

### **Q1.What is a Program?**

**ANS :-** A Program is a set of instruction for the computer can execute to perform specific task or solve problem.

**Explain in your own words what a program is and how it functions.**

**ANS :-**

- A program is a set of instructions that tells a computer what to do. It is written by a person using a programming language.
- The program can perform many kinds of tasks, like doing calculations, showing messages, playing music, or even running games.
- In short, a program is like a recipe for the computer — it tells the computer what steps to follow to get a specific result.

### **What is Programming?**

**ANS :-**

- Programming is telling a computer what to do.
- programming means telling the computer what to do, step by step, using code.

**What are the key steps involved in the programming process?**

**ANS :-**

#### **Understand the problem**

– Know what you want the computer to do.

#### **Make a plan**

– Think about how to solve the problem step by step.

#### **Write the code**

– Use a programming language to write instructions for the computer.

#### **Test the program**

– Run the program to see if it works.

#### **Fix mistakes**

– Find and correct any errors in the program.

#### **Improve the program**

– Make the program better if needed.

### Use the program

– When it works well, use it to do the task.

### Types of Programming Languages

#### Mainly Two Types of Programming languages :-

1. High-level programming language

#### Types of High-level Programming language :-

1. Procedural Programming Languages
2. Functional Programming Language
3. Object-oriented programming language
4. Scripting Languages

2. Low-level programming language

#### What are the main differences between high-level and low-level programming languages?

ANS :-

| Low-Level Programming Language   | High-Level Programming Language                             |
|--|---|
| Low-level programming language is understand only machine code or binary language. | High-level programming language is human-readable language. |
| Low-level language is hard to read and write.                                      | High-level language is easy to read and write.              |
| Slower execution   | Faster execution  |
| Python , java , c, c++   | Machine code , Assembly language                            |
| Manual memory management   | Automatic memory management                                 |

### World Wide Web & How Internet Works

#### Describe the roles of the client and server in web communication.

ANS :-

The client (like a web browser) sends requests to the server (like a web server) for resources, and the server responds by sending the requested data.

Client :- A client is a program that runs on the local machine requesting service from the server. A client program is a finite program means that the service started by the user and terminates when the service is completed.

#### Examples:

---

## Module 1 – Overview of IT Industry

---

- Google Chrome, Safari, Firefox (browsers)
- Mobile apps (e.g., food delivery, email apps)

Server :- A server is a program that runs on the remote machine providing services to the clients. When the client requests for a service, then the server opens the door for the incoming requests, but it never initiates the service.

### Examples:

- Web server (Apache, Nginx)
- Application server (Node.js, Java EE)
- Database server (MySQL, MongoDB)

### Network Layers on Client and Server

#### Explain the function of the TCP/IP model and its layers

- The TCP/IP model (Transmission Control Protocol/Internet Protocol) is a set of communication protocols used for transmitting data over the internet.
- The TCP/IP model is a conceptual framework that standardizes how computers communicate over a network.

#### It divides communication into four layers:

##### 1. Application layer

This is the topmost layer that directly interacts with the user or application software (like browsers or email apps).

Key Protocols: HTTP, FTP, SMTP, DNS

Function: Provides services like web browsing, email, file transfer, and name resolution to users.

##### 2. Transport layer

It manages end-to-end communication between devices. It also ensures the reliability of data transfer.

Key Protocols: TCP (reliable), UDP (fast but unreliable)

Function: Ensures complete, ordered, and error-checked delivery of data between applications.

##### 3. Internet layer

This layer is responsible for addressing and routing the data across multiple networks. It ensures the data reaches the correct destination.

Key Protocols: IP (Internet Protocol), ICMP (used for error messages and diagnostics)

Function: Determines the best path for data to travel across networks.

##### 4. Network Access layer

This is the lowest layer of the TCP/IP model. It handles the physical transmission of data over a network (like cables or Wi-Fi).

Key Protocols: Ethernet, Wi-Fi, ARP

Function: Transmits raw data (called frames) between devices on the same local network.

### Explain Client Server Communication

**ANS :-** Client-server communication is a network model in which a client (such as a computer, app, or browser) requests services or data from a server, and the server responds by providing the required data or service.

### 2. How It Works

1. The client sends a request to the server (e.g., open a website or access a file).
2. The server processes the request and prepares a response.
3. The server sends the response back to the client.
4. The client receives and displays or uses the data.

### 3. Example

- When you open a website:
  - Your browser (the client) sends a request to the website's server.
  - The server processes the request and sends back the website content.
  - The browser displays the webpage.

### Types of Internet Connections

**ANS :-**

Types of Internet Connections (Simplified)

#### 1. Dial-Up Connection

- Uses a modem and telephone line to connect to the internet.
- Very slow and outdated.
- Cannot use phone and internet at the same time.
- Must dial a number to connect.

#### 2. Broadband Connection

- High-speed internet access.
- Can work over cable or telephone lines.
- Supports multiple users at once.
- No need to dial, always connected.
- Can use internet and phone together.

### 3. DSL (Digital Subscriber Line)

- Provides internet through telephone lines.
- Always-on connection (no dialing needed).
- Speed: 128 Kbps to 8 Mbps.
- Can use internet and landline simultaneously.

### 4. Cable Connection

- Uses cable TV lines for internet access.
- Requires a cable modem.
- Speed: 512 Kbps to 20 Mbps.
- Fast for both downloads and uploads.

### 5. Satellite Connection

- Used in rural or remote areas.
- Connects via satellites in Earth's orbit.
- Speed: 512 Kbps to 2 Mbps.
- Higher delay/latency due to long-distance signals.

### 6. Wireless Connection

- Uses radio frequency to connect.
- No cables or telephone lines needed.
- Speed: 5 Mbps to 20 Mbps.
- Can be accessed from anywhere within range.

### 7. Cellular Connection (3G/4G)

- Internet via mobile networks (3G, 4G).
- 3G speed: ~2 Mbps, 4G: ~21 Mbps.
- No wires required, portable.
- Depends on mobile service provider.

### 8. ISDN (Integrated Services Digital Network)

- Uses digital telephone lines.
- Transmits voice and data.
- Speed: Up to 128 Kbps.
- Better quality than older dial-up systems.

**How does broadband differ from fiber-optic internet?**

**ANS :-**

## Module 1 – Overview of IT Industