­­­­­

|  |
| --- |
|  |

**CE257-DATA COMMUNICATION AND NETWORKING**

**CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY**

**DEVANG PATEL INSTITUTE OF ADVANCE TECHNOLOGY & RESEARCH**

**Department of Computer Science & Engineering**

**PRACTICAL - 1**

**AIM:**

|  |
| --- |
| **Illustration of various networking commands:**   * **Ping (types of responses)** * **tracert/traceroute** * **Netstat** * **Arp** * **ipconfig/ifconfig** * **nslookup** |

**THEORY:**

**1. Ping (types of responses):**

* ping is a command-line utility, available on virtually any operating system with network connectivity, that acts as a test to see if a networked device is reachable.
* The ping command sends a request over the network to a specific device. A successful ping results in a response from the computer that was pinged back to the originating computer.
* **ping Command:**
* **ping** [**-t**] [**-a**] [**-n** count] [**-l** size] [**-f**] [**-i** TTL] [**-v** TOS] [**-r** count] [**-s** count] [**-w** timeout] [**-R**] [**-S** srcaddr] [**-p**] [**-4**] [**-6**] target [**/?**]

**2. Tracert:**

* The tracert command is a Command Prompt command that's used to show several details about the path that a packet takes from the computer or device you're on to whatever destination you specify
* **tracert Command:**
* **tracert** [**-d**] [**-h** *MaxHops*] [**-w** *TimeOut*] [**-4**] [**-6**] *target* [**/?**]

**3. Netstat**

* The netstat [command](https://www.lifewire.com/what-is-a-command-2625828), meaning network statistics, is a [Command Prompt command](https://www.lifewire.com/list-of-command-prompt-commands-4092302) used to display very detailed information about how your computer is communicating with other computers or network devices.
* Specifically, the netstat command can show details about individual network connections.

## netstat Command:

* **netstat** [**-a**] [**-b**] [**-e**] [**-f**] [**-n**] [**-o**] [**-p** protocol] [**-r**] [**-s**] [**-t**] [**-x**] [**-y**] [time\_interval] [**/?**]

**4. Arp**

* The ARP commands to view, display, or modify the details/information in an ARP table/cache.
* The ARP cache or table has the dynamic list of IP and MAC addresses of those devices to which your computer has communicated recently in a local network. The purpose of maintaining an [ARP table](https://www.javatpoint.com/arp-table) is that when you want to communicate with another device, your device does not need to send the [ARP request](https://www.javatpoint.com/arp-request) for the MAC address of that device.
* The [ARP](https://www.javatpoint.com/address-resolution-protocol) commands also helps to find out the duplicate [IP](https://www.javatpoint.com/ip-full-form) address and invalid entries in an ARP table/cache.
* **Arp Command:**
* arp -a

**5. Ipconfig**

* **ipconfig**is a console application designed to run from the Windows command prompt. This utility allows you to get the [IP address information of a Windows computer](https://www.lifewire.com/finding-ip-address-817565). It also allows some control over your network adapters, IP addresses (DHCP-assigned specifically), even your DNS cache. i**pconfig**replaced the older winipcfg utility.
* **ipconfig Command:**
* ipconfig

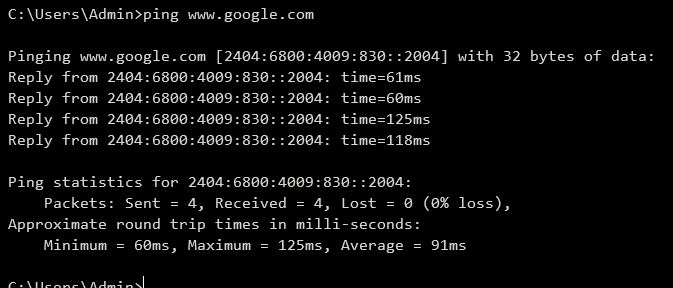
**6. nslookup**

* The **Nslookup** command line command allows you to displays information that you can use to diagnose Domain Name System (DNS) infrastructure. **Nslookup** can be used directly from the command prompt or as a shell.
* **nslookup Command:**
* nslookup

