~client writes on Server Something~~ Server needs sth from client.] most important methods.

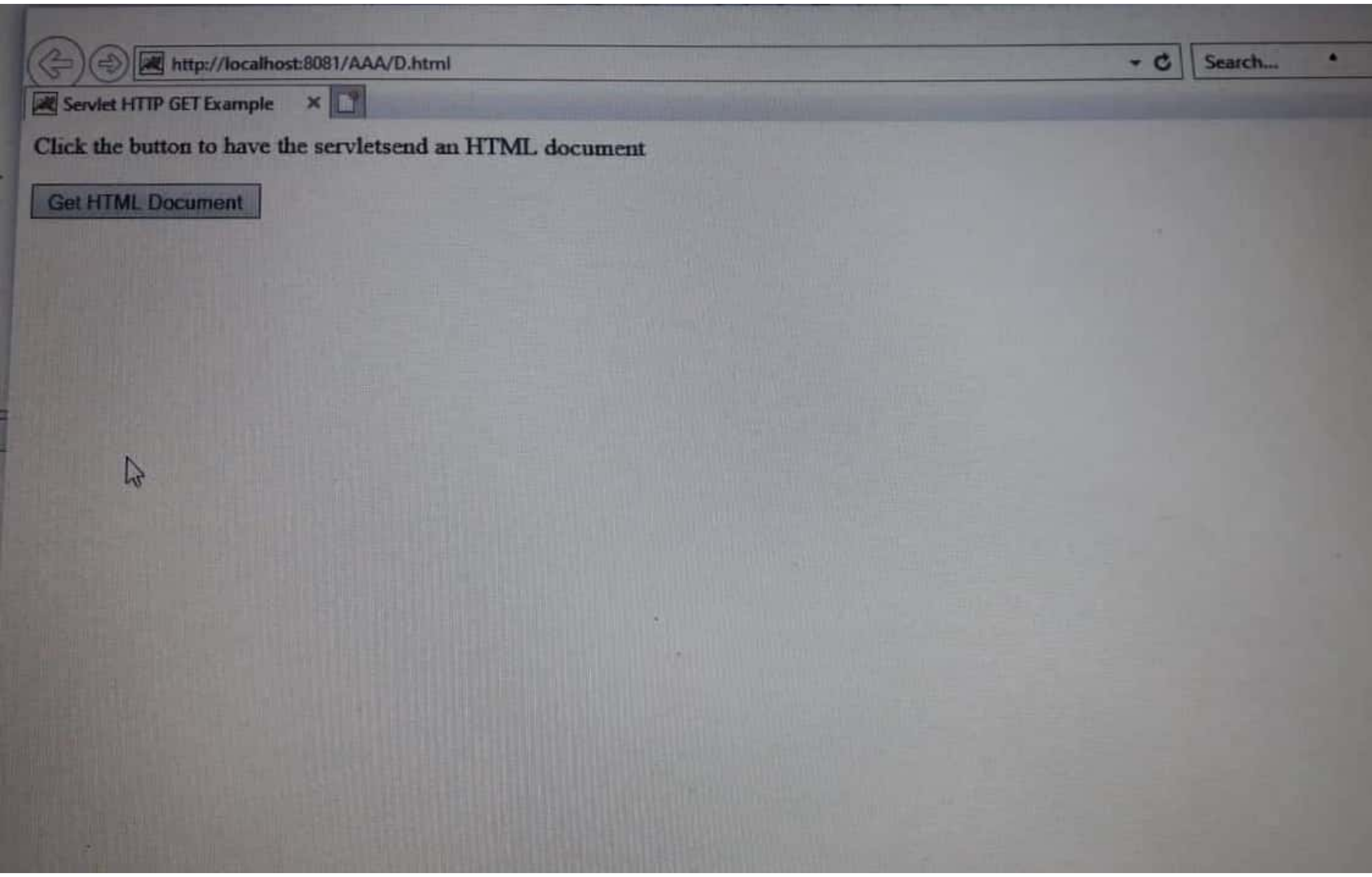
→ in any html text: [ACTION, method, Inputs] must assign them, otherwise Optional.

