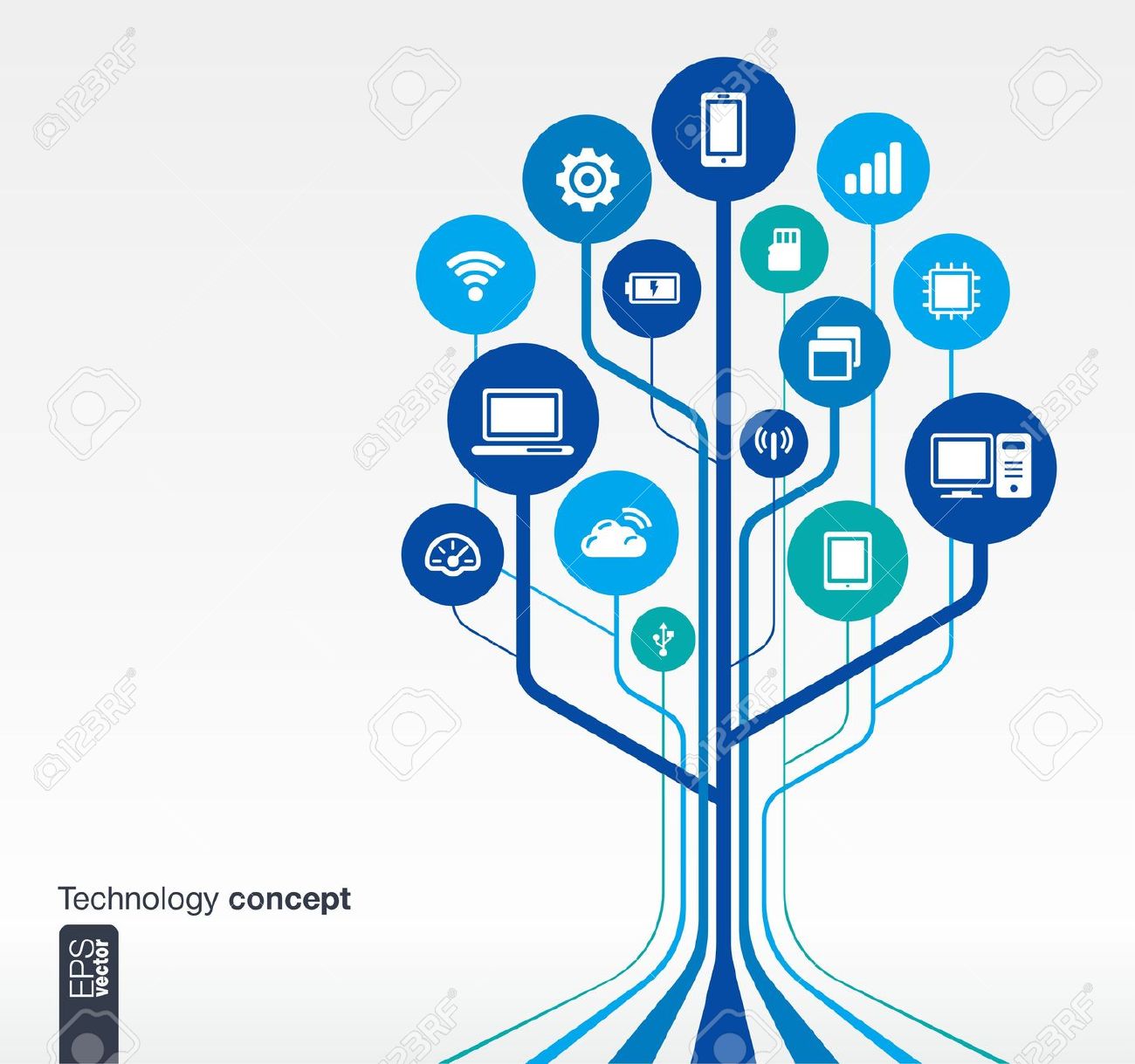
**JAVA**

**PROJECT**

NETWORK PROGRAMMING

***Semester 6 project work***

***Submitted to:***

**Dr. Aakanksha Vatsa**

***By:***

*Deepti Sharma*

*Mansi Shukla*

*Nikita Bhagat*

*Nikita Kumari*

*Shivani Tiwari*

|  |  |  |
| --- | --- | --- |
| **Index** | | |
| **S.no.** | **Program** | **Page No** |
| (i) | Certificate | (i) |
| 1. | TCP: Echo Client and Server(Iterative) | 4 |
| 2. | TCP: Daytime Client and Server(Iterative) | 7 |
| 3. | UDP: Echo Client and Server | 10 |
| 4. | UDP: Daytime Client and Server | 13 |
| 5. | Display Host Name | 15 |
| 6. | Display All IP Address | 16 |

**Certificate**

This is to certify that the project entitled

***Java Network Programming*** on ***Linux Background***

is a bona fide work done by the following students:

1. ***Deepti Sharma(140066270012)***
2. ***Mansi Shukla(140066270024)***
3. ***Nikita Bhagat(140066270031)***
4. ***Nikita Kumari(140066270032)***
5. ***Shivani Tiwari(140066270047)***

Of

***BSc(Hons). Computer Science***

and

has been carried out under my supervision

and guidance to the best of my knowledge that

the project is

**genuine.**

Signature

(Dr. Aakanksha Vatsa)

Date

**TCP**

*Implement TCP Echo Client and Server (Iterative).*

***Program:***

1. ***Server:***

*import java.net.\*;*

*import java.io.\*;*

public class EchoServer

{

public static void main(String[] args) throws IOException

{

ServerSocket serverSocket = null;

try

{

serverSocket = new ServerSocket(1007);

}

catch(IOException e)

{

System.err.println("Could not listen on port: 1007.");

System.exit(1);

}

Socket clientSocket = null;

System.out.println("Waiting for Connection....");

try

{

clientSocket = ServerSocket.accept();

}

catch(IOException e)

{

System.err.println("Accept failed.");

System.exit(1);

}

System.out.println("Connection successful");

System.out.println("Waiting for Connection....");

PrinrWriter out = new PrinrWriter(clientSocket.getOutputStream(),true);

