

EX.NO: 10	Simulation of DNS
DATE:	

Aim

To simulate a Domain Name System (DNS) using User Datagram Protocol (UDP) Sockets in Java.

Theory:

The Domain Name System (DNS) is a hierarchical decentralized naming system for computers, services, or any resource connected to the Internet or a private network. UDP is a connectionless protocol that operates at the transport layer and provides a simple interface between a network application and the Internet Protocol (IP).

ALGORITHM

Algorithm for DNS Server:

- Start the program.
- Initialize arrays for hosts and corresponding IP addresses.
- Create a DatagramSocket on port 1362.
- Receive a DatagramPacket from the client.
- Extract the requested hostname from the received packet.
- Check if the hostname exists in the hosts array.
- If found, send back the corresponding IP address; otherwise, send "Host Not Found" message.
- Create a response DatagramPacket with the IP address or error message.
- Send the response packet to the client.
- Close the server socket.

Algorithm for DNS Client:

- Start the program.
- Create a DatagramSocket for communication.
- Get the IP address of the server from command line arguments or use localhost by default.
- Prompt user to enter a hostname.
- Convert the hostname to bytes and create a DatagramPacket to send to the server.
- Send the packet to the server.
- Receive the response packet from the server.
- Extract and display the IP address received from the server.
- Close the client socket.

Program

UDP DNS Server

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

public class dnsserver
{
    private static int indexOf(String[] array, String str)
    {
        str = str.trim();
        for (int i=0; i < array.length; i++)
        {
            if (array[i].equals(str)) return i;
        }
        return -1;
    }
}
```

```
public static void main(String arg[])throws IOException
{

String[] hosts = {"zoho.com", "gmail.com","google.com", "facebook.com"};

String[] ip = {"172.28.251.59", "172.217.11.5","172.217.11.14",

"31.13.71.36"}; System.out.println("Press Ctrl + C to Quit");

while (true)
{

receivedata.length);

DatagramSocket serversocket=new DatagramSocket(1362);
byte[] senddata = new byte[1021];
byte[] receivedata = new byte[1021];
DatagramPacket recvpack = new DatagramPacket(receivedata,

serversocket.receive(recvpack);

String sen = new String(recvpack.getData());

InetAddress ipaddress = recvpack.getAddress();
```

```
int port = recvpack.getPort();

String capsent;

System.out.println("Request for host " + sen);

if(indexOf (hosts, sen) != -1)

capsent = ip[indexOf (hosts, sen)];

else

capsent = "Host Not Found"; senddata = capsent.getBytes();

DatagramPacket pack = new DatagramPacket (senddata,

senddata.length,ipaddress,port);

serversocket.send(pack);

serversocket.close();

}

}

}
```

UDP DNS Client –

```
import java.io.*;

import java.net.*;

public class dnsclient
{

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

        BufferedReader br = new BufferedReader(new

        InputStreamReader(System.in));

        DatagramSocket clientsocket = new DatagramSocket();

        InetAddress ipaddress;

        if (args.length == 0)

            ipaddress = InetAddress.getLocalHost();

        else

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

        byte[] senddata = new byte[1024];

        byte[] receivedata = new byte[1024];

        int portaddr = 1362;

        System.out.print("Enter the hostname : ");
```

```
String sentence = br.readLine();

senddata = sentence.getBytes();

DatagramPacket pack = new DatagramPacket(senddata,senddata.length,

ipaddress,portaddr);


clientsocket.send(pack);

DatagramPacket recvpack =new
DatagramPacket(receivedata,receivedata.length);
clientsocket.receive(recvpack);

String modified = new String(recvpack.getData());