in Servlet Example:

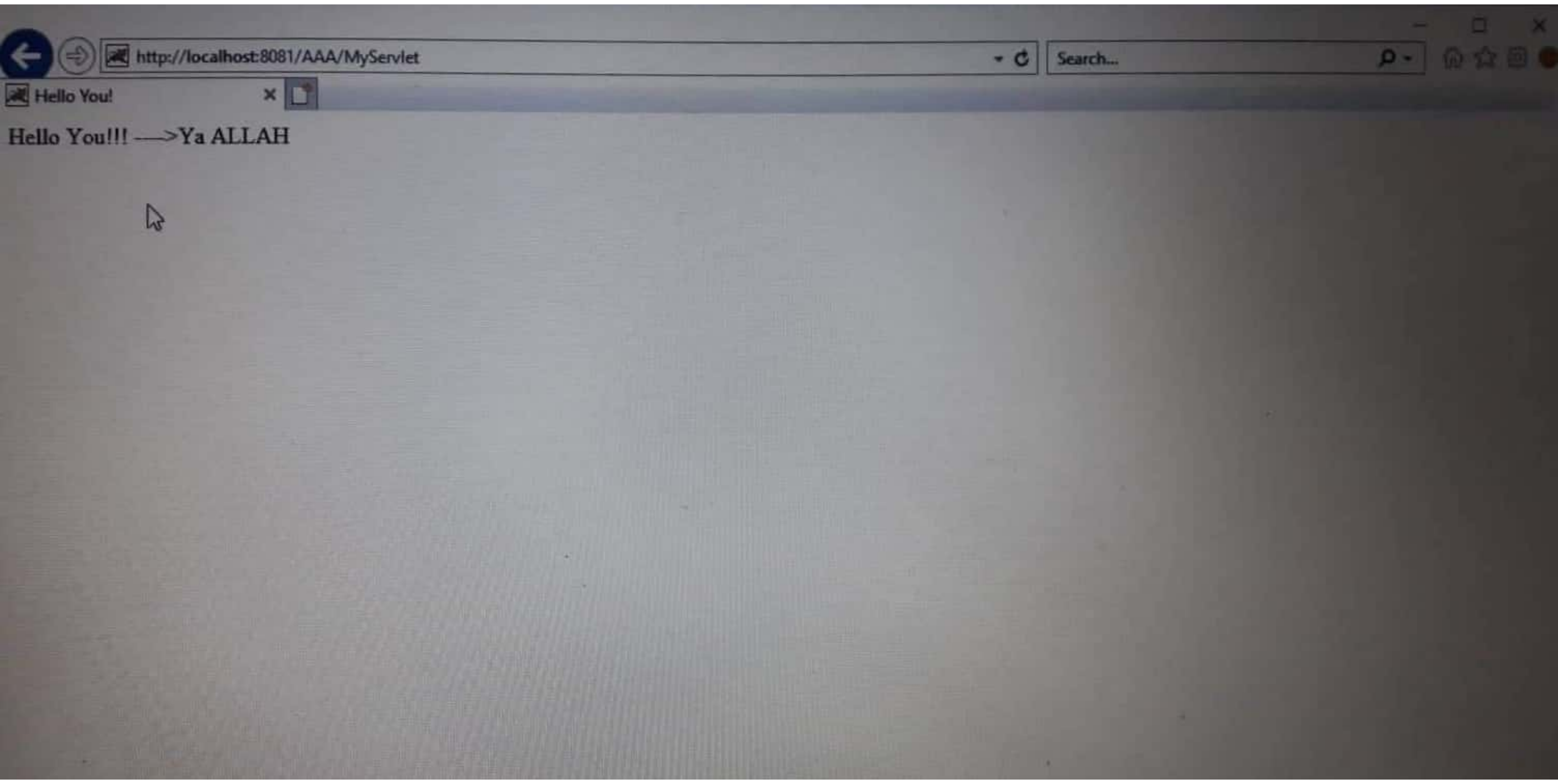
there is 2 buttons. [Get html Documnt] → pressed will print something