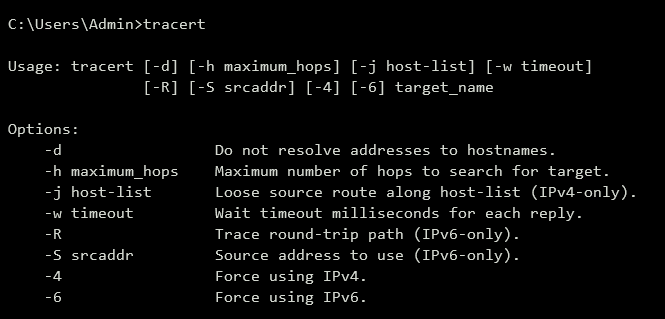
**PRACTICAL OUTPUT:**

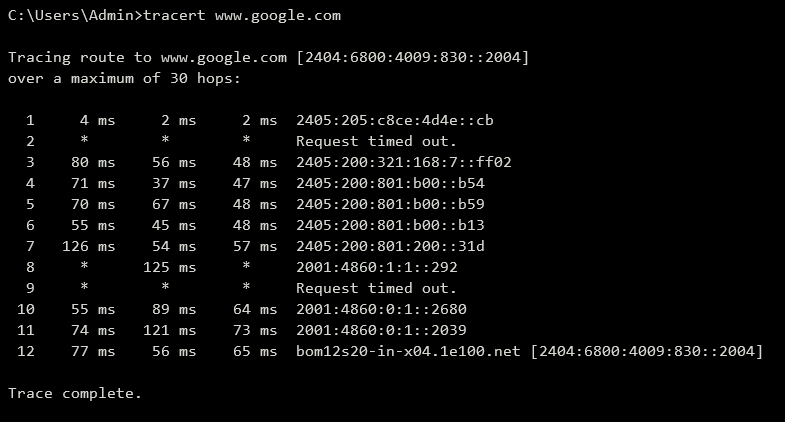
* **Ping Command Output**

****

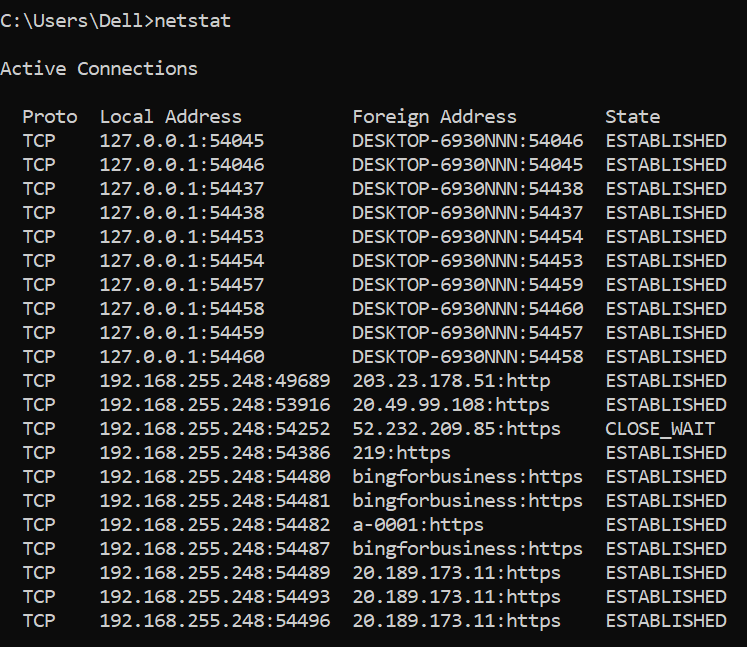
****

* **tracert Command Output**



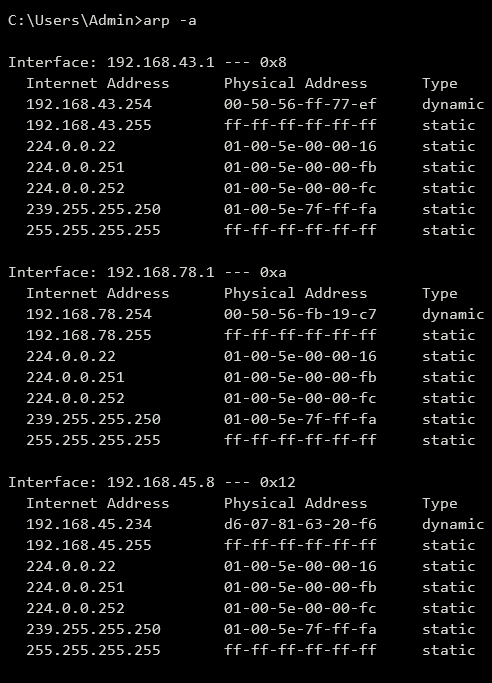
****

* **Netstat Command Output**

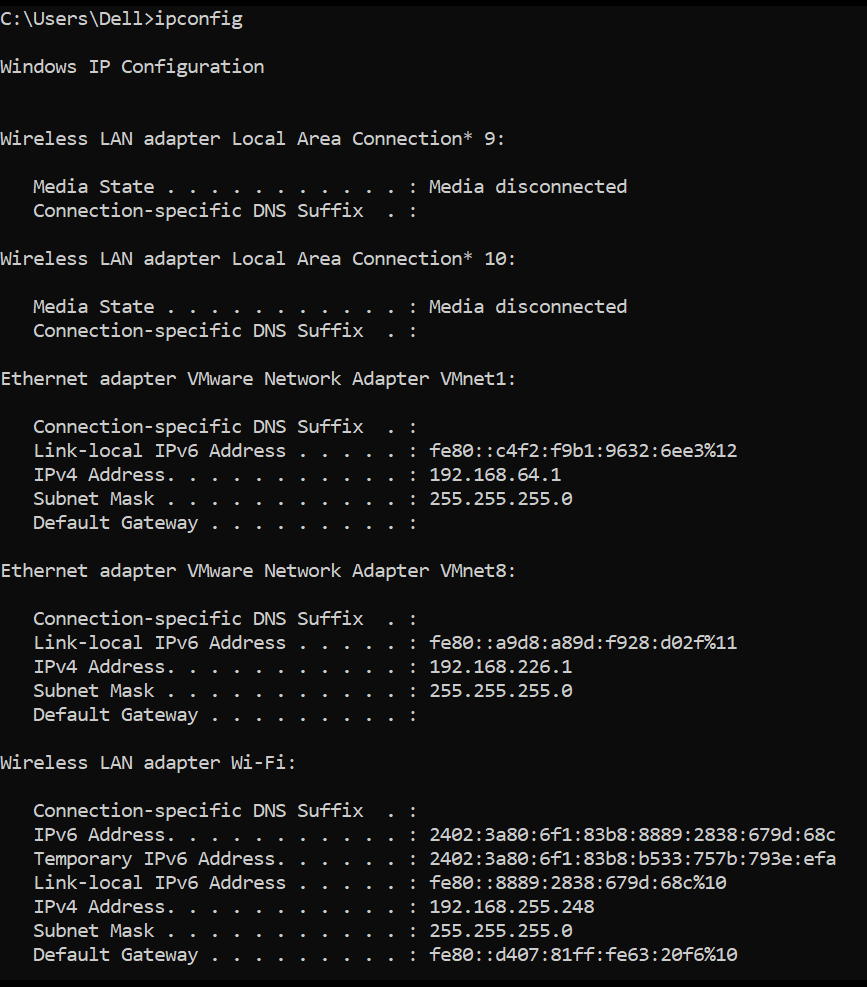
****

* **Arp Command Output**

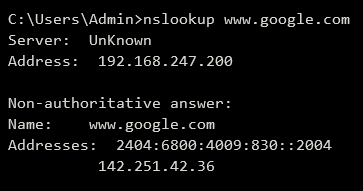
****



* **Ipconfig Command Output**

****

* **nslookup Command Output**



**CONCLUSION:**

* In this Practical we learnt about various Networking Command Uses and Practical Output.

**PRACTICAL - 2**

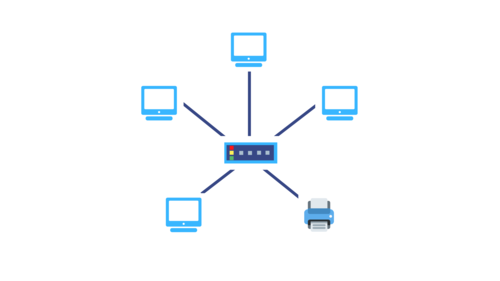
**AIM:**

|  |
| --- |
| **An organization is having size of 10 employees. The PC of all the employees are connected with each other in a single network using a switch. Create the topology and configure the devices in such a way** |