BufferedReader in = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));

String inputLine;

while((inputLine = in.readLine()) != null)

{

System.out.println("Server: " + inputLine);

out.println(inputLine);

}

out.close();

in.close();

clientSocket.close();

serverSocket.close();

}

}

1. ***Client:***

*import java.net.\*;*

*import java.io.\*;*

public class EchoClientIterative

{

public static void main(String[] args) throws IOException

{

String serverHostname = new String("127.0.0.1");

if(args.length > 0)

serverHostname = args[0];

System.out.println("Attempting to connect to host " + serverHostname + " on port 100.");

Socket echoSocket = null;

PrinrWriter out = null;

BufferedReader in = null;

try {

echoSocket = new Socket(serverHostname,1007);

echoSocket.setSoTimeout(1500);

out = new PrinrWriter(echoSocket.getOutputStream(),true);

in = new BufferedReader(newInputStreamReader(echoSocket.getInputStream());

}

catch(UnknownHostException e)

{

System.err.println("Dont know about host: " + serverHostname);

System.exit(1);

}

catch(IOException e)

{

System.err.println("Couldn't get I/O for " + "the connection to:" + serverHostname ");

System.exit(1);

}

BufferedReader stdIn = new BufferedReader(new InputstreamReader(System.in));

String userInput;

System.out.println("Connection successful");

System.out.println("Enter 'Bye.' to terminate connection.");

System.out.println();

System.out.println("INPUT: ");

while((userInput = stdIn.readLine()) != null)

{

out.println(userInput);

System.out.println("ECHO: " + in.readLine());

if(userInput.equals("Bye."))

break;

System.out.println("INPUT: ");

}

out.close();

in.close();

stdIn.close();

echoSocket.close();

}

}

***Output:***

[u6302@linux6 tcpecho]$ javac client.java

[u6302@linux6 tcpecho]$ java server &

[1] 7769

[u6302@linux6 tcpecho]$ Waiting for Connection....

java client

Attempting to connect to host 127.0.0.1 on port 100.

