

Date -

Experiment - 1

Java PROGRAM using InetAddress class and Socket

PROGRAM – 1

AIM :

To implement simple ping pong applicataion between two or more different machines.

PROGRAM :

```
import java.io.*;
import java.net.*;
import java.util.*;
public class Ping
{
    public static void main(String[] args) throws IOException
    {
        Scanner sc = new Scanner(System.in);
        int n = 5;

        for (int i = 0; i < n; i++)
        {
            System.out.print("Enter IP Address : ");
            String ip = sc.next();
            InetAddress inet = InetAddress.getByName(ip);
            System.out.println("Sending ping request to... " +
inet);

            if (inet.isReachable(5000))
                System.out.println("Host Reachable ");
            else
                System.out.println("Host NOT Reachable ");
        }
    }
}
```

OUTPUT:

```
Enter IP Address : 192.168.0.1
Sending ping request to... /192.168.0.1
Host Reachable
Enter IP Address : 192.168.0.101
Sending ping request to... /192.168.0.101
Host NOT Reachable
```

RESULT :

Thus the program is executed and output is obtained.

PROGRAM — 2

AIM :

To implement date and time display from client to server using Sockets.

PROGRAM :

Client.java:

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

public class DateTimeClient
{
    public static void main(String args[]) throws Exception
    {
        Socket soc=new
        Socket(InetAddress.getLocalHost(),9876);

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

        BufferedReader in=new BufferedReader(new
        InputStreamReader(soc.getInputStream() ));
        System.out.println(in.readLine());
    }
}
```

Server.java:

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

class DateTimeServer
{
    public static void main(String args[]) throws Exception
    {
        ServerSocket s=new ServerSocket(9876);
        System.out.println("Server running on port 9876...");
        System.out.println("Waiting For Connection ...");

        Socket soc=s.accept();
        System.out.println("Client Connected");

        DataOutputStream out=new
        DataOutputStream(soc.getOutputStream());
        out.writeBytes("Server Date: " + (new
        Date()).toString() + "\n");
        System.out.println("Data Sent");
    }
}
```

```
        System.out.println("Server Disconnecting");
        out.close();
        soc.close();
    }
}
```

OUTPUT:

Server	Client
Server running on port 9876	Connection established
Waiting For Connection ...	
Client Connected	Server Date: Sun Feb 24
Data Sent	12:49:52 IST 2019
Server Disconnecting	

RESULT :

Thus the program is executed and output is obtained.

PROGRAM — 3

AIM:

To write a ping pong client and server application. When a client sends a ping message to the server, the server will respond with a pong message. Other messages sent by the client can be safely dropped by the client.

PROGRAM :

Client.java:

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

public class PingPongClient
{
    public static void main(String args[])
    {
        try
        {
            Socket socket = new Socket("localhost", 9876);
            System.out.println("Connected to server");

            DataInputStream in=new
                DataInputStream(socket.getInputStream());
            DataOutputStream out=new
                DataOutputStream(socket.getOutputStream());

            Scanner sc = new Scanner(System.in);

            while(true)
            {
                System.out.print("Enter message for server :
");
                String msg = sc.nextLine();

                out.writeUTF(msg);
                if(msg.equals("Ping"))
                    System.out.println("Message Recieved : "
                                       + in.readUTF());

                if(msg.equals("exit"))
                    break;
            }
            System.out.println("Server disconnected");
            out.close();
            socket.close();
        }
    }
}
```

```

    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
}

```

Server.java:

```

import java.io.*;
import java.net.*;

public class PingPongServer
{
    public static void main(String args[])
    {
        try
        {
            ServerSocket server = new ServerSocket(9876);
            System.out.println("Server running in port
                                9876...");
            System.out.println("Waiting for client to
                                connect...");

            Socket socket=server.accept();
            System.out.println("Client connected\n\n");

            DataInputStream in=new DataInputStream(new
                BufferedInputStream(socket.getInputStream()));
            DataOutputStream out=new
                DataOutputStream(socket.getOutputStream());

            while(true)
            {
                String msg = in.readUTF();
                if(msg.equals("Ping"))
                {
                    System.out.println("Replying for Ping
                                        message");
                    out.writeUTF("Pong");
                }
                if(msg.equals("exit"))
                    break;
            }
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }
}

```

