

subnet.java

```
import java.util.Scanner; public class
Subnet {    public static void main(String
args[]) {    Scanner sc = new
Scanner(System.in);
    System.out.print("IP address: ");
    String ip = sc.nextLine();
    String[] split_ip = ip.split("\\.");
    String[] split_bip = new String[4]; // Split binary IP
    String bip = "";

    for (int i = 0; i < 4; i++) {
        split_bip[i] = appendZeros(Integer.toBinaryString(Integer.parseInt(split_ip[i])));
bip += split_bip[i];
    }

    System.out.println("Binary Format: " + bip);

    System.out.print("Enter the number of addresses in each subnet: ");
int n = sc.nextInt();

    // Calculation of mask
    int bits = (int) Math.ceil(Math.log(n) / Math.log(2));
int mask = 32 - bits;
    System.out.println("Subnet mask = " + mask);

    int[] fbip = new int[32];

    for (int i = 0; i < 32; i++) {
        fbip[i] = bip.charAt(i) - '0'; // convert character 0,1 to integer 0,1
    }

    for (int i = 31; i > 31 - bits; i--) {
        fbip[i] &= 0; // Clear last bits
    }

    String[] fip = {"", "", "", ""};

    for (int i = 0; i < 32; i++) {
        fip[i / 8] = new String(fip[i / 8] + fbip[i]);
    }

    System.out.print("Network address is = ");

    for (int i = 0; i < 4; i++) {
        System.out.print(Integer.parseInt(fip[i], 2));
if (i != 3)
```

```

        System.out.print(".");
    }

    System.out.println();
    int[] lbip = new int[32];

    for (int i = 0; i < 32; i++) {
        lbip[i] = bip.charAt(i) - '0'; // convert character 0,1 to integer 0,1
    }

    for (int i = 31; i > 31 - bits; i--) {
        lbip[i] |= 1; // Set last bits to 1
    }

    String[] lip = {"", "", "", ""};

    for (int i = 0; i < 32; i++) {
        lip[i / 8] = new String(lip[i / 8] + lbip[i]);
    }

    System.out.print("Broadcast address is = ");

    for (int i = 0; i < 4; i++) {
        System.out.print(Integer.parseInt(lip[i], 2));
        if (i != 3)
            System.out.print(".");
    }

    System.out.println();
}

static String appendZeros(String s) {
String temp = "00000000";
    return temp.substring(s.length()) + s;
}
}

```

OUTPUT:

```

PS D:\> java Subnet
IP address: 172.23.45.0
Binary Format: 10101100000010111001011010000000
Enter the number of addresses in each subnet: 230
Subnet mask = 24
Network address is = 172.23.45.0
Broadcast address is = 172.23.45.255
PS D:\> |

```

DNSServer.java

```
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");
    DatagramSocket serversocket = new DatagramSocket(1362);
    while (true) {
      byte[] receivedata = new byte[1021];
      DatagramPacket recvpack = new DatagramPacket(receivedata, receivedata.length);
      serversocket.receive(recvpack);
      String sen = new String(recvpack.getData(), 0, recvpack.getLength());
      InetAddress ipaddress = recvpack.getAddress();      int port =
      recvpack.getPort();
      String capsent;
      System.out.println("Request for host " + sen);
      int index = indexOf(hosts, sen);      if (index != -1)
      capsent = ip[index];      else
      capsent = "Host Not Found";
      byte[] senddata = capsent.getBytes();
      DatagramPacket pack = new DatagramPacket(senddata, senddata.length, ipaddress, port);
      serversocket.send(pack);
    }
  }
}
```

DNSClient.java

```
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.getByNames(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:

```

PS D:\> java DNSServer
Press Ctrl + C to Quit
Request for host google.com
|

```

```

PS D:\> java DNSClient
Enter the hostname: google.com
IP Address: 172.217.11.14
PS D:\> |

```