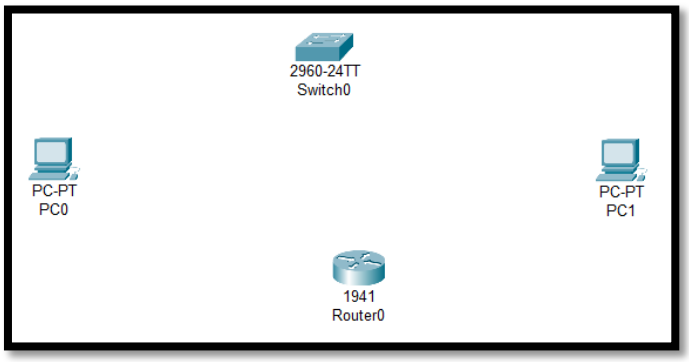
Shri G.S Institute of Technology & Science

Computer Networks

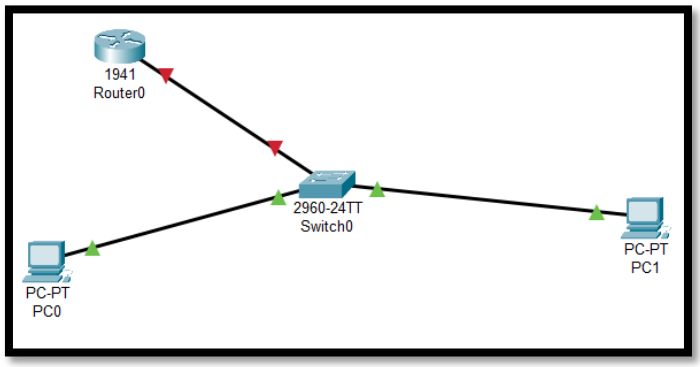
Assignment 4 – INDEX

Ques.3) Basic Device Configuration Add a Router, Switch, and PCs to the workspace. Assign IP addresses to PCs. Configure a Router's interfaces with IP addresses. Set hostnames for the devices.

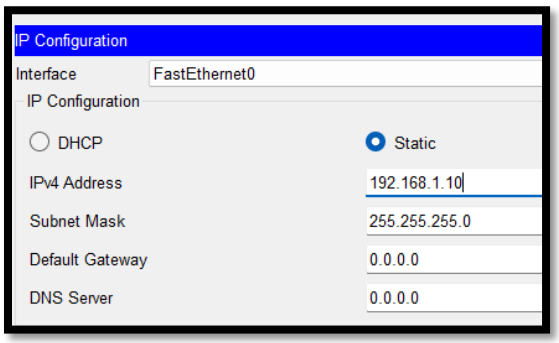
Ans.) Step 1: Add Devices to Workspace
Add a Router, Switch, and two PCs to the workspace from the device toolbar.



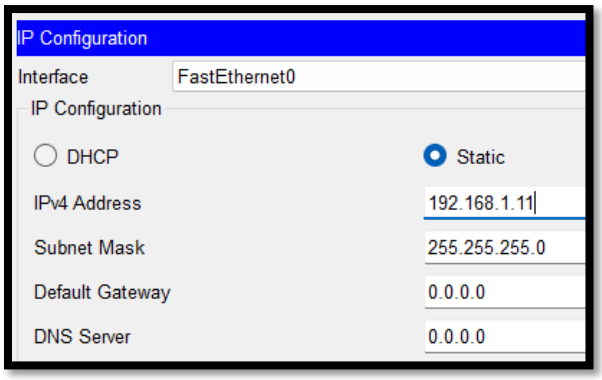
Step 2: Connect Devices with Cables
Use Copper Straight-Through cables to connect:
• PC0 to Switch
• PC1 to Switch
• Switch to Router



Step 3: Assign IP Addresses to PCs
Click on each PC > Desktop > IP Configuration:
• PC0: IP Address: 192.168.1.10
Subnet Mask: 255.255.255.0