```
}  
}
```

OUTPUT:

Server	Client
Server running in port 9876...	Connected to server
Waiting for client to connect...	Enter message for server : Hello
Client connected	Enter message for server : Ping
	Message Recieved : Pong
Message Recieved : Hello	Enter message for server : exit
Replying for Ping message	
Server exiting	

RESULT :

Thus the program is executed and output is obtained.

PROGRAM — 4

AIM:

To write a socket based Java server PROGRAM that responds to client messages as follows : When it receives a message from client, it simply converts the message into all uppercase letters and sends back the same to the client. Write both client and server PROGRAMs demonstrating this.

PROGRAM :

Client.java:

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

public class UpperChatClient
{
    public static void main(String args[])
    {
        try
        {
            Socket socket = new Socket("localhost", 9876);
            System.out.println("Connected to server");

            DataInputStream in=new
                DataInputStream(socket.getInputStream());
            DataOutputStream out=new
                DataOutputStream(socket.getOutputStream());

            Scanner sc = new Scanner(System.in);

            while(true)
            {
                System.out.print("Enter message for server :
                ");
                String msg = sc.nextLine();

                out.writeUTF(msg);

                System.out.println("Message Recieved : " +
                                    in.readUTF());

                if(msg.equals("exit"))
                    break;
            }
            System.out.println("Server disconnected");
            out.close();
            socket.close();
        }
    }
}
```



```

    }
    catch(Exception e)
    {
        e.printStackTrace();
    }
}
}

```

Server.java:

```

import java.io.*;
import java.net.*;

public class UpperChatServer
{
    public static void main(String args[])
    {
        try
        {
            ServerSocket server = new ServerSocket(9876);
            System.out.println("Server running in port 9876...");
            System.out.println("Waiting for client to
                                connect...");

            Socket socket=server.accept();
            System.out.println("Client connected\n\n");

            DataInputStream in=new DataInputStream(new
                BufferedInputStream(socket.getInputStream()));
            DataOutputStream out=new
                DataOutputStream(socket.getOutputStream());

            while(true)
            {
                String msg = in.readUTF();

                System.out.println("Message recieved : "+msg);
                out.writeUTF(msg.toUpperCase());

                if(msg.equals("exit"))
                    break;
            }
        }
        catch(Exception e)
        {
            e.printStackTrace();
        }
    }
}

```

OUTPUT:

Server	Cient
Server running in port 9876...	Connected to server
Waiting for client to connect...	Enter message for server : Hello
Client connected	Message Recieved : HELLO
	Enter message for server : Voila
	Message Recieved : VOILA
Message recieved : Hello	Enter message for server : I'm
Message recieved : Voila	so cool
Message recieved : I'm so cool	Message Recieved : I'M SO COOL
Message recieved : exit	Enter message for server : exit
	Message Recieved : EXIT
	Server disconnected

RESULT :

Thus the program is executed and output is obtained.

PROGRAM — 5

AIM:

To implement Broadcasting messages in your lab.

PROGRAM :

BroadcastServer.java

```
import java.io.*;
import java.net.*;
import java.util.*;
class broadcastServer{
    public static void main(String[] args){
        try{
            DatagramSocket ds=new DatagramSocket();
            String msg;
            DatagramPacket dp;
            InetAddress
            ip=InetAddress.getByName("255.255.255.255");
            Scanner sc=new Scanner(System.in);
            byte[] buf;
            while(true){
                System.out.println("Enter the message to
                                   be broadcasted..\n");
                msg=sc.nextLine();
                buf=msg.getBytes();
                dp=new
                DatagramPacket(buf,buf.length,ip,5000);
                ds.send(dp);
                if(msg.equals("exit"))
                    break;
            }
            ds.close();
        }
        catch(Exception e){
            e.printStackTrace();
        }
    }
}
```

BroadcastClient.java

```
import java.net.*;
import java.io.*;
import java.util.*;
class broadcastClient{
    public static String toString(byte[] arr){
        String s = "";
```

```

        for (int i=0; i<arr.length && arr[i] != 0; s +=
                                                    (char)arr[i++]);
        return s;
    }
    public static void main(String[] args){
        try{
            DatagramSocket ds=new DatagramSocket(5000);
            String s1;byte[] buf;
            DatagramPacket dp;
            while(true){
                buf=new byte[1024];
                dp=new DatagramPacket(buf,buf.length);
                ds.receive(dp);
                s1=toString(buf);
                //System.out.println(buf.toString());
                if(s1.equals("exit"))
                    break;
                System.out.println(s1);
            }
            ds.close();
        }
        catch(Exception e){
            e.printStackTrace();
        }
    }
}

```

OUTPUT :

Server	Client
Enter the message to be broadcasted..	Hello World
	Ping
<p>Hello World</p> <p>Enter the message to be broadcasted..</p>	
<p>Ping</p> <p>Enter the message to be broadcasted..</p>	
exit	

RESULT :

Thus the program is executed and output is obtained.