ASSIGNMENT –10

DATE:04/11/24

REGISTER NO:2022503045

1.TCP FOR CAPITALIZE

CODE:

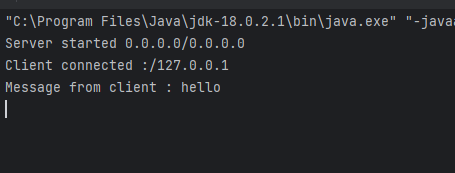
Server side:

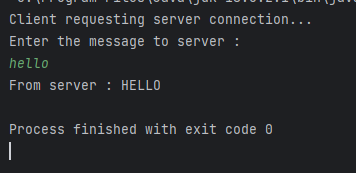
import java.io.DataInputStream;  
import java.io.DataOutputStream;  
import java.net.\*;  
  
public class Server\_3045\_TCP {  
 public static void main(String[] args){  
 try(ServerSocket ss=new ServerSocket(9100);){  
 System.*out*.println("Server started "+ss.getInetAddress());  
 while(true){  
 try(Socket socket=ss.accept()){  
 System.*out*.println("Client connected :"+socket.getInetAddress());  
 DataInputStream from\_client=new DataInputStream(socket.getInputStream());  
 DataOutputStream to\_client=new DataOutputStream(socket.getOutputStream());  
 String from=from\_client.readUTF();  
 System.*out*.println("Message from client : "+from);  
 String to=from.toUpperCase();  
 to\_client.writeUTF(to);  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
  
 }  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
 }  
}

Client side:

import java.io.DataInputStream;  
import java.io.DataOutputStream;  
import java.net.Socket;  
import java.util.Scanner;  
  
public class Client\_3045\_TCP {  
 public static void main(String[] args){  
 try(Socket s=new Socket("localhost",9100);){  
 System.*out*.println("Client requesting server connection...");  
 DataInputStream from\_server=new DataInputStream(s.getInputStream());  
 DataOutputStream to\_server=new DataOutputStream(s.getOutputStream());  
  
 Scanner sc=new Scanner(System.*in*);  
 System.*out*.println("Enter the message to server : ");  
 String to=sc.nextLine();  
  
 to\_server.writeUTF(to);  
  
 String from=from\_server.readUTF();  
 System.*out*.println("From server : "+from);  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
 }  
}

Output :





2.TCP FOR MAIL VALIDATION

CODE:

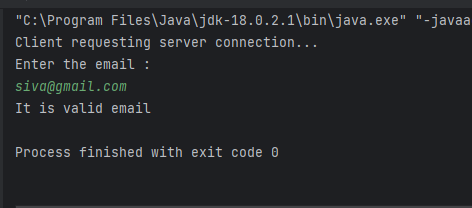
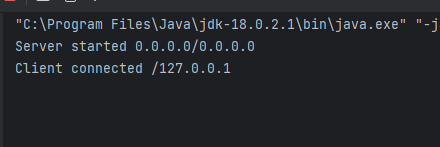
Server Side:

import java.io.DataInputStream;  
import java.io.DataOutputStream;  
import java.net.ServerSocket;  
import java.net.Socket;  
  
public class TCP\_ServerEmail\_45 {  
 public static boolean isValidEmail(String email){  
 if(email==null||email.isEmpty())  
 return false;  
 int atIndex=email.indexOf('@');  
 int dotIndex=email.lastIndexOf('.');  
 if(atIndex>0 && dotIndex>atIndex){  
 String localPart=email.substring(0,atIndex);  
 String domainPart=email.substring(atIndex+1);  
 boolean isValidLocal=*isValidLocal*(localPart);  
 boolean isValidDomain=*isValidDomain*(domainPart);  
 return isValidDomain && isValidLocal;  
 }  
 return false;  
 }  
 public static boolean isValidLocal(String localPart){  
 for(char ch:localPart.toCharArray()){  
 if(!Character.*isLetterOrDigit*(ch)||ch=='.'||ch=='\_'){  
 return false;  
 }  
 }  
 return true;  
 }  
 public static boolean isValidDomain(String domainPart){  
 for(char ch:domainPart.toCharArray()){  
 if(!Character.*isLetterOrDigit*(ch) && ch!='.'){  
 return false;  
 }  
 }  
 return !domainPart.endsWith(".") && !domainPart.startsWith(".");  
 }  
 public static void main(String[] args){  
 try(ServerSocket server=new ServerSocket(9200)){  
 System.*out*.println("Server started "+server.getInetAddress());  
 while(true){  
 try(Socket client=server.accept()){  
 System.*out*.println("Client connected "+client.getInetAddress());  
 DataInputStream from\_client=new DataInputStream(client.getInputStream());  
 DataOutputStream to\_client =new DataOutputStream(client.getOutputStream());  
 String mail=from\_client.readUTF();  
 boolean isValid=*isValidEmail*(mail);  
 to\_client.writeBoolean(isValid);  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
 }  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
 }  
}

Client side:

import java.io.DataInputStream;  
import java.io.DataOutputStream;  
import java.net.Socket;  
import java.util.Scanner;  
  
public class TCP\_ClientEmail\_45 {  
 public static void main(String[] args){  
 try(Socket s=new Socket("localhost",9200);){  
 System.*out*.println("Client requesting server connection...");  
 DataInputStream from\_server=new DataInputStream(s.getInputStream());  
 DataOutputStream to\_server=new DataOutputStream(s.getOutputStream());  
  
 Scanner sc=new Scanner(System.*in*);  
 System.*out*.println("Enter the email : ");  
 String to=sc.nextLine();  
  
 to\_server.writeUTF(to);  
  
 if(from\_server.readBoolean())  
 System.*out*.println("It is valid email");  
 else  
 System.*out*.println("It is Invalid email");  
  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
 }  
}

Output



3.UDP FOR CAPITALIZE

CODE:

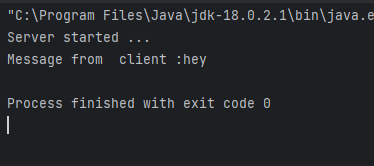
Server side:

import java.net.DatagramPacket;  
import java.net.DatagramSocket;  
  
public class UDP\_serverCap\_45 {  
  
 public static void main(String[] args){  
 try(DatagramSocket socket=new DatagramSocket(9999)){  
 System.*out*.println("Server started ...");  
 byte[] from\_client=new byte[1024];  
 byte[] to\_client=new byte[1024];  
 DatagramPacket dr=new DatagramPacket(from\_client,from\_client.length);  
 socket.receive(dr);  
 String from=new String(dr.getData(),0,dr.getLength());  
 System.*out*.println("Message from client :"+from);  
 String to=from.toUpperCase();  
 to\_client=to.getBytes();  
 DatagramPacket ds=new DatagramPacket(to\_client,to\_client.length,dr.getAddress(),dr.getPort());  
 socket.send(ds);  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
 }  
}

Client side:

import java.net.DatagramPacket;  
import java.net.DatagramSocket;  
import java.net.InetAddress;  
import java.util.Scanner;  
  
public class UDP\_clientCap\_45 {  
 public static void main(String[] args){  
 try(DatagramSocket client=new DatagramSocket()){  
 System.*out*.println("Client started..");  
 byte[] from\_server=new byte[1024];  
 byte[] to\_server=new byte[1024];  
 Scanner sc=new Scanner(System.*in*);  
 System.*out*.println("Enter the message to server :");  
 String to=sc.nextLine();  
 to\_server=to.getBytes();  
 InetAddress ip=InetAddress.*getByName*("localhost");  
 DatagramPacket dr=new DatagramPacket(to\_server,to\_server.length,ip,9999);  
 client.send(dr);  
  
 DatagramPacket ds=new DatagramPacket(from\_server,from\_server.length);  
 client.receive(ds);  
 String from=new String(ds.getData(),0,ds.getLength());  
 System.*out*.println("Message from server "+from);  
  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
 }  
}

Output :



4.UDP FOR MAIL VALIDATION

CODE:

Server side:

import java.net.DatagramPacket;  
import java.net.DatagramSocket;  
  
public class UDP\_EmailServer\_45 {  
 public static boolean isValidEmail(String email){  
 if(email==null||email.isEmpty())  
 return false;  
 int atIndex=email.indexOf('@');  
 int dotIndex=email.lastIndexOf('.');  
 if(atIndex>0 && dotIndex>atIndex){  
 String localPart=email.substring(0,atIndex);  
 String domainPart=email.substring(atIndex+1);  
 boolean isValidLocal=*isValidLocal*(localPart);  
 boolean isValidDomain=*isValidDomain*(domainPart);  
 return isValidDomain && isValidLocal;  
 }  
 return false;  
 }  
 public static boolean isValidLocal(String localPart){  
 for(char ch:localPart.toCharArray()){  
 if(!Character.*isLetterOrDigit*(ch)||ch=='.'||ch=='\_'){  
 return false;  
 }  
 }  
 return true;  
 }  
 public static boolean isValidDomain(String domainPart){  
 for(char ch:domainPart.toCharArray()){  
 if(!Character.*isLetterOrDigit*(ch) && ch!='.'){  
 return false;  
 }  
 }  
 return !domainPart.endsWith(".") && !domainPart.startsWith(".");  
 }  
 public static void main(String[] args){  
 try(DatagramSocket socket=new DatagramSocket(9999)){  
 System.*out*.println("Server started ...");  
 byte[] from\_client=new byte[1024];  
 byte[] to\_client=new byte[1024];  
 DatagramPacket dr=new DatagramPacket(from\_client,from\_client.length);  
 socket.receive(dr);  
 String from=new String(dr.getData(),0,dr.getLength());  
 System.*out*.println("mail from client :"+from);  
 boolean valid=*isValidEmail*(from);  
 String to;  
 if(valid)  
 to="Valid";  
 else  
 to="Invalid";  
  
  
 to\_client=to.getBytes();  
 DatagramPacket ds=new DatagramPacket(to\_client,to\_client.length,dr.getAddress(),dr.getPort());  
 socket.send(ds);  
 }  
 catch(Exception e){  
 System.*out*.println(e.getMessage());  
 }  
 }  
}

Client side:

import java.net.DatagramPacket;  
import java.net.DatagramSocket;  
import java.net.InetAddress;  
import java.util.Scanner;  
  
public class UDP\_EmailClient\_45 {  
 public static void main(String[] args) {  
 try (DatagramSocket client = new DatagramSocket()) {  
 System.*out*.println("Client started..");  
 byte[] from\_server = new byte[1024];  
 byte[] to\_server = new byte[1024];  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.println("Enter the mail to server :");  
 String to = sc.nextLine();  
 to\_server = to.getBytes();  
 InetAddress ip = InetAddress.*getByName*("localhost");  
 DatagramPacket dr = new DatagramPacket(to\_server, to\_server.length, ip, 9999);  
 client.send(dr);  
  
 DatagramPacket ds = new DatagramPacket(from\_server, from\_server.length);  
 client.receive(ds);  
 String from = new String(ds.getData(), 0, ds.getLength());  
 System.*out*.println("Your email is "+from);  
  
 } catch (Exception e) {  
 System.*out*.println(e.getMessage());  
 }  
 }  
}

Output :