• PC1: IP Address: 192.168.1.11
Subnet Mask: 255.255.255.0



Step 4: Configure Router Interface

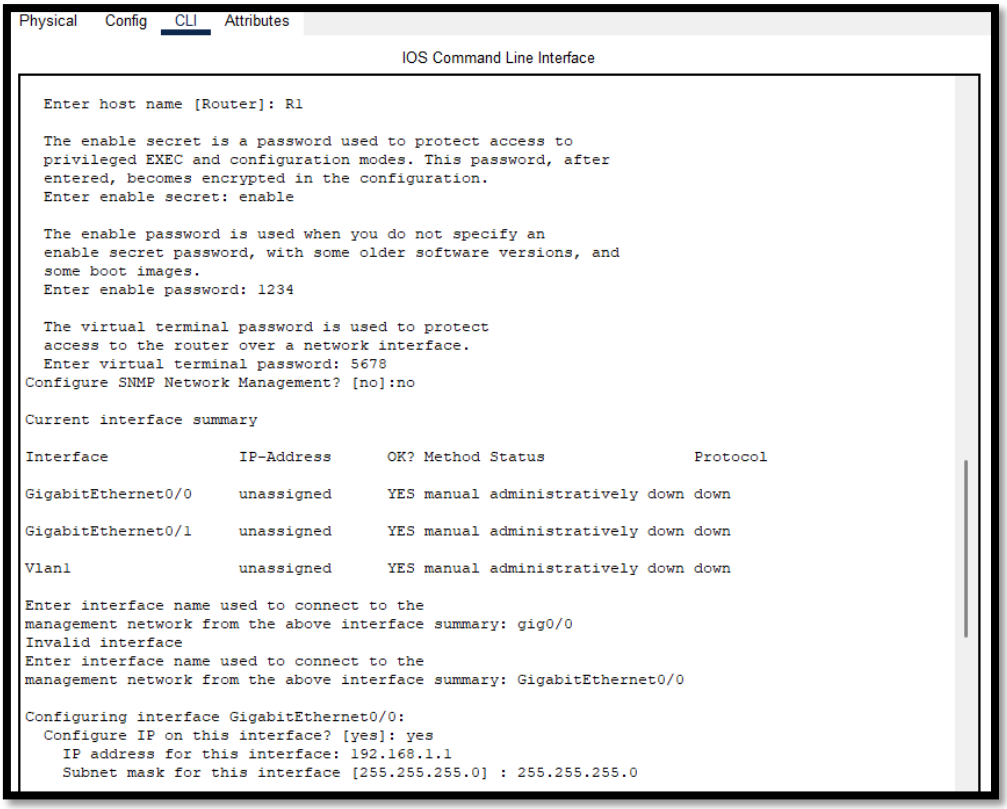
• Click Router > CLI and enter:

enable configure terminal hostname R1 interface gig0/0

ip address 192.168.1.1 255.255.255.0

no shutdown

exit



Step 5: Set Hostnames

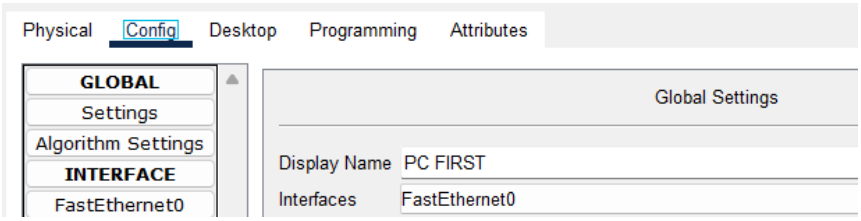
• Router: Already set above as R1

• Switch: CLI > enable > configure terminal > hostname S1

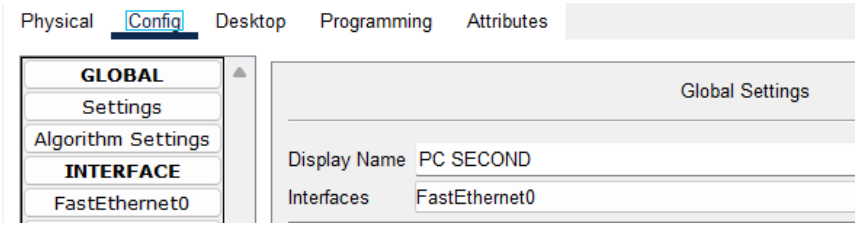
```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S1
```