**messages can be sent to each system using cisco packet tracer.**

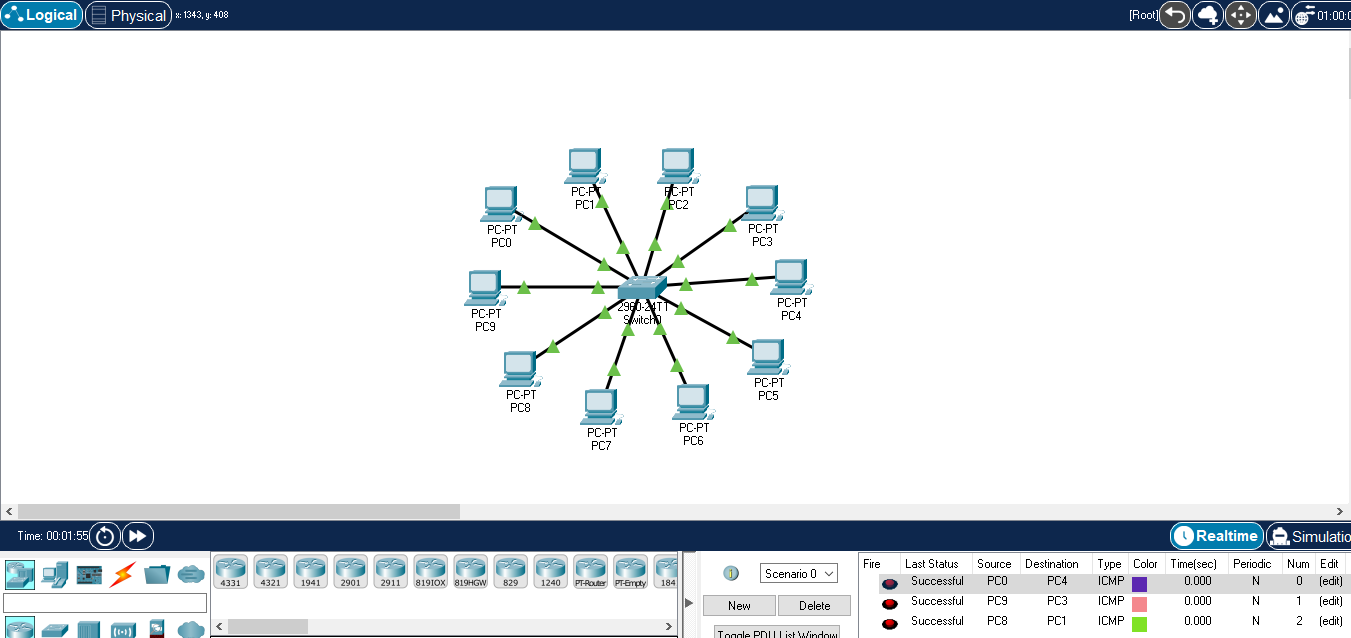
**THEORY:**

* **Star Topology:**
* A star may be a topology for a Local Area Network (LAN) during which all nodes are individually connected to a central connection point, sort of a hub or a switch. A star takes more cable than e.g. a bus, but the benefit is that if a cable fails, just one node is going to be brought down. Each device within the network is connected to a central device called hub. If one device wants to send data to another device, it’s to first send the info to hub then the hub transmits that data to the designated device.



* **Advantages of Star Topology**
* It is very reliable – if one cable or device fails then all the others will still work
* It is high-performing as no data collisions can occur
* Less expensive because each device only need one I/O port and wishes to be connected with hub with one link.
* Easier to put in
* Robust in nature
* Easy fault detection because the link are often easily identified.
* No disruptions to the network when connecting or removing devices.
* Each device requires just one port i.e. to attach to the hub.
* If N devices are connected to every other in star, then the amount of cables required to attach them is N. So, it’s easy to line up.
* **Disadvantages of Star Topology**
* Requires more cable than a linear bus.
* If the connecting network device (network switch) fails, nodes attached are disabled and can’t participate in network communication.
* More expensive than linear bus topology due to the value of the connecting devices (network switches)
* If hub goes down everything goes down, none of the devices can work without hub.
* Hub requires more resources and regular maintenance because it’s the central system of star.
* Extra hardware is required (hubs or switches) which adds to cost
* Performance is predicated on the one concentrator i.e. hub.

**PRACTICAL OUTPUT:**

****

**CONCLUSION:**

* In This Practical We Learnt Concept of Star Topology and Perform a Practical.

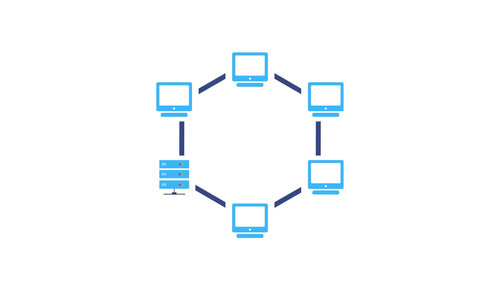
**PRACTICAL - 3**

**AIM:**

**Bank of Baroda at Changa has implemented the loan department. All the computers of loan department are connected in closed loop format. Create the topology and configure the devices to analysis the network in cisco packet tracer.**

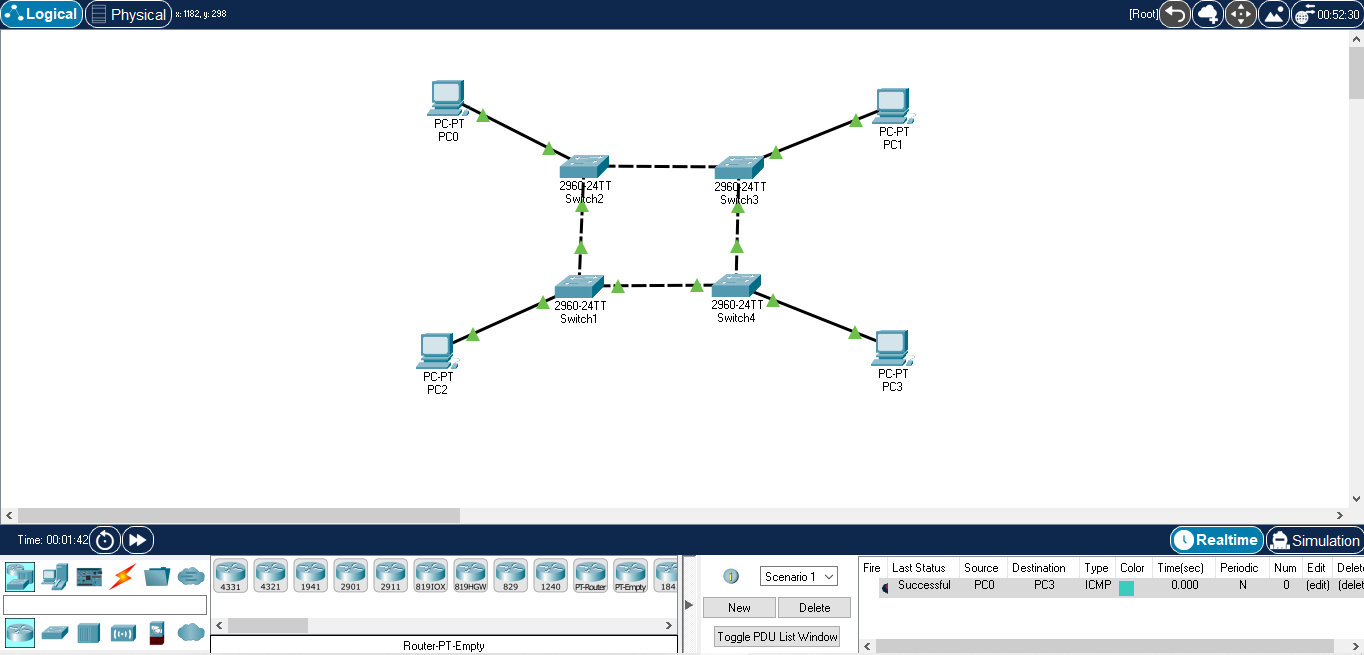
**THEORY:**

* **Ring Topology**
* A **ring topology** is a [network](https://www.computerhope.com/jargon/n/network.htm) configuration where device connections create a circular [data](https://www.computerhope.com/jargon/d/data.htm) path. Each networked device is connected to two others, like points on a circle. Together, devices in a ring topology are referred to as a **ring network**.
* In a ring network, [packets](https://www.computerhope.com/jargon/p/packet.htm) of data travel from one device to the next until they reach their destination. Most ring topologies allow packets to travel only in one direction, called a **unidirectional** ring network. Others permit data to move in either direction, called **bidirectional**.



* **Advantages of Ring Topology**
* In this data flows in one direction which reduces the chance of packet collisions.
* In this topology additional workstations can be added after without impacting performance of the network.
* Equal access to the resources.
* There is no need of server to control the connectivity among the nodes in the topology.
* It is cheap to install and expand.
* Minimum collision.
* Speed to transfer the data is very high in this type of topology.
* Due to the presence of token passing the performance of ring topology becomes better than bus topology under heavy traffic.
* Easy to manage.
* Ring network is extremely orderly organized where every device has access to the token and therefore the opportunity to transmit.
* **Disadvantages of Ring topology**
* Due to the Uni-directional Ring, a data packet (token) must have to pass through all the nodes.
* If one workstation shuts down, it affects whole network or if a node goes down entire network goes down.
* It is slower in performance as compared to the bus topology
* It is Expensive.
* Addition and removal of any node during a network is difficult and may cause issue in network activity.
* Difficult to troubleshoot the ring.
* In order for all the computer to communicate with each other, all computer must be turned on.
* Total dependence in on one cable.
* They were not Scalable.

**PRACTICAL OUTPUT:**

****

**CONCLUSION:**

* In This Practical We Learnt The Concept of Ring Topology and Perform a Practical.

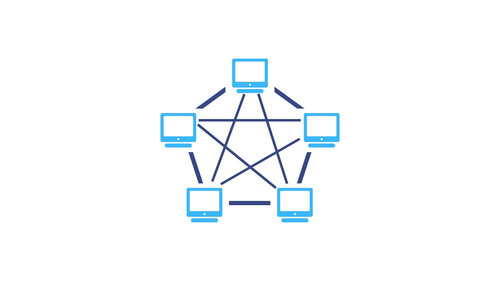
**PRACTICAL - 4**

**AIM:**

**An organization has configured all the systems in such a way that each system, has the direct connection to other system. Create the topology and analysis the network in cisco packet tracer.**

**THEORY:**

* **Mesh Topology**
* A mesh topology is a network setup where each computer and network device is interconnected with one another. This topology setup allows for most transmissions to be distributed even if one of the connections goes down. It is a topology commonly used for wireless networks.

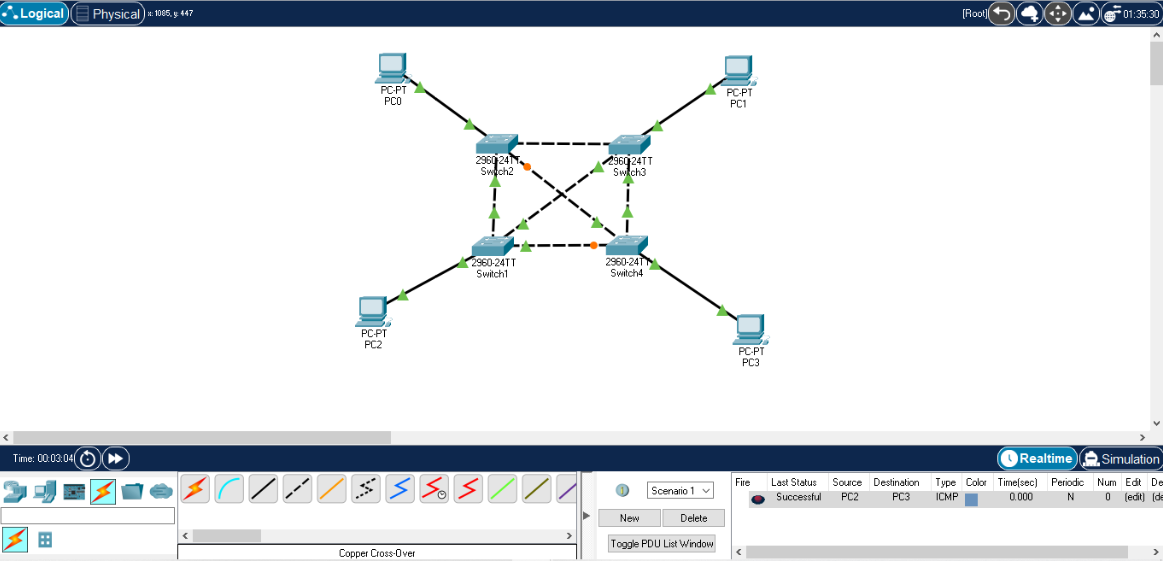


* There are two types of Mesh Topologies:

1. **Fully-connected Mesh Topology**
2. **Partially-connected Mesh Topology**

* **Full Mesh Topology :**
* All the nodes within the network are connected with every other If there are n number of nodes during a network, each node will have an n-1 number of connections. A full mesh provides an excellent deal of redundancy, but because it is prohibitively expensive to implement, it’s usually reserved for network backbones.
* **Partial Mesh Topology :**
* The partial mesh is more practical as compared to the full mesh. In a partially connected mesh, all the nodes aren’t necessary to be connected with one another during a network. Peripheral networks are connected using partial mesh and work with a full-mesh backbone in tandem.
* **Advantages of Mesh Topology**
* Failure during a single device won’t break the network.
* There is no traffic problem as there is a dedicated point to point links for every computer.
* Fault identification is straightforward.
* This topology provides multiple paths to succeed in the destination and tons of redundancy.
* It provides high privacy and security.
* Data transmission is more consistent because failure doesn’t disrupt its processes.
* Adding new devices won’t disrupt data transmissions.
* This topology has robust features to beat any situation.
* A mesh doesn’t have a centralized authority.
* **Disadvantages of Mesh Topology**
* It’s costly as compared to the opposite network topologies i.e. star, bus, point to point topology.
* Installation is extremely difficult in the mesh.
* Power requirement is higher as all the nodes will need to remain active all the time and share the load.
* Complex process.
* The cost to implement mesh is above other selections.
* There is a high risk of redundant connections.
* Each node requires a further utility cost to think about.
* Maintenance needs are challenging with a mesh.

