Ans 1 :- There are three kinds of transmission modes in computer networks. They are -
a.) Simplex mode :- In Simplex mode, the communication is unidirectional, as on a one-way street. Only one of the two devices on a link can transmit, the other can only receive. The simplex mode can use the entire capacity of the channel to send data in one direction.
Example: Keyboard and traditional monitors. The keyboard can only introduce input, the monitor can only give the output.
b.) Half-duplex mode :- In half-duplex mode, each station can both transmit and receive, but not at the same time. When one device is sending, the other can only receive, and vice versa. The half-duplex mode is used in cases where there is no need for communication in both directions at the same time. The entire capacity of the channel can be utilized for each direction.
Example: Walkie-talkie in which messages are sent one at a time and messages are sent in both directions.
c.) Full-duplex mode :- In full-duplex mode, both stations can transmit and receive simultaneously. In full duplex mode, signals going in one direction share the capacity of the link with signals going in another direction, this sharing can occur in two ways: Either the link must contain two separate paths one for sending and one for receiving or the capacity is divided between signals in both directions.

Ans 2 :- The basic communication model in computer networking is where the Sender (encodes the message) channel sends a message over a channel or medium and receiver (decodes the message) gives Feedback.
The components involved in the successful implementation of the basic communication model are as follows −

a.Sender − Who sends the message.
b.Encodes − Translates messages into symbols like words, pictures, sound, etc.
c.Channel or medium − It used to transmit messages. Some channels are face-to-face communication, over telephone, letters, television, newspapers, radio, etc.
d.Decode − Receiver decodes these symbols to understand what the sender wants to say.
e.Receiver − A person who receives the message.
f.Feedback − After receiving a message, the receiver sends feedback to the sender, answer and what he understands from the message.

Ans 3 :- A unique identifier (UID) is a numeric or alphanumeric string that is associated with a single entity within a given system. UIDs make it possible to address that entity, so that it can be accessed and interacted with.
Use :- The most widely known use of unique identifiers occurs when users register for a website or service. Customers are often provided with a username or user ID that allows the company they are registering with to differentiate them within their user logs. These identifiers are then also used for security and log on purposes.
Eg :- A Uniform Resource Locator (URL) is a particular type of URI that targets Web pages so that when a browser requests them, they can be found and served to users.

Ans 4 :- There are 5 types of computer networks :-

a. LAN (Local Area Network) –
   Systems connected in a small network like in a building or a small office.It is inexpensive It uses Ethernet or Token-ring technology.Two or more personal computers can be connected through wires or cables acting as nodes.Transfer of data is fast and is highly scored.
b. PAN (Personal Area Network) –
   The smallest computer network.Devices may be connected through Bluetooth or other infra-red enabled device.It has a connectivity range of up to 10 metres.It covers an area of up to 30 feet.Personal devices belonging to a single person can be connected to each other using PAN.
c. MAN (Metropolitan Area Network) –
   A network that can be connected within a city, for example, cable TV Connection.It can be in the form of Ethernet, ATM, Token-ring and FDDI.It has a higher range.This type of network can be used to connect citizens with the various Organisations.
d. WAN (Wide Area Network) –
   A network which covers over a country or a larger range of people.Telephonic lines are also connected through WAN.Internet is the biggest WAN in the world.Mostly used by Government Organisations to manage data and information.
e. VPN (Virtual Private Network): –
   A network which is constructed by using public wires to connect to a private network. There are a number of systems which enable you to create networks using the Internet as a medium for transporting data.These systems use encryptions and other security mechanisms to ensure only authorised users can access.

Ans 5 :- There are three main types of network protocols. These include network management protocols, network communication protocols and network security protocols:

a. Communication protocols include basic data communication tools like TCP/IP and HTTP.
b. Management protocols maintain and govern the network through protocols such as ICMP and SNMP.
c. Security protocols include HTTPS, SFTP, and SSL.

Ans 6 :- The arrangement of a network that comprises nodes and connecting lines via sender and receiver is referred to as network topology. The various network topologies are:

a. Mesh Topology:
   In a mesh topology, every device is connected to another device via a particular channel. In Mesh Topology, the protocols used are AHCP (Ad Hoc Configuration Protocols), DHCP (Dynamic Host Configuration Protocol), etc.
b. Star Topology:
   In star topology, all the devices are connected to a single hub through a cable. This hub is the central node and all other nodes are connected to the central node. The hub can be passive in nature i.e., not an intelligent hub such as broadcasting devices, at the same time the hub can be intelligent known as an active hub. Active hubs have

repeaters in them. Coaxial cables or RJ-45 cables are used to connect the computers. In Star Topology, many popular Ethernet LAN protocols are used as CD(Collision Detection), CSMA (Carrier Sense Multiple Access), etc.

c. Bus Topology:
   Bus topology is a network type in which every computer and network device is connected to a single cable. It is bi-directional. It is a multi-point connection and a non-robust topology because if the backbone fails the topology crashes. In Bus Topology, various MAC (Media Access Control) protocols are followed by LAN ethernet connections like TDMA, Pure Aloha, CDMA, Slotted Aloha, etc.