• PCs: Go to Config tab > Settings > Change Display Name

• PC0 change to PC FIRST



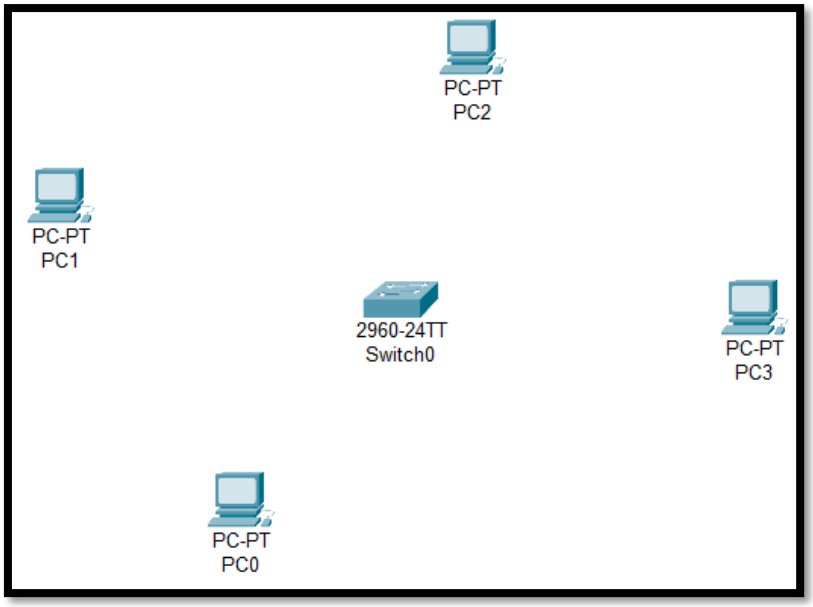
• PC1 change to PC SECOND



Ques.4) Configuring VLANs. Add a switch and multiple PCs.Create VLANs and assign ports. Configure trunk ports between switches.Verify VLANs with show vlan brief.

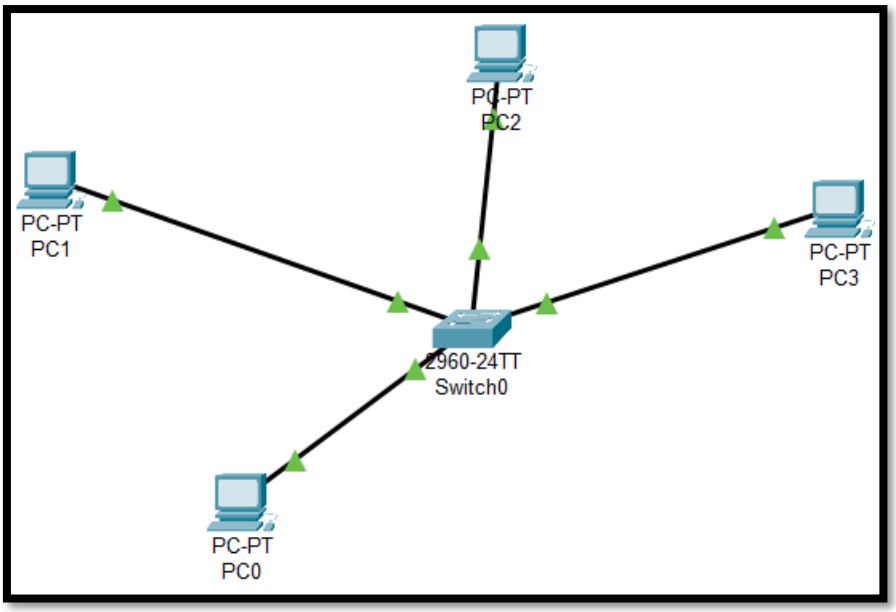
Ans.) Step 1: Add Devices to Workspace

• Add at least one Switch (e.g., Switch0) and multiple PCs (e.g., PC0, PC1, PC2, PC3) to the workspace.



Step 2: Connect PCs to the Switch

• Use Copper Straight-Through cables to connect each PC to a different FastEthernet port on the switch (e.g., Fa0/1 to Fa0/4).



Step 3: Create VLANs

• Click on the switch > CLI and enter:

enable configure terminal vlan 10 name HR

exit

vlan 20 name IT

exit

```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name HR
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#name IT
Switch(config-vlan)#exit
```

Step 4: Assign VLANs to Switch Ports

• Assign ports to VLANs: interface range fa0/1 – 2 switchport mode access switchport access vlan 10

exit

interface range fa0/3 – 4 switchport mode access switchport access vlan 20

exit

```
S!(config)#interface fastethernet0/1
S!(config-if)#switchport mode access
S!(config-if)#switchport access vlan 10
S!(config-if)#exit

S!(config)#interface FastEthernet1/1
S!(config-if)#switchport mode access
S!(config-if)#switchport access vlan 10
S!(config-if)#exit
S!(config)#interface FastEthernet2/1
S!(config-if)#switchport mode access
S!(config-if)#switchport access vlan 20
S!(config-if)#exit
S!(config)#interface FastEthernet3/1
S!(config-if)#switchport mode access
S!(config-if)#switchport access vlan 20
S!(config-if)#exit
S!(config)#
```

Step 5: Save Configure

```
S!#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

Step 6: Verify VLAN Configuration

• Use the following command to check VLANs:

show vlan brief

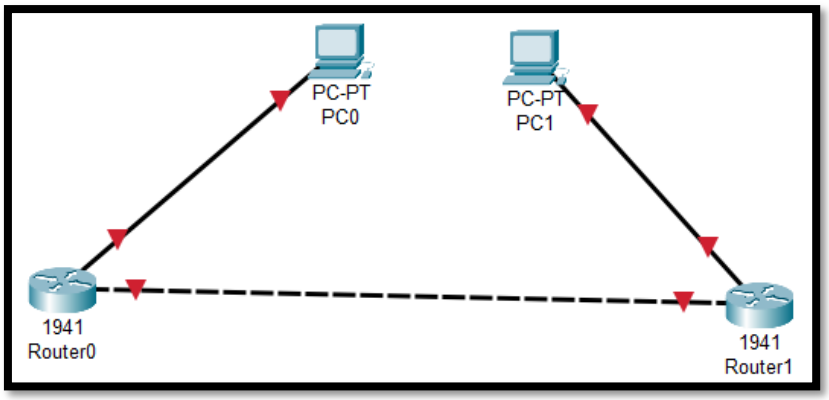
```
S!#show vlan brief

VLAN Name                              Status    Ports
---- --------------------------------- --------- -------------
1    default                           active    Fa4/1, Fa5/1
10   HR                                active    Fa0/1, Fa1/1
20   IT                                active    Fa2/1, Fa3/1
1002 fddi-default                      active
1003 token-ring-default                active
1004 fddinet-default                   active
1005 trnet-default                     active
```

Ques.5) Configuring Static Routing Use two routers and connect them with a network. Assign IP addresses and subnet masks. Configure static routes using the ip route command. Test connectivity between devices using ping.

Ans.) Step 1: Network Setup

• Use two routers (Router0 and Router1) and connect them via a crossover cable. Connect a PC to each router using a straight-through cable.



Step 2: IP Addressing Scheme

Device	Interface	IP Address	Subnet Mask
Router0	FastEthernet0/0	192.168.1.1	255.255.255.0
Router0	Serial0/0/0	10.0.0.1	255.255.255.0
Router1	Serial0/0/0	10.0.0.2	255.255.255.0
Router1	FastEthernet0/0	192.168.2.1	255.255.255.0
PC0	IP	192.168.1.2	255.255.255.0
PC1	IP	192.168.2.2	255.255.255.0

Step 3: Assign IP to PC0 & PC1

→ Click PC0 → Desktop → IP Configuration

• IP Address: 192.168.1.2

• Subnet Mask: 255.255.255.0

• Default Gateway: 192.168.1.1

→ Click PC1 → Desktop → IP Configuration

• IP Address: 192.168.2.2

• Subnet Mask: 255.255.255.0

• Default Gateway: 192.168.2.1

Step 4: Configure Router0

• Go to Router0 CLI:

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#ip address 10.0.0.1 255.255.255.252
Router(config-if)#clock rate 64000
This command applies only to DCE interfaces
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

```
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 192.168.2.0 255.255.255.0 10.0.0.2
Router(config)#exit
```

Step 5: Configure Router1

• Go to Router1 CLI:

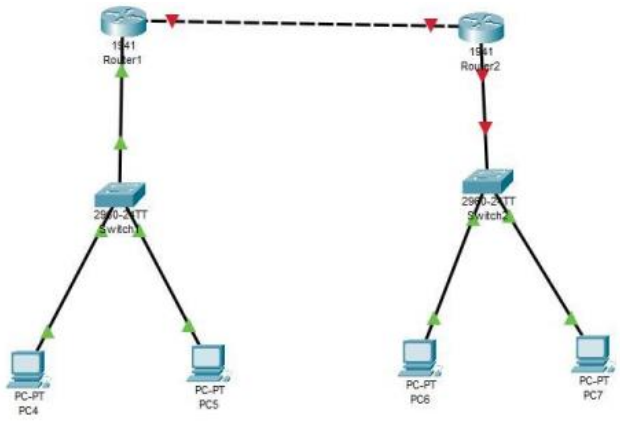
```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#ip address 10.0.0.2 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Router#configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
Router(config)#ip route 192.168.1.0 255.255.255.0
10.0.0.1
Router(config)#exit
```

Step 6: Test Connectivity

• Go to PC0 → Command Prompt

• Go to PC1 → Command Prompt

Ques.6) Configuring dynamic routing. Use multiple routers and assign IP addresses and configure RIP/OSPF



Basic Setup Before Dynamic Routing

Assume:

- 3 Routers (R1, R2, R3)
- PCs connected to each router
- Use serial or fast Ethernet connections between routers

- Use IP addresses in different subnets

A. Configuring RIP (Routing Information Protocol)

Step 1: Assign IP Addresses to Interfaces

For each router (example for R1):

```
Router> enable
Router# configure terminal
Router(config)# interface FastEthernet0/0
Router(config-if)# ip address 192.168.1.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# interface Serial0/0/0
Router(config-if)# ip address 10.0.0.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# exit
```

(Repeat for R2 and R3 with different IPs)

Step 2: Enable RIP Routing Protocol

```
Router(config)# router rip
Router(config-router)# version 2
Router(config-router)# no auto-summary
Router(config-router)# network 192.168.1.0
Router(config-router)# network 10.0.0.0
Router(config-router)# exit
```

(Do the same for R2, R3 with their networks)

Example RIP Networks:

- R1:
  - 192.168.1.0
  - 10.0.0.0
- R2:
  - 192.168.2.0
  - 10.0.0.0
  - 10.0.1.0
- R3:
  - 192.168.3.0
  - 10.0.1.0

B. Configuring OSPF (Open Shortest Path First)

Step 1: Assign IPs to Interfaces (Same as RIP) Step 2: Enable OSPF

```
Router(config)# router ospf 1
Router(config-router)# network 192.168.1.0 0.0.0.255 area 0
Router(config-router)# network 10.0.0.0 0.0.0.255 area 0
Router(config-router)# exit
```

- Replace network addresses according to your topology.
- The wildcard mask 0.0.0.255 = subnet mask 255.255.255.0

Test Connectivity

On each PC:

ping If routing is correctly configured, pings should succed.

Shri G.S Institute of Technology & Science

Computer Networks

Assignment 5 – INDEX

1.Write a Java program for TCP client-server chat (one way communication).

Server Code (TCPServer.java)

```
import java.io.*;
import java.net.*;
public class TCPServer {
   public static void main(String[] args) {
      int port = 1234;
      try (ServerSocket serverSocket = new ServerSocket(port)) {
         System.out.println("Server is listening on port " + port);
         Socket socket = serverSocket.accept();
         System.out.println("Client connected.");
         InputStream input = socket.getInputStream();
         BufferedReader reader = new BufferedReader(new InputStreamReader(input));
         String message;
         while ((message = reader.readLine()) != null) {
            System.out.println("Client: " + message);      }
         socket.close();
         System.out.println("Connection closed.");
      } catch (IOException ex) {
         System.out.println("Server exception: " + ex.getMessage());
         ex.printStackTrace();
      }   }}
```

Client Code (TCPClient.java)

```
import java.io.*;
import java.net.*;
public class TCPClient {
   public static void main(String[] args) {
      String hostname = "localhost";
      int port = 1234;
      try (Socket socket = new Socket(hostname, port)) {
         OutputStream output = socket.getOutputStream();
         PrintWriter writer = new PrintWriter(output, true);
```

```
        BufferedReader consoleReader = new BufferedReader(new InputStreamReader(System.in));
        String text;
        System.out.println("Enter messages to send to the server (type 'exit' to quit):");
        while (true) {
            System.out.print("You: ");
            text = consoleReader.readLine();
            if ("exit".equalsIgnoreCase(text)) {
                break;   }
            writer.println(text);    }
        socket.close();
        System.out.println("Disconnected from server.");
    } catch (UnknownHostException ex) {
        System.out.println("Server not found: " + ex.getMessage());
    } catch (IOException ex) {
        System.out.println("I/O error: " + ex.getMessage()); }  }}
```

2.Write a Java program for UDP server and receiver .

UDP Receiver (UDPServer.java)