Connection successful

Waiting for Connection....

Connection successful

Enter 'Bye.' to terminate connection.

INPUT:

Hello

Server: Hello

ECHO: Hello

INPUT:

You r great

Server: You r great

ECHO: You r great

*Implement TCP Daytime Client and Server (Iterative)*

***Program:***

1. ***Client***

*import java.net.\*;*

*import java.io.\*;*

public class DaytimeClient

{

public static final int SERVICE\_PORT = 13;

public static void main(String[] args)

{

//check for hostname parameter

if(args.length != 1)

{

System.out.println("Syntax - DaytimeClient host");

return;

}

//get the hostname of server

String hostname = args[0];

try

{

//get a socket to the daytime service

Socket daytime = new Socket (hostname, SERVICE\_PORT);

//set the socket option just in case server stalls

daytime.setSoTimeout(2000);

System.out.println("Connection established...");

BufferedReader reader = new BufferedReader(new InputstreamReader(daytime.getInputStream()));

System.out.println("Current day and time : " + readLine());

//close the connection

daytime.close();

}

catch(IOException e)

{

System.err.println("!!!Input/Output Error!!!" + e);

}

catch(SocketTimeoutException e)

{

System.err.println("!!!Connection Error!!!");

}

}

}

1. ***Server***

*import java.net.\*;*

*import java.nio.\*;*

public class DaytimeServer

{

public static void main(String[] args) throws java.io.IOException

{

int port = 13;

if(args.length > 0)

port = Integer.parseInt(args[0]);

//create a channel to listen for connections on.

ServerSocketChannel server = ServerSocketChannel.open();

//bind the channel to a local port.

server.socket().bind(new InetSocketAddress(port));

//get an encoder for converting strings to bytes

CharsetEncoder encoder = Charset.forName("US-ASCII").newEncoder();

for(;;)

{ //loop forever, processing client connections

//wait for a client to connect

SocketChannel client = server.accept();

//build response string, wrap, and encode to bytes

String date = new java.util.Date().toString() + "\r\n";

ByteBuffer response = encoder.encode(CharBuffer.wrap(date));

//send the response to the client and disconnect.

client.write(response);

client.close();

}

}

}

***Output:***

[u6302@linux6 tcpdaytime]$ javac server.java

[u6302@linux6 tcpdaytime]$ javac client.java

[u6302@linux6 tcpdaytime]$ java server &

[2] 9319

[u6302@linux6 tcpdaytime]$ java client 127.0.0.1

Connection established...

Current day and time : Mon Apr 10 17:55:31 IST 2017

UDP

*Implement UDP Echo Client and Server*

***Program:***

1. ***Client :***

*import java.net.\*;*

*import java.io.\*;*

public class ue\_client

{

private final static int PACKETSIZE=100;

private final static int myport=2015;

public static void main(String[] args)

{

if(args.length!=1)

{

System.out.println("Hostname Missing");

return;

}

DatagramSocket socket=null;

try

{

InetAddress host=InetAddress.getByName(args[0]);

socket=new DatagramSocket();

BufferedReader in=new BufferedReader(new InputStreamReader(System.in));

System.out.println("Enter:");

String msg=in.readLine();

byte[] data=msg.getBytes();

DatagramPacket packet=new DatagramPacket(data,data.length,host,myport);

socket.send(packet);

socket.setSoTimeout(2000);

packet.setData(new byte[PACKETSIZE]);

socket.receive(packet);

String response=new String(packet.getData());

InetAddress retip=packet.getAddress();

System.out.println("Response:"+response);

}

catch(SocketTimeoutException e)

{

System.out.println(e);

}

finally

{

if( socket!=null)

socket.close();

}

}

}

1. ***Server:***

*import java.net.\*;*

public class ue\_serv

{

private final static int PACKETSIZE=100;

private final static int myport=2015;

public static void main(String[] args)

{

try

{

DatagramSocket socket=new DatagramSocket(myport);

for(;;)

{

DatagramPacket packet =new DatagramPacket(new byte[PACKETSIZE],PACKETSIZE);

socket.receive(packet);

System.out.println(packet.getAddress()+ " "+ packet.getPort() + ":"+new String(packet.getData()));

socket.send(packet);

}

}

catch(Exception e)

{

System.out.println(e);

}

}

}

***Output:***

[u6302@linux6 udpecho]$ javac client.java

[u6302@linux6 udpecho]$ java server &

[1] 7549

[u6302@linux6 udpecho]$ Waiting for Connection....

java client

Attempting to connect to host 127.0.0.1 on port 2015.