**PRACTICAL OUTPUT:**

****

**CONCLUSION:**

* In This Practical We Learnt The Concept of Mesh Topology and Perform a Practical.

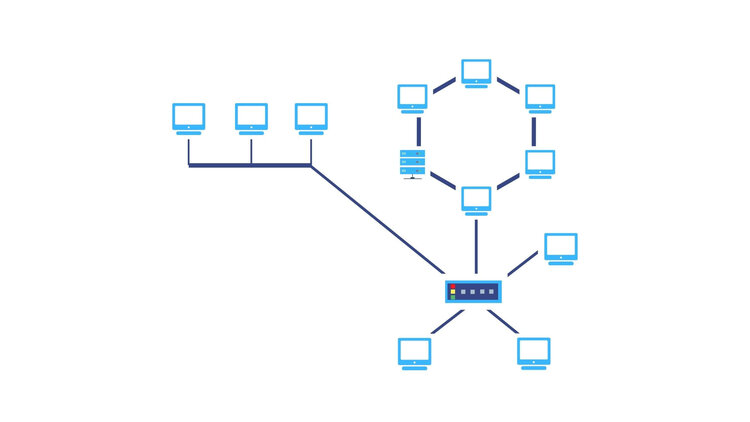
**PRACTICAL - 5**

**Aim:**

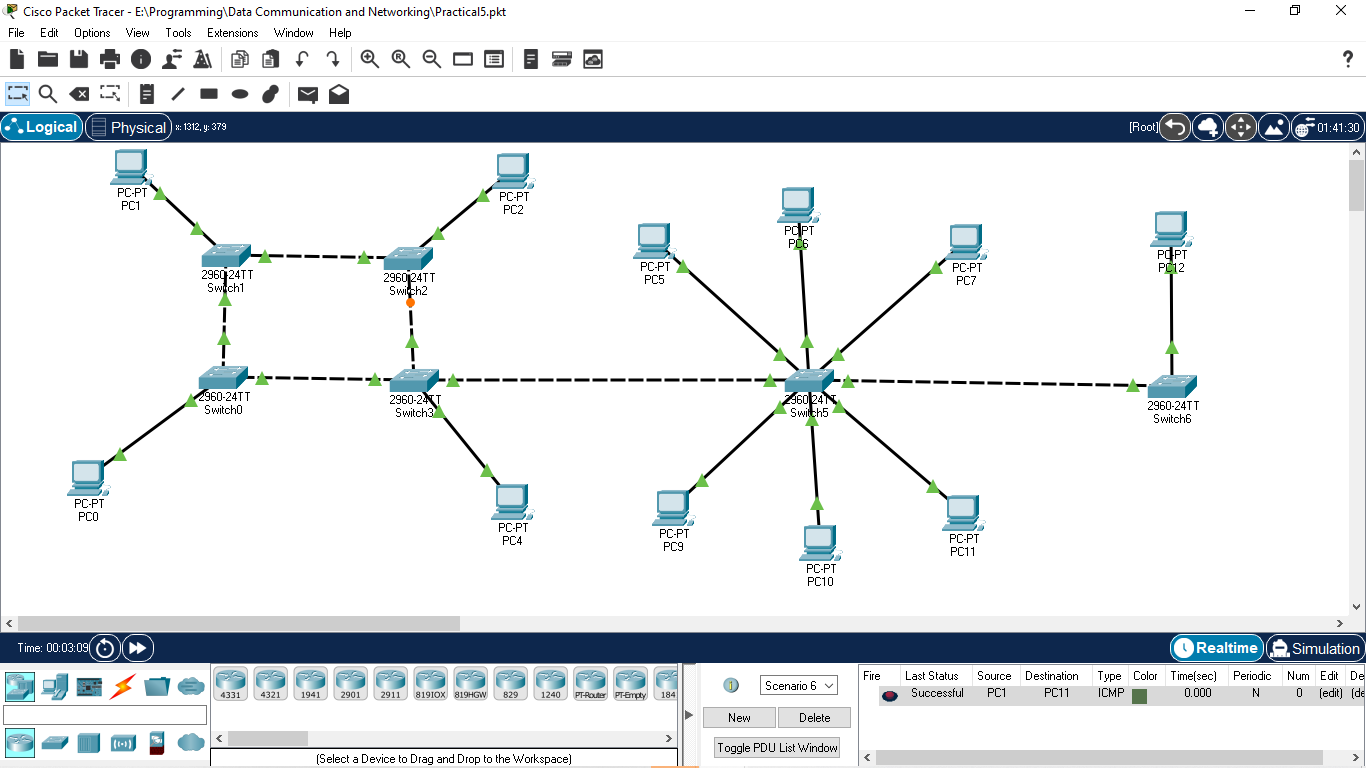
**Bank of Baroda has implemented the new environment in which all the departments i.e. loan, cash, check is connected in a bus topology. The structure of Loan department is in closed loop format while the Cash department is in star topology. Create and configure the network architecture of bank in cisco packet tracer**

**THEORY:**

* **Hybrid Topology:**
* Hybrid topology is an interconnection of two or more basic network topologies, each of which contains its own nodes. The resulting interconnection allows the nodes in a given basic topology to communicate with other nodes in the same basic topology as well as those in other basic topologies within the hybrid topology.
* **Advantages of Hybrid Topology :**
* This type of topology combines the benefits of different types of topologies in one topology.
* Can be modified as per requirement.
* It is extremely flexible.
* It is very reliable.
* It is easily scalable as Hybrid networks are built in a fashion which enables for easy integration of new hardware components.
* Error detecting and trouble shooting is easy.
* Handles large volume of traffic.
* It is used for create large network.
* **Disadvantages of Hybrid Topology :**
* It is a type of network expensive.
* Design of a hybrid network is very complex.
* There is change hardware in order to connect topology with another topology.
* Usually hybrid architectures are usually larger in scales so they requires a lot of cables in installation process.
* Hubs which are used to connect two distinct networks, are very costly. And hubs are different from usual hubs as they need to be intelligent enough to work with different architectures.
* Installation is a difficult process.



**PRACTICAL OUTPUT:**



**CONCLUSION:**

* In this practical, I learned how to make hybrid topology on Cisco Packet Tracer and also about its characteristics.

**PRACTICAL - 6**

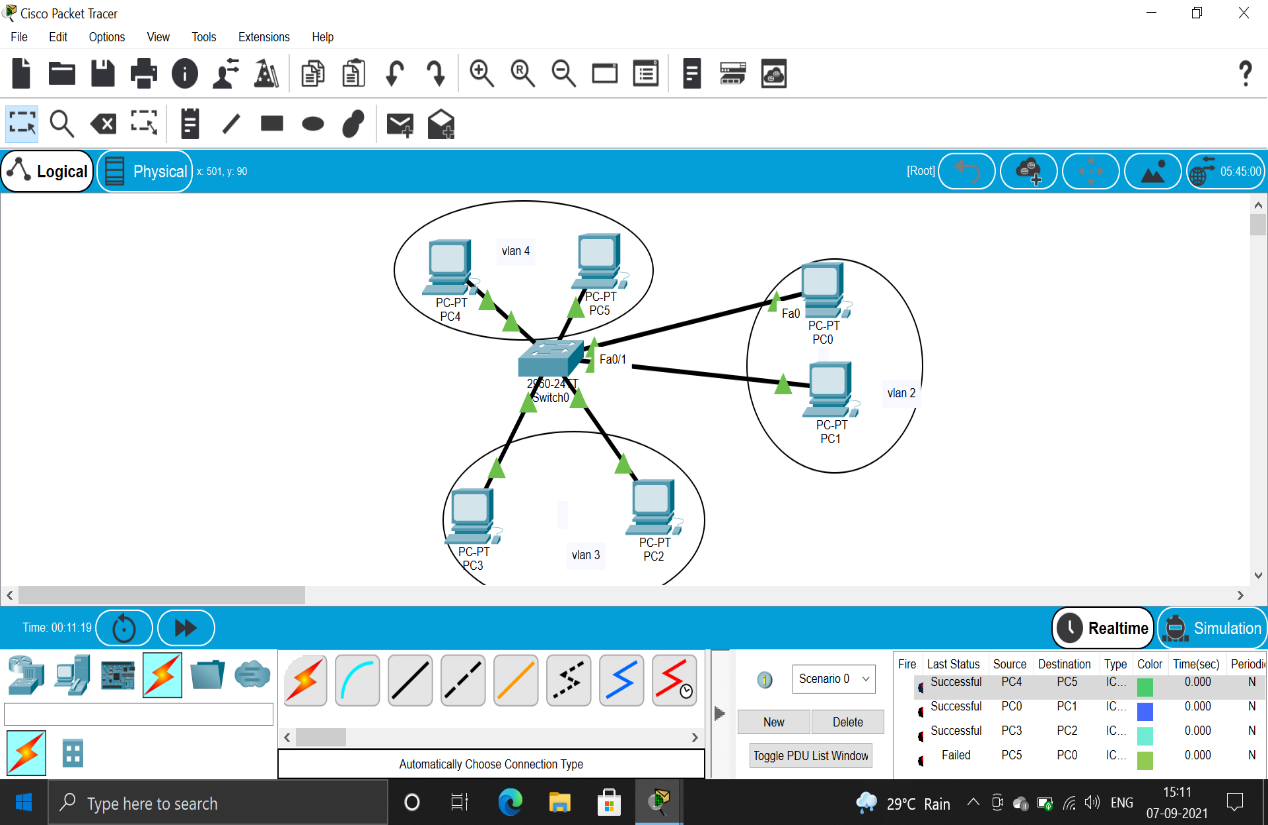
**AIM:**

**An organization works on IT projects. It has mainly 3 departments i.e. php, .net and android. CEO of that organization wants to configure a single network but virtually divided into 3 department in such a way that the packets can travel or broadcasted within the same department only. Demonstrate the configuration of such network in cisco packet tracer.**

**THEORY:**

* **Virtual LAN(Local Area Network):**
* Virtual local area networks, or VLANs, have become important as network complexity has exceeded the capacity of typical local area networks (LANs). Originally, a LAN connected a group of computers and associated devices to a server via cables in a shared physical location (hence the term “local”). Many LANs now connect devices via wireless internet, rather than Ethernet, although most LANs use a combination of both connectivity types. Over time, organizations have grown in their networking needs, requiring solutions that enable networks to grow in size, flexibility, and complexity.
* VLANs circumvent the physical limitations of a LAN through their virtual nature, allowing organizations to scale their networks, segment them to increase security measures, and decrease network latency.

**PRACTICAL OUTPUT:**



**CONCLUSION:**

* In this practical, I learned Virtual Local Area Networks and how to configure and implement them on Cisco Packet Tracer.

**PRACTICAL - 7**

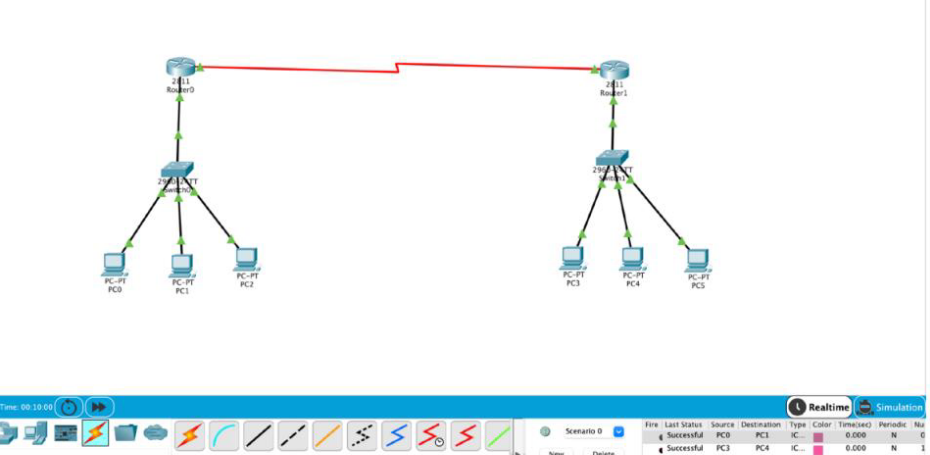
**AIM:**

**Demonstrate the simple network configuration with a router that connects two local area network (LAN) segments using cisco packet tracer.**

**THEORY:**

* **Local Area Network:**
* A LAN is a computer network that consists of access points, cables, routers, and switches that enable devices to connect to web servers and internal servers within a single building, campus, or home network, and to other LANs via Wide Area Networks (WAN) or Metropolitan Area Network (MAN). Devices on a LAN, typically personal computers and workstations, can share files and be accessed by each other over a single Internet connection.
* A router assigns IP addresses to each device on the network and facilitates a shared Internet connection between all the connected devices. A network switch connects to the router and facilitates communication between connected devices, but does not handle Local Area Network IP configuration or sharing Internet connections. Switches are ideal tools for increasing the number of LAN ports available on the network.
* **Router:**
* The router is a physical or virtual internetworking device that is designed to receive, analyze, and forward data packets between computer networks. A router examines a destination IP address of a given data packet, and it uses the headers and forwarding tables to decide the best way to transfer the packets.