```
import java.net.*;
public class UDPServer {
    public static void main(String[] args) {
        int port = 1234;
        try (DatagramSocket socket = new DatagramSocket(port)) {
            System.out.println("UDP Server is listening on port " + port);
            byte[] buffer = new byte[1024];
            while (true) {
                DatagramPacket packet = new DatagramPacket(buffer, buffer.length);
                socket.receive(packet);
                String message = new String(packet.getData(), 0, packet.getLength());
                System.out.println("Received from client: " + message);
                if (message.equalsIgnoreCase("exit")) {
                    System.out.println("Server shutting down.");
                    break;    }    }
```

```
    } catch (Exception ex) {
        System.out.println("Server error: " + ex.getMessage());
    }  }}
```

UDP Sender (UDPClient.java)

```
import java.net.*;
import java.io.*;
public class UDPClient {
  public static void main(String[] args) {
    String hostname = "localhost";
    int port = 1234;
    try (DatagramSocket socket = new DatagramSocket()) {
      BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
      String message;
      System.out.println("Type messages to send to the server (type 'exit' to quit):");
      while (true) {
        System.out.print("You: ");
        message = reader.readLine();
        byte[] buffer = message.getBytes();
        InetAddress address = InetAddress.getByName(hostname);
        DatagramPacket packet = new DatagramPacket(buffer, buffer.length, address, port);
        socket.send(packet);
        if (message.equalsIgnoreCase("exit")) {
          break;
        } }
      System.out.println("Client disconnected.");
    } catch (Exception ex) {
      System.out.println("Client error: " + ex.getMessage());
    }  }}
```

3.Write a Java program to get a website source code using URL .

```
import java.net.*;
import java.io.*;
public class WebsiteSourceCode {
```

```
public static void main(String[] args) {
    String websiteURL = "https://example.com";
    try {
        URL url = new URL(websiteURL);
        URLConnection connection = url.openConnection();
        BufferedReader reader = new BufferedReader(
            new InputStreamReader(connection.getInputStream())
        );
        String line;
        System.out.println("Website Source Code of " + websiteURL + ":\n");
        while ((line = reader.readLine()) != null) {
            System.out.println(line);  }
        reader.close();
    } catch (IOException e) {
        System.out.println("Error fetching website source code: " + e.getMessage());
    }  }}
```

```
Website Source Code of https://example.com:

<!doctype html>
<html>
<head>
    <title>Example Domain</title>

    <meta charset="utf-8" />
    <meta http-equiv="Content-type" content="text/html; charset=utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <style type="text/css">
    body {
        background-color: #f0f0f2;
        margin: 0;
        padding: 0;
        font-family: -apple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "Helvetica Neue", Helvetica
, Arial, sans-serif;

    }
    div {
        width: 600px;
        margin: 5em auto;
        padding: 2em;
        background-color: #fdfdff;
        border-radius: 0.5em;
        box-shadow: 2px 3px 7px 2px rgba(0,0,0,0.02);
    }
    a:link, a:visited {
        color: #38488f;
        text-decoration: none;
    }
    @media (max-width: 700px) {
        div {
            margin: 0 auto;
            width: auto;
        }
```

4.Write a Java program for multi-client chat server .

```
import java.io.*;
import java.net.*;
import java.util.*;
public class ChatServer {
    private static final int PORT = 1234;
```

```
    private static Set<ClientHandler> clientHandlers = Collections.synchronizedSet(new HashSet<>());
    public static void main(String[] args) {
        System.out.println("Chat server started on port " + PORT + "...");
        try (ServerSocket serverSocket = new ServerSocket(PORT)) {
            while (true) {
                Socket clientSocket = serverSocket.accept();
                System.out.println("New client connected.");
                ClientHandler clientHandler = new ClientHandler(clientSocket);
                clientHandlers.add(clientHandler);
                new Thread(clientHandler).start();    }
        } catch (IOException ex) {
            System.out.println("Server exception: " + ex.getMessage());       }    }
    static void broadcast(String message, ClientHandler sender) {
        synchronized (clientHandlers) {
            for (ClientHandler client : clientHandlers) {
                if (client != sender) {
                    client.sendMessage(message);
                }   }   }
    static void removeClient(ClientHandler clientHandler) {
        clientHandlers.remove(clientHandler);
    }}
class ClientHandler implements Runnable {
    private Socket socket;
    private PrintWriter out;
    private BufferedReader in;
    public ClientHandler(Socket socket) {
        this.socket = socket;}
    public void run() {
        try {
            out = new PrintWriter(socket.getOutputStream(), true);
            in = new BufferedReader(new InputStreamReader(socket.getInputStream()));
            String message;
```

```
            while ((message = in.readLine()) != null) {
                System.out.println("Received: " + message);
                ChatServer.broadcast(message, this); }
        } catch (IOException e) {
            System.out.println("Client disconnected.");
        } finally {
            try {
                socket.close();
            } catch (IOException e) {
                e.printStackTrace();   }
            ChatServer.removeClient(this);}  }
    void sendMessage(String message) {
        out.println(message);    }}
```

5.Write a Java program for TCP echo server.

TCP Echo Server (EchoServer.java)

```
import java.io.*;
import java.net.*;
public class EchoServer {
    public static void main(String[] args) {
        int port = 1234;
        try (ServerSocket serverSocket = new ServerSocket(port)) {
            System.out.println("Echo server started. Waiting for a client...");
            Socket socket = serverSocket.accept();
            System.out.println("Client connected.");
            BufferedReader in = new BufferedReader(
                new InputStreamReader(socket.getInputStream())
            );
            PrintWriter out = new PrintWriter(socket.getOutputStream(), true);
            String received;
            while ((received = in.readLine()) != null) {
                System.out.println("Received: " + received);
                out.println("Echo: " + received);
```

```
            if (received.equalsIgnoreCase("exit")) {
                break;      }   }
        socket.close();
        System.out.println("Connection closed.");
    } catch (IOException e) {
        System.out.println("Server exception: " + e.getMessage());
    }}}
```

TCP Echo Client (EchoClient.java)

```
import java.io.*;
import java.net.*;
public class EchoClient {
   public static void main(String[] args) {
      String hostname = "localhost";
      int port = 1234;
      try (Socket socket = new Socket(hostname, port);
           BufferedReader userInput = new BufferedReader(new InputStreamReader(System.in));
           PrintWriter out = new PrintWriter(socket.getOutputStream(), true);
           BufferedReader in = new BufferedReader(new InputStreamReader(socket.getInputStream()))
      ) {
         System.out.println("Connected to Echo Server. Type messages (type 'exit' to quit):");
         String inputLine;
         while ((inputLine = userInput.readLine()) != null) {
            out.println(inputLine);
            String response = in.readLine();
            System.out.println(response);
            if ("exit".equalsIgnoreCase(inputLine)) {
               break;
            }  }
         socket.close();
         System.out.println("Disconnected.");
      } catch (IOException ex) {
         System.out.println("Client exception: " + ex.getMessage());
```

```
    }}}
```

6.Write a Java program to display local host information .

```
import java.net.*;


public class LocalHostInfo {
   public static void main(String[] args) {
      try {
         InetAddress localHost = InetAddress.getLocalHost();
         System.out.println("Local Host Name : " + localHost.getHostName());
         System.out.println("Local IP Address: " + localHost.getHostAddress());
         System.out.println("Full Info       : " + localHost.toString());
      } catch (UnknownHostException e) {
         System.out.println("Unable to retrieve local host information.");
         e.printStackTrace();
      }}}
```

```
Local Host Name : Check
Local IP Address: 172.17.0.63
Full Info       : Check/172.17.0.63
```

7.Write a Java program find ip address of a website.

```
import java.net.*;
public class WebsiteIPFinder {
   public static void main(String[] args) {
      String website = "www.google.com"; // Replace with any website
      try {
         InetAddress inetAddress = InetAddress.getByName(website);
         System.out.println("Website: " + website);
         System.out.println("IP Address: " + inetAddress.getHostAddress());
      } catch (UnknownHostException e) {
         System.out.println("Unable to find IP address for: " + website);
         e.printStackTrace();
      } }}
```

```
Website: www.google.com
IP Address: 142.250.192.36
```

8.Write a Java program to check internet connectivity.

```
import java.io.IOException;
import java.net.Socket;
import java.net.UnknownHostException;

public class InternetCheck {
    public static void main(String[] args) {
        String host = "www.google.com";
        int port = 80;
        System.out.println("Checking internet connectivity...");
        try (Socket socket = new Socket(host, port)) {
            System.out.println("Internet is connected.");
        } catch (UnknownHostException e) {
            System.out.println("Host unreachable. No internet connection.");
        } catch (IOException e) {
            System.out.println("Cannot connect to the internet.");
        }   }}
```