and: [ ] [Submit] → enter name and submit

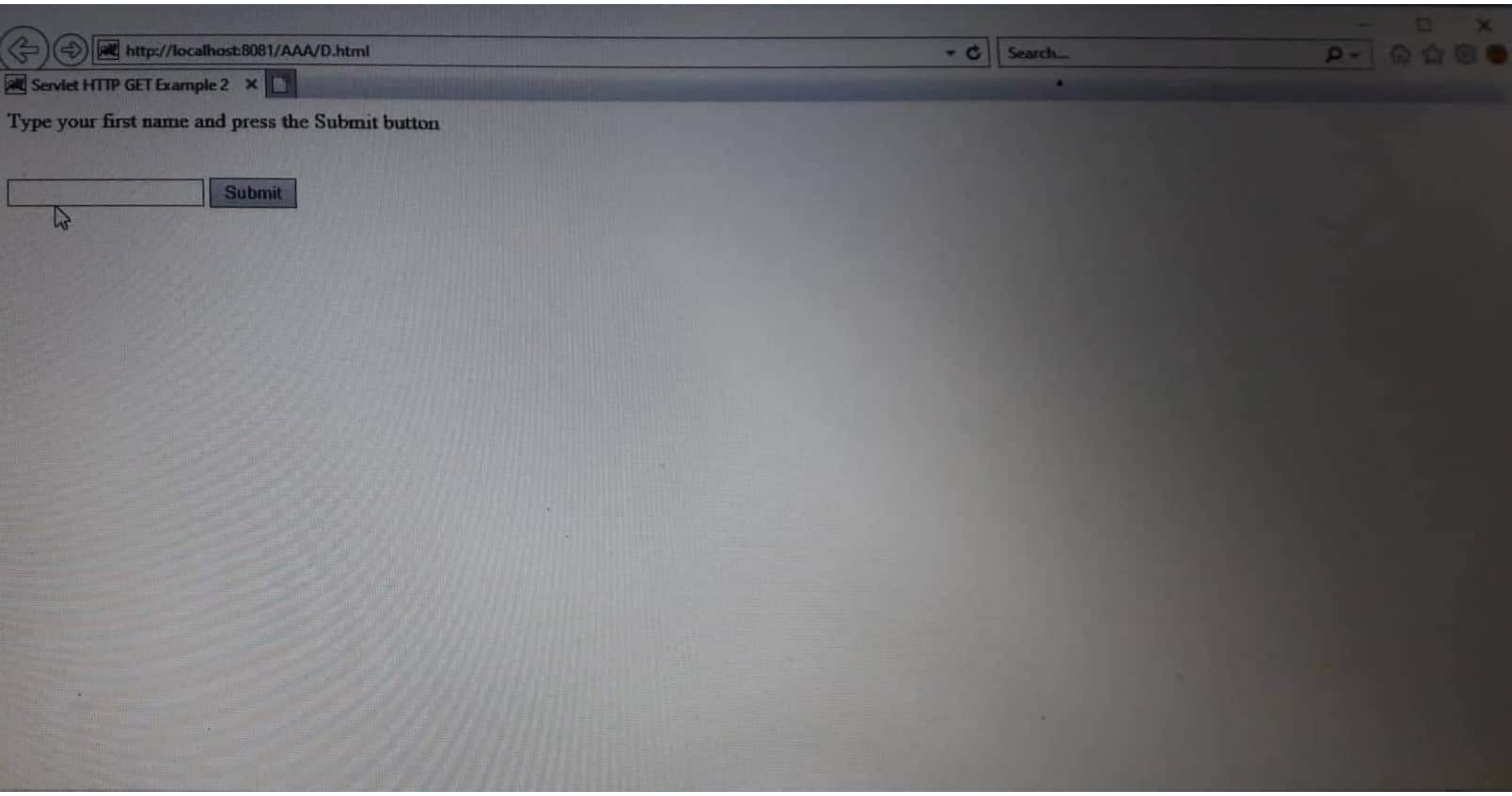
enter name.



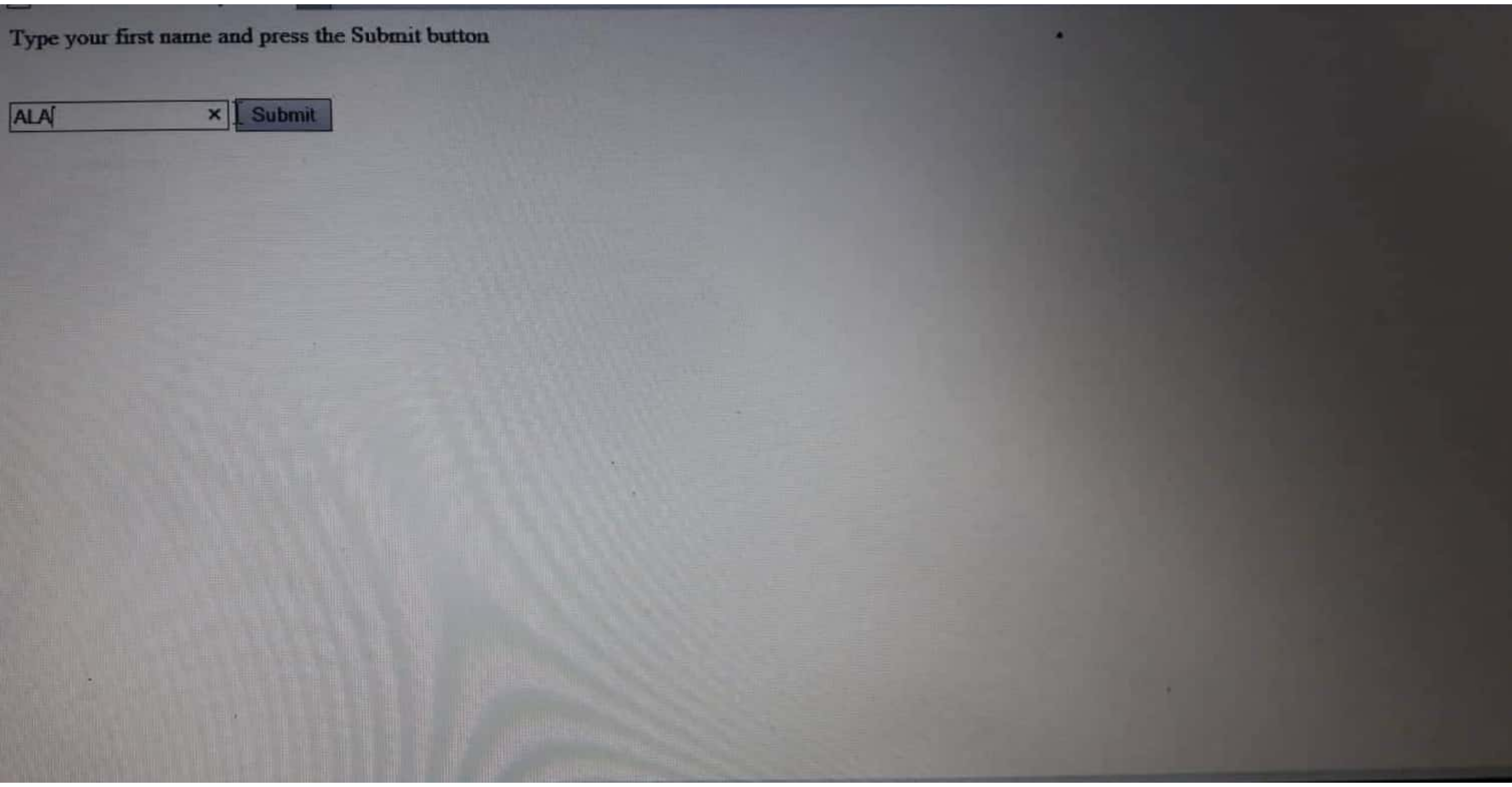
Click the button to have the servletsend an HTML document