**PRACTICAL OUTPUT:**

****

**CONCLUSION:**

* In this practical, I learned about the use of router and characteristics of Local Area Network. I learned how to use a router to inter connect two or more different LANs.

**PRACTICAL - 8**

**AIM:**

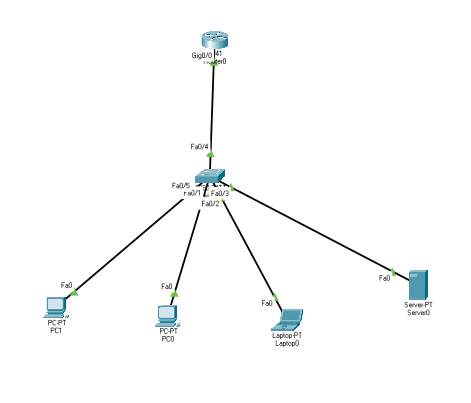
**Configure the client server architecture to access the resource from HTTP Server using the domain name. Also configure the DHCP server to provide IP addresses to the PCs connected in a network using cisco packet tracer**.

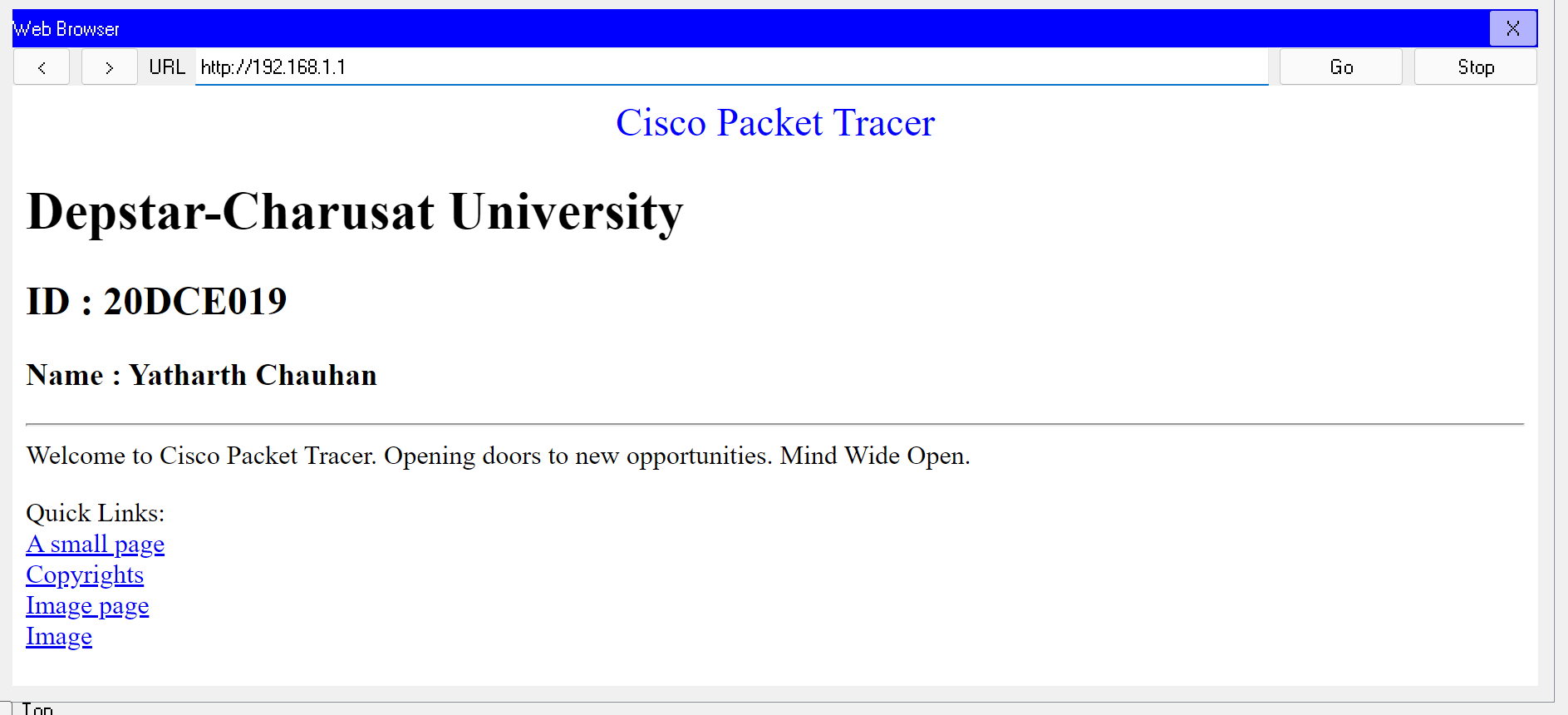
**THEORY:**

**HTTP Server:** The web server or HTTP server is a server software (or hardware dedicated to running a server software) that implements the request/response model using the World Wide Web and the HTTP (protocol.) o The HTTP or Hypertext Transfer Protocol is an application layer protocol that is used to virtually transmit files and other data on the World Wide Web, whether they’re HTML files, image files, query results, or anything else. o In simpler words, o The HTTP is a universally-spoken language with a simple grammar that “translates” the communication between the client and the server by specifying how the information must be requested and how the responses are formed.

**DCHP:** Dynamic Host Configuration Protocol (DHCP) is a network management protocol used to automate the process of configuring devices on IP networks, thus allowing them to use network services such as DNS, NTP, and any communication protocol based on UDP or TCP.

**PRACTICAL OUTPUT:**

****

****

**CONCLUSION:**

* In this practical I learned about DHCP(Dynamic Host Configuration Protocol) and how to configure devices with DHCP.

**PRACTICAL - 9**

**AIM:**

**Implement Client-server mechanism using socket programming in Java.**

**THEORY:**

* **Socket Programming:**
* A socket is a communications connection point (endpoint) that you can name and address in a network. Socket programming shows how to use socket APIs to establish communication links between remote and local processes.
* The processes that use a socket can reside on the same system or different systems on different networks. Sockets are useful for both stand-alone and network applications. Sockets allow you to exchange information between processes on the same machine or across a network, distribute work to the most efficient machine, and they easily allow access to centralized data. Socket application program interfaces (APIs) are the network standard for TCP/IP. A wide range of operating systems support socket APIs. i5/OS sockets support multiple transport and networking protocols. Socket system functions and the socket network functions are thread safe.
* Programmers who use Integrated Language Environment® (ILE) C can refer to this topic collection to develop socket applications. You can also code to the sockets API from other ILE languages, such as RPG.