Ans 7 :- The Open Systems Interconnection (OSI) model describes seven layers that computer systems use to communicate over a network. It was the first standard model for network communications, adopted by all major computer and telecommunication companies in the early 1980s.

Ans 8 :- The OSI model has 7 layers :-

a. Physical Layer
   The physical layer is responsible for the physical cable or wireless connection between network nodes. It defines the connector, the electrical cable or wireless technology connecting the devices, and is responsible for transmission of the raw data, which is simply a series of 0s and 1s, while taking care of bit rate control.
b. Data Link Layer
   The data link layer establishes and terminates a connection between two physically-connected nodes on a network. It breaks up packets into frames and sends them from source to destination. This layer is composed of two parts—Logical Link Control (LLC), which identifies network protocols, performs error checking and synchronises frames, and Media Access Control (MAC) which uses MAC addresses to connect devices and define permissions to transmit and receive data.
c. Network Layer
   The network layer has two main functions. One is breaking up segments into network packets, and reassembling the packets on the receiving end. The other is routing packets by discovering the best path across a physical network. The network layer uses network addresses (typically Internet Protocol addresses) to route packets to a destination node.
d. Transport Layer
   The transport layer takes data transferred in the session layer and breaks it into “segments” on the transmitting end. It is responsible for reassembling the segments on the receiving end, turning it back into data that can be used by the session layer. The transport layer carries out flow control, sending data at a rate that matches the connection speed of the receiving device, and error control, checking if data was received incorrectly and if not, requesting it again.
e. Session Layer
   The session layer creates communication channels, called sessions, between devices. It is responsible for opening sessions, ensuring they remain open and functional while data

is being transferred, and closing them when communication ends. The session layer can also set checkpoints during a data transfer—if the session is interrupted, devices can resume data transfer from the last checkpoint.

f. Presentation Layer
The presentation layer prepares data for the application layer. It defines how two devices should encode, encrypt, and compress data so it is received correctly on the other end. The presentation layer takes any data transmitted by the application layer and prepares it for transmission over the session layer.

g. Application Layer
The application layer is used by end-user software such as web browsers and email clients. It provides protocols that allow software to send and receive information and present meaningful data to users. A few examples of application layer protocols are the Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Post Office Protocol (POP), Simple Mail Transfer Protocol (SMTP), and Domain Name System (DNS).

Ans 9 :- The advantages of the OSI model are

a. It is a generic model and acts as a guidance tool to develop any network model.
b. It is a layered model. Changes are one layer do not affect other layers, provided that the interfaces between the layers do not change drastically.
c. It distinctly separates services, interfaces, and protocols. Hence, it is flexible in nature. Protocols in each layer can be replaced very conveniently depending upon the nature of the network.
d. It supports both connection-oriented services and connectionless services.

The disadvantages of the OSI model are

a. It is purely a theoretical model that does not consider the availability of appropriate technology. This restricts its practical implementation.
b. The launching timing of this model was inappropriate. When OSI appeared, the TCP/IP protocols were already implemented. So, the companies were initially reluctant to use it.
c. The OSI model is very complex. The initial implementation was cumbersome, slow and costly.
d. Though there are many layers, some of the layers like the session layer and presentation layer have very little functionality when practically deployed.
e. There is a duplication of services in various layers. Services like addressing, flow control and error control are offered by multiple layers.
f. The standards of OSI model are theoretical and do not offer adequate solutions for practical network implementation.

Ans 10 :- In 1972, Bell Labs made a breakthrough in the field of version control. They created SCCS (Source Code Control System), which was written in C and developed by Marc Rochkind. This system shared many characteristics with modern version control systems such as the ability to create, edit, and track changes to files. However, it lacked the ability for more than one user to check out and work on a file at the same time. SCCS was made available to the public in 1977 and was the primary version control system into the early 1980s.

In 1982, Walter Tichy developed a new system called RCS, or Revision Control System. RCS still only allowed one user at a time to make edits and only supported the ability to work on single files, rather than a whole project. However, it did pioneer a new way of tracking changes called reverse deltas. Rather than store all of the versions of a file, RCS used a single recent version as its baseline from which all other versions were created. For the time, this was a faster and more efficient way of tracking changes.
A few years later in 1986, Dick Grune developed CVS (Concurrent Versions Systems), which was also written in C. CVS finally allowed more than one developer to work on a file at the same time. Users would deploy the UpdateVersion command to update a file to the latest version of that file on the server. CVS used delta encoding or compression, which tracks differences but not entire versions of files. With its use of a client-server model and branches, CVS is a much more modern example of version control
The next major version control system was Subversion (SVN), which was created in 2000 by CollabNet. SVN preserved many of the features included in CVS so that users could easily transition between the two. By 2010, SVN was renamed Apache Subversion after it became part of the Apache Software Foundation.