Hello You!!! —>Ya ALLAH

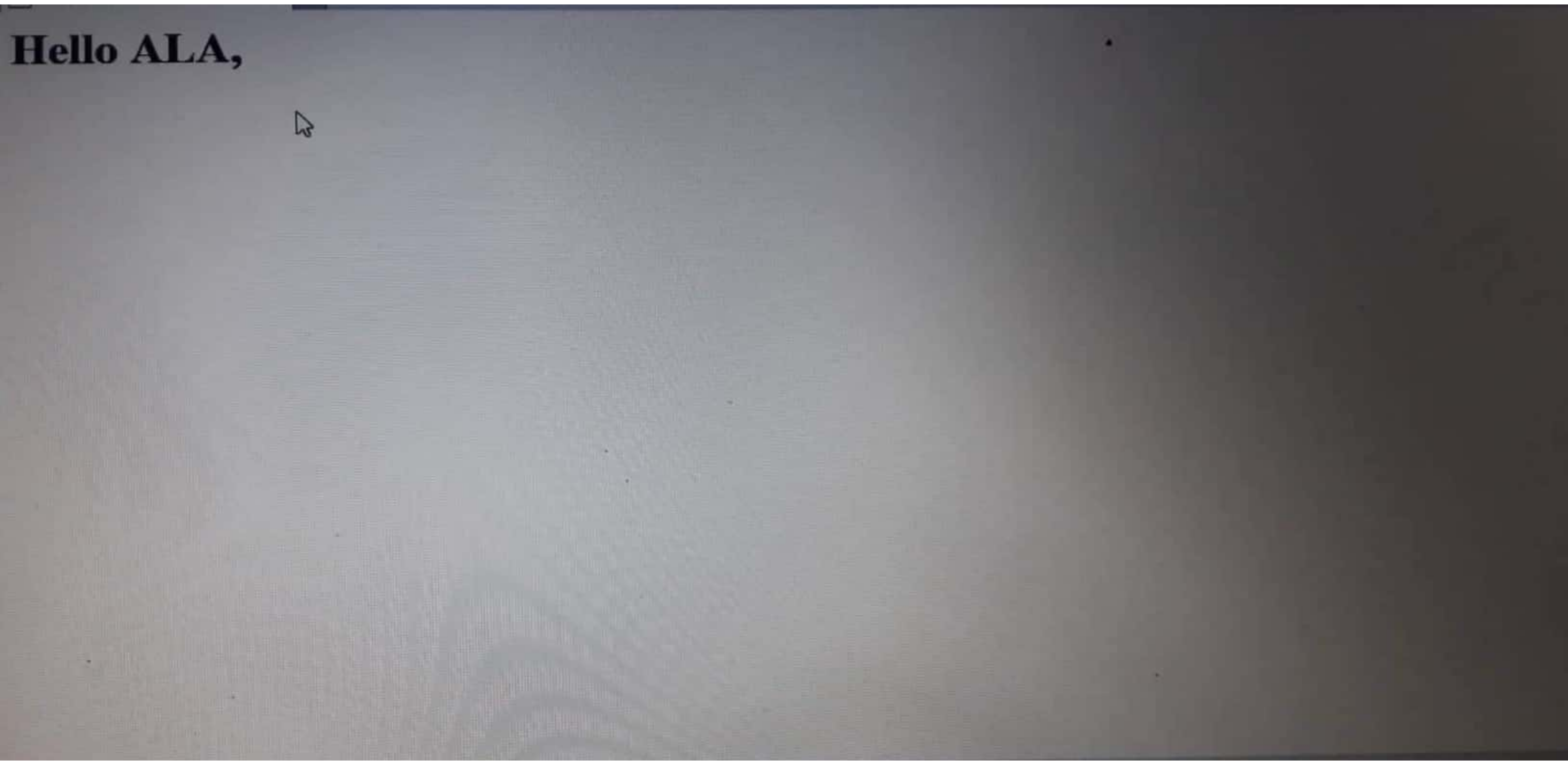


Type your first name and press the Submit button



Type your first name and press the Submit button

ALA

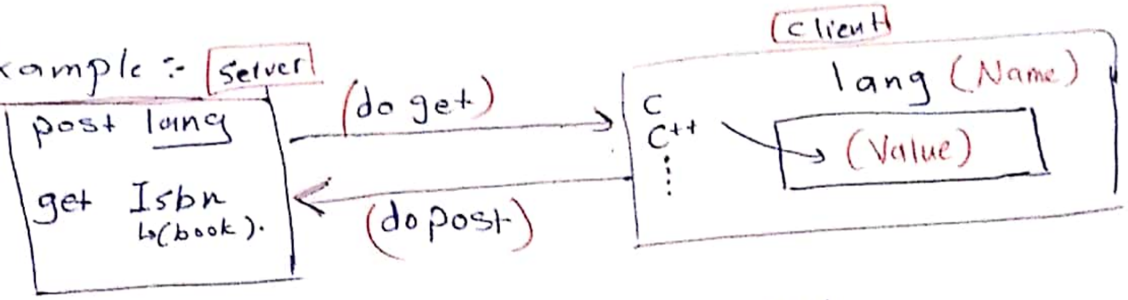


Hello ALA,

Session Tracking : (7-2)

* methods : - putValue (name , Value);
- get Value (name);
- getValueNames ( )
- removeValue(name)

* Example :-



الفكرة :

*

* 2classes in two Seperate files → dopost
→ doget

① do post

- String language = request.getParameter("lang")

* بدنا نجيب من (client) ← (lang) ونخزنها بال (language).

- Http Session session = request.getSession(true);

* اعمل ال create لل session بهاد Server مع ال Client

- Session.putValue (Language , getISBN (language));

name value.

ON Session

- output = response.getWriter()

* هون بدي اكتب على client باستخدام ← getWriter or getOutputStream

- Output.Pointln Statements : Send to client the html tags.

② Do get !

- HttpSession session = request.getSession (false);
  - → not true ! final Question.
  - → Session = null

- If (session != null) ⋮] [if session established before return Valuenames.]

* the second if statement with for:

Recommendations , value , How to program. ← (Valuename) ازا فيه ←

ISBN #: , book ,

* Why For loop ! IF there is multiple Values . F

* Note: String value = (String) session.getValue (ValueNames [i])

بيرجع Object ← بعمل Casting ل String

← ازا مافيه (Value) يعني مادخل و language X : رح يطبع الموجود بال

→ (else statement)

* Required html files:-

ACTION, METHOD, INPUT → important , else → Optional.

- ACTION: link
- METHOD: get, post
- INPUT: Components.

→ in input Statements:

Type → built in (understood by Server)

example : sumbit, reset, radio.

Name → important For Server (client does not use it)

So 2 html Files Required: One For get , One for post.

* CHapter [8] multithreading

- ref, start() ⟶ in Ready State.
- run() ⟶ From Ready to Running State.
- ~~[illegible]~~ sleep(ms) ⟶ From Running to Sleeping State.

program #1:-

Created ~~four~~ 4 references and 4 threads, each thread for each ref. then Start all threads, when we create threads: For example first thread ⟶ point 1 ⟶ will go to class thread1 and pass ref1 to int[]i and "First thlead" to S1 ⟶ then go to run method, Sum For values in ref 1 and print:

[I am, the first thread, done With Sum = المجموع ~~[illegible]~~ loop بال]

⟶ It will do the Same for thread 2,3,4

* When Finished all threads ⟶ isAlive() will be false for every thread So; See in the main function, if all isAlive() False ⟶ it will print: The final output is - summation for all ref1, ref2, ref3, ref4 and print false ⟶ 4 times.

program #2:

- Why creat 4 threads with 4 classes with same code!
  Create array of threads (point1[0], [1], ...) and One class for all

* and do the Same thing we did in program 1 with less Code.

We use: For example For thread1 ⟶ ref = point[0]
to bring the name ⟶ point1[i].getName()
to know if it is alive ⟶ point1[i].isAlive()

(this.S1) in class thread1 means this Object (String).
(i) " " " " ref array for Point1[1], point1[2], P ----

Scanned by CamScanner

Note : program 2

It will not execute Sequentially as shown in Output for trial 1 & trial 2

- program #3

we create Two threads with two Reference (point 1, point2)
we put the higher priority for pointer 1 ,and do the Same things as before .

Output if don't Set priority : (Comment on priority Statement)
no one before one ,(first or Second) → depends on OS it will execute .

if with priority statement :
point1 → thread1 will execute first because of higher priority.

- Program #4:

As program #3 ,but with one Object (point 1) and two References (point 1 [0], point 2[o)

- Program #5

as program #4, except: Using Thread.yield()

How it is work? doing first thread then give chance For Second thread and so on --

- program #6-A

Created Object as global (before main) ,must put Static, and Same Program but used → (Join) inside loop in class thread 2, Join means: Don't execute thread 2 before thread1 go to dead State (finish)

ThreadDemo5.point1.join()

↳ to access thread 1

↓ wait for thread1 to die

→ check every loop if thread1 finished or not!

Program #6-B:

Same as (6-A) but → put the join before loop.
Why to check every loop and consume CPU time !!
Check before enter thread2, with same Output.

Program #7:

Same as before.
but used Sleep (1000) → means: thread2 will be in Sleep State for 1000ms, so thread 1 will execute first until 1000ms finish, when 1000ms finished, thread 2 will go to ready State then run.

Program #8:

Using Interrupt

- For First time thread1 will execute, and this statement will print.

First Object thread False → is Interrupted()
↓
no Interrupt in first time.

From: Thread.currentThread.getName()

- then execute thread2:

→ ThreadsDemo7.point1.interrupt()
make interrupt for thread1. which it's Ref is Point1

→ and print: second object thread False
↳ False for thread 2 I didn't make Interrupt For thread2 so false

- then by returning to thread1:

First Object thread true
↳ I made Interrupt by so true

and So on....

* program #9

- I can implement Runnable (→ have run function), ~~because~~ not Extend because Java is not multiple inherited.

- System.out.println(getName()) → exception will happen.
  → Thread.currentthread.getName().

* Thread synchronization

Shared memory → Critical Section → thread at a time.

in multithreading → monitor by Synchronize Function.
If thread Comes on monitor → enter Critical section
and lock (by wait() method). until finish then
Unlock (by ~~[illegible]~~ notify(), notifyall()) then Back to Ready State.

* Example:

- There is Shared area as shown in figure..
  must Synchronization, wait, notify, notifAll.
- put the shared Area in critical section, so Once at a time just.
  when put packet #1 from machine A to Shared Area → Wait()
  then machine B will take it & notify, see Solution.

* by two Functions: set Shared Area() ← اكتب , get shared Area() → تاخذ منك
← لازم يكون بينهم Synchronize

* flag → (to execute A first)

* getshared area → (WAIT) من خلال ال flag ازا مش دوري بدي استنى (B)

* Set Shared area → " " " " " " " " (A)