**Server Side Code:**

#include <unistd.h>

#include <stdio.h>

#include <sys/socket.h>

#include <stdlib.h>

#include <netinet/in.h>

#include <string.h>

#define PORT 8080

*int* main(*int* *argc*, *char* const \**argv*[])

{

*int* server\_fd, new\_socket, valread;

*struct* sockaddr\_in address;

*int* opt = 1;

*int* addrlen = sizeof(address);

*char* buffer[1024] = {0};

*char* \*hello = "Hello from server";

    if ((server\_fd = socket(AF\_INET, SOCK\_STREAM, 0)) == 0)

    {

        perror("socket failed");

        exit(EXIT\_FAILURE);

    }

    if (setsockopt(server\_fd, SOL\_SOCKET, SO\_REUSEADDR | SO\_REUSEPORT,

                   &opt, sizeof(opt)))

    {

        perror("setsockopt");

        exit(EXIT\_FAILURE);

    }

    address.sin\_family = AF\_INET;

    address.sin\_addr.s\_addr = INADDR\_ANY;

    address.sin\_port = htons(PORT);

    if (bind(server\_fd, (*struct* sockaddr \*)&address,

             sizeof(address)) < 0)

    {

        perror("bind failed");

        exit(EXIT\_FAILURE);

    }

    if (listen(server\_fd, 3) < 0)

    {

        perror("listen");

        exit(EXIT\_FAILURE);

    }

    if ((new\_socket = accept(server\_fd, (*struct* sockaddr \*)&address,

                             (*socklen\_t* \*)&addrlen)) < 0)

    {

        perror("accept");

        exit(EXIT\_FAILURE);

    }

    valread = read(new\_socket, buffer, 1024);

    printf("%s\n", buffer);

    send(new\_socket, hello, strlen(hello), 0);

    printf("Hello message sent\n");

    return 0;

}

**Client Side Code:**

#include <stdio.h>

#include <sys/socket.h>

#include <arpa/inet.h>

#include <unistd.h>

#include <string.h>

#define PORT 8080

*int* main(*int* *argc*, *char* const \**argv*[])

{

*int* sock = 0, valread;

*struct* sockaddr\_in serv\_addr;

*char* \*hello = "Hello from client";

*char* buffer[1024] = {0};

    if ((sock = socket(AF\_INET, SOCK\_STREAM, 0)) < 0)

    {

        printf("\n Socket creation error \n");

        return -1;

    }

    serv\_addr.sin\_family = AF\_INET;

    serv\_addr.sin\_port = htons(PORT);

    if (inet\_pton(AF\_INET, "127.0.0.1", &serv\_addr.sin\_addr) <= 0)

    {

        printf("\nInvalid address/ Address not supported \n");

        return -1;

    }

    if (connect(sock, (*struct* sockaddr \*)&serv\_addr, sizeof(serv\_addr)) < 0)

    {

        printf("\nConnection Failed \n");

        return -1;

    }

    send(sock, hello, strlen(hello), 0);

    printf("Hello message sent\n");

    valread = read(sock, buffer, 1024);

    printf("%s\n", buffer);

    return 0;

}

**PRACTICAL OUTPUT:**

****

**CONCLUSION:**

* In this practical we learnt about socket programming using c.

**PRACTICAL - 10**

**AIM:**

**Implement cyclic redundancy check in c to understand error handling at data link layer.**

**THEORY:**

* **CRC (Cyclic Redundancy Check):** CRC is a redundancy error technique used to determine the error.
* Following are the steps used in CRC for error detection:
* In CRC technique, a string of n 0s is appended to the data unit, and this n number is less than the number of bits in a predetermined number, known as division which is n+1 bits.
* Secondly, the newly extended data is divided by a divisor using a process is known as binary division. The remainder generated from this division is known as CRC remainder.
* Thirdly, the CRC remainder replaces the appended 0s at the end of the original data. This newly generated unit is sent to the receiver.
* The receiver receives the data followed by the CRC remainder. The receiver will treat this whole unit as a single unit, and it is divided by the same divisor that was used to find the CRC remainder.
* If the resultant of this division is zero which means that it has no error, and the data is accepted.
* CRC uses Generator Polynomial which is available on both sender and receiver side. An example generator polynomial is of the form like x3 + x + 1. This generator polynomial represents key 1011. Another example is x2 + 1 that represents key 101.
* **Source Code:**

#include <stdio.h>

#include <conio.h>

#include <string.h>

*void* main()

{

*int* i, j, keylen, msglen;

*char* input[100], key[30], temp[30], quot[100], rem[30], key1[30];

    printf("Enter Data: ");

    gets(input);

    printf("Enter Key: ");

    gets(key);

    keylen = strlen(key);

    msglen = strlen(input);

    strcpy(key1, key);

    for (i = 0; i < keylen - 1; i++)

    {

        input[msglen + i] = '0';

    }

    for (i = 0; i < keylen; i++)

        temp[i] = input[i];

    for (i = 0; i < msglen; i++)

    {

        quot[i] = temp[0];

        if (quot[i] == '0')

            for (j = 0; j < keylen; j++)

                key[j] = '0';

        else

            for (j = 0; j < keylen; j++)

                key[j] = key1[j];

        for (j = keylen - 1; j > 0; j--)

        {

            if (temp[j] == key[j])

                rem[j - 1] = '0';

            else

                rem[j - 1] = '1';

        }

        rem[keylen - 1] = input[i + keylen];

        strcpy(temp, rem);

    }

    strcpy(rem, temp);

    printf("\nQuotient is ");

    for (i = 0; i < msglen; i++)

        printf("%c", quot[i]);

    printf("\nRemainder is ");

    for (i = 0; i < keylen - 1; i++)

        printf("%c", rem[i]);

    printf("\nFinal data is: ");

    for (i = 0; i < msglen; i++)

        printf("%c", input[i]);

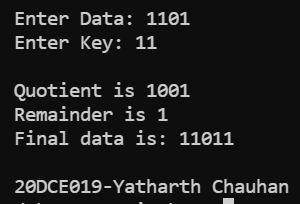
    for (i = 0; i < keylen - 1; i++)

        printf("%c", rem[i]);

    printf("\n20DCE019-Yatharth Chauhan");

}

**PRACTICAL OUTPUT:**

****

**CONCLUSION:**

* In this practical we practical we learnt about error handling using C language.