SVN is an example of a centralized version control system (CVCS). Changes are made to a single copy of the project on a server and other users can pull down the latest version of the project to make their edits. There are many Subversion clients still in use today, such as Tortoise SVN and SmartSVN. However, CVCS has been eclipsed in recent years by a more modern form of version control: the distributed version control system (DVCS).

This brings us to the present. Currently, the most well-known DVCS is Git, which was created in 2005 by Linus Torvalds. The basic logic behind a DVCS is that a copy of the repository and its history is downloaded by every user. Systems like Git are known for being fast and reliable, with good branching capabilities. For now, it seems that DVCS is the future of version control, but as repositories continue to grow in size, it may be necessary to innovate even further.

Ans 11 :- There are two basic types of version control systems. They are:-

a. Centralized version control
   With centralized version control systems, you have a single “central” copy of your project on a server and commit your changes to this central copy. You pull the files that you need, but you never have a full copy of your project locally. Some of the most common version control systems are centralized, including Subversion (SVN) and Perforce.

b. Distributed version control
   With distributed version control systems (DVCS), you don't rely on a central server to store all the versions of a project’s files. Instead, you clone a copy of a repository locally so that you have the full history of the project. Two common distributed version control systems are Git and Mercurial.

Ans 12 :- Switch to the branch that you want to create a pull request for.

Click Create Pull Request.
GitHub Desktop will open your default browser to take you to GitHub.
On GitHub, confirm that the branch in the base: drop-down menu is the branch where you want to merge your changes. Confirm that the branch in the compare: drop-down menu is the topic branch where you made your changes.Type a title and description for your pull request.
To create a pull request that is ready for review, click Create Pull Request. To create a draft pull request, use the drop-down and select Create Draft Pull Request, then click Draft Pull.

Ans 13 :- Git has three main states that your files can reside in: modified, staged, and committed:

Modified means that you have changed the file but have not committed it to your database yet.

Staged means that you have marked a modified file in its current version to go into your next commit snapshot.

Committed means that the data is safely stored in your local database.

Ans 14 :- A Git workflow is a recipe or recommendation for how to use Git to accomplish work in a consistent and productive manner. Git workflows encourage developers and DevOps teams to leverage Git effectively and consistently. Git offers a lot of flexibility in how users manage changes. Given Git's focus on flexibility, there is no standardized process on how to interact with Git. When working with a team on a Git-managed project, it’s important to make sure the team is all in agreement on how the flow of changes will be applied. To ensure the team is on the same page, an agreed-upon Git workflow should be developed or selected.

Ans 15 :- A file system defines how files are named, stored, and retrieved from a storage device.

Every time you open a file on your computer or smart device, your operating system uses its file system internally to load it from the storage device.

Or when you copy, edit, or delete a file, the file system handles it under the hood.

Whenever you download a file or access a web page over the Internet, a file system is involved too.

Ans 16 :- Storing Large content files eventually consumes a huge amount of space on the Git Server.

Moreover, since Git is a Distributed Version Control System, every clone and every checkout or pull of this repository will have to download each version of this huge file present on the Server; unlike Centralized Systems wherein just the latest version is downloaded.

Git LFS server comes to our rescue in terms of storing huge files.
Git LFS is a system for managing and versioning large files of upto 2GM in association with a Git repository. Instead of storing the large files within the Git repository as blobs, Git LFS stores special "pointer files" in the repository, while storing the actual file contents on a Git LFS server. The contents of the large file are downloaded automatically when needed, for example when a Git branch containing the large file is checked out.

Ans 17 :- In Git, storage is used to store all the information related to a project in a git repository. The git repository contains all the files, commits, and other data related to the project. Git uses a specialized storage mechanism called a "content-addressable storage" to store all the data.

The git storage system is based on the SHA-1 (Secure Hash Algorithm 1) cryptographic hash function, which is used to create a unique hash for each piece of data in the repository. Each file, commit, and other object in the git repository is associated with a unique SHA-1 hash, which serves as its identifier.

The git storage system stores the data in a set of objects, which include:

Blob objects: These objects store the contents of individual files in the repository.
Tree objects: These objects store the structure of the project's directories and subdirectories.
Commit objects: These objects store metadata about a particular commit, such as the author, the committer, and the commit message.
Tag objects: These objects store extra information about a specific commit, such as a release version number.
When you make changes to a file in a git repository, git creates new versions of the affected objects, including the blob object for the file, the tree object for the directory containing the file, and the commit object for the commit. The new versions of the objects are stored in the repository, and they are identified by new SHA-1 hashes.

Git uses a technique called "delta compression" to minimize the amount of storage used in the repository. Instead of storing a full copy of every version of every file, git only stores the differences (deltas) between versions.

Overall, the storage in git is designed to be efficient, secure and highly reliable, it stores all the data related to the project in a way that allows git to efficiently track the entire history of the project, including every change that has been made to it.