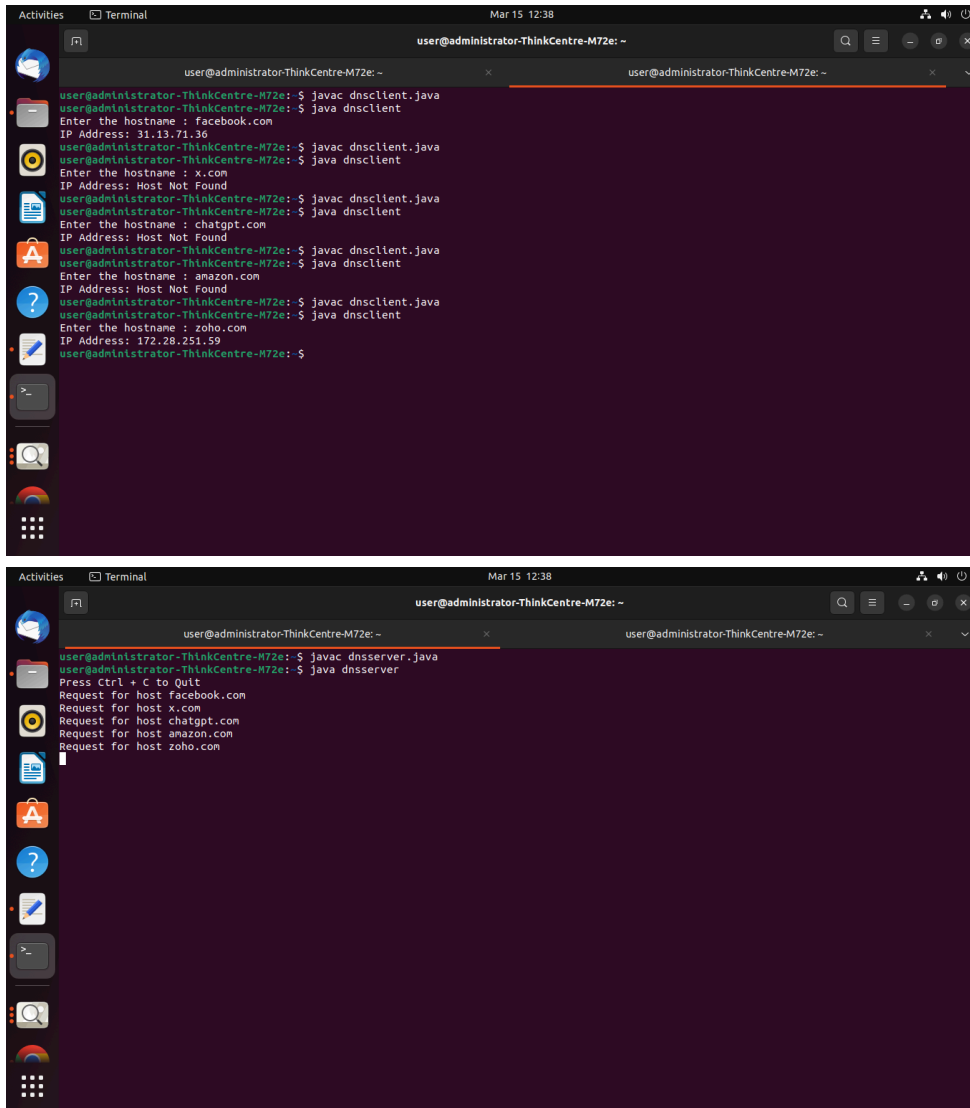
System.out.println("IP Address: " + modified);

clientsocket.close();

}

}
```

OUTPUT



```
user@administrator-ThinkCentre-M72e: ~  
user@administrator-ThinkCentre-M72e:~$ javac dnsclient.java  
user@administrator-ThinkCentre-M72e:~$ java dnsclient  
Enter the hostname : facebook.com  
IP Address: 31.13.71.36  
user@administrator-ThinkCentre-M72e:~$ javac dnsclient.java  
user@administrator-ThinkCentre-M72e:~$ java dnsclient  
Enter the hostname : x.com  
IP Address: Host Not Found  
user@administrator-ThinkCentre-M72e:~$ javac dnsclient.java  
user@administrator-ThinkCentre-M72e:~$ java dnsclient  
Enter the hostname : chatgpt.com  
IP Address: Host Not Found  
user@administrator-ThinkCentre-M72e:~$ javac dnsclient.java  
user@administrator-ThinkCentre-M72e:~$ java dnsclient  
Enter the hostname : amazon.com  
IP Address: Host Not Found  
user@administrator-ThinkCentre-M72e:~$ javac dnsclient.java  
user@administrator-ThinkCentre-M72e:~$ java dnsclient  
Enter the hostname : zoho.com  
IP Address: 172.28.251.59  
user@administrator-ThinkCentre-M72e:~$
```

```
user@administrator-ThinkCentre-M72e: ~  
user@administrator-ThinkCentre-M72e:~$ javac dnsserver.java  
user@administrator-ThinkCentre-M72e:~$ java dnsserver  
Press Ctrl + C to quit  
Request for host facebook.com  
Request for host x.com  
Request for host chatgpt.com  
Request for host amazon.com  
Request for host zoho.com
```

Conclusion

Thus the DNS application program was executed.