Connection successful

Waiting for Connection....

Connection successful

Enter 'Bye.' to terminate connection.

INPUT:

Hello

Server: Hello

ECHO: Hello

INPUT:

You rnew

Server: You rnew

ECHO: You rnew

*Implement UDP Daytime Client and Server*

***Program:***

1. ***Client :***

*import java.io.\*;*

*import java.net.\*;*

public class UDP\_daytimeClient

{

private final static int PACKETSIZE=100;

private final static int port=13;

public static void main(String[] args)

{

//check the arguments

if(args.length != 1)

{

System.out.println("usage: java DatagramClient hostname");

return;

}

DatagramSocket socket=null;

try

{

//Convert the arguments first, to ensure that they are valid

InetAddress host = InetAddress.getByName(args[0]);

socket=new DatagramSocket();

socket.setSoTimeout(5000);

DatagramPacket packet = new DatagramPacket(new byte[256],1,host,port);

socket.send(packet);

packet.setLength(packet.getData().length);

socket.receive(packet);

byte[] data = packet.getData();

int length = packet.getLength();

System.out.println("\nCurrent Date and time is::" + new String(data,0,length,"latin1"));

}

catch(IOException e)

{

System.out.println(e);

}

finally

{

if(socket != null)

socket.close();

}

}

}

1. ***Server:***

*import java.io.\*;*

*import java.net.\*;*

import java.util.Date;

public class UDP\_daytimeServer

{

public static final int DEFAULT\_PORT=13;

public static void main(String[] args) throws IOException

{

DatagramSocket socket=new DatagramSocket(DEFAULT\_PORT);

DatagramPacket packet = new DatagramPacket(new byte[1],1);

while(true)

{

socket.receive(packet);

System.out.println("Received from: " + packet.getAddress() + ":" + packet.getPort());

Date date = new Date();

byte[] outBuffer = date.toString().getBytes("latin1");

packet.setData(outBuffer);

packet.setLength(outBuffer.length);

socket.send(packet);

}

}

}

***Output:***

[u6302@linux6 udaytime]$ javac client.java

[u6302@linux6 udaytime]$ javac server.java

[u6302@linux6 udaytime]$ java server &

[1] 8659

[u6302@linux6 udaytime]$ java client 127.0.0.1

Received from: /127.0.0.1:38849

Current day and time : Mon Apr 10 17:55:45 IST 2017

HOST INFORMATION

*Program to display Host name*

***Program:***

*import java.net.\*;*

*import java.io.\*;*

public class test{

public static void main(String args[]) throws Exception

{

InetAddress addr=InetAddress.getLocalHost();

String ip=addr.getHostAddress();

System.out.println("IP ADDRESS OF LOCAL HOST :"+ip);

String Host=addr.getHostName();

System.out.println("HOST NAME :"+Host);

}

}

***Output:***

[u6302@linux6 ~]$ java test 127.0.0.1

IP ADDRESS OF LOCAL HOST :127.0.0.1

HOST NAME: linux6.com

*Program to display IP addresses of the Host*

***Program:***

*import java.net.\*;*

public class localip

{

public static void main(String[] args)

{

if(args.length!=1)

{

System.out.println("Hostname missing");

return;

}

try

{

InetAddress[] add=InetAddress.getAllByName(args[0]);

System.out.println("\n The List of IP Address ");

for(int i=0;i<add.length;i++)

System.out.println(add[i]);

}

catch(UnknownHostException e)

{

System.out.println("Could not find host");

}

}

}

***Output:***

[u6302@linux6 ipaddr]$ java localip 127.0.0.1

The List of IP Address 127.0.0.1