

## **// Clienthttp.java**

```
import java.io.*;
import java.net.*;
public class Clienthttp {
    public static void main (String argv[]) {

        try {
            // Création d'une socket client
            Socket client = new Socket (argv[0], Integer.parseInt(argv[1]));

            // Création du flux reception
            PrintWriter emission = new PrintWriter( new BufferedWriter (new
            OutputStreamWriter(client.getOutputStream()), true);

            // Création du flux reception
            BufferedReader reception = new BufferedReader
            ( new InputStreamReader(client.getInputStream()));

            // Envoi d'une requete
            // emission.println ("GET /home/talantik5/recherche/zineb/www/access_fr.html HTTP/1.0\n\n");

            emission.println("GET index.html HTTP/1.0\n\n");


            // Réception du résultat ligne par ligne
            boolean continu = true;
            String ligne = reception.readLine();
            if (ligne == null) continu = false;

            else {System.out.println(ligne);

            }

            }
            catch(IOException e) {System.out.println("Erreur : " + e ); }
        }
    }

    import java.io.*;
    import java.net.*;
    import java.util.regex.*;
```

**class Application implements Runnable {**

Socket client;

Application( Socket client ) throws SocketException{  
this.client = client;  
}

// Corps de l'application

public void run() {  
try {

// Creation du flux in de lecture sur la socket

InputStream in = client.getInputStream();

BufferedReader pin = new BufferedReader (new InputStreamReader (in));

// Creation du flux pout d'envoi sur la socket

// OutputStream out = client.getOutputStream();

OutputStream out = client.getOutputStream();

PrintWriter pout = new PrintWriter (new BufferedWriter (new  
OutputStreamWriter(client.getOutputStream()), true);

// Reception de la requete HTTP dans la socket

String requete = pin.readLine();

// Impression de la requete HTTP

System.out.println( "Requete: "+requete);

// Filtrage de la requete grace a l'expression reguliere

Matcher get = Pattern.compile("GET (/?\\S\*).\*").matcher(requete);

// Traitement de la requete apres filtrage

if(get.matches()){

requete = get.group(1);

if(requete.endsWith("/") || requete.equals(""))

requete = requete + "index.html";

System.out.println("Fichier" + requete);

// Envoi du fichier html bloc par bloc

try  
{

FileInputStream fis = new FileInputStream(requete);

byte [] Bloc = new byte [64\*1024];

```
int bloc;
    for(int bloc_lu; (bloc_lu=fis.read(Bloc)) > -1;)
        out.write(Bloc, 0, bloc_lu);
    out.flush();

} catch(FileNotFoundException e) {

pout.println (" Objet non trouve"); }
} else
pout.println("requete erronee");
client.close();
} catch(IOException e) {
System.out.println("Erreur E/S" +e);}
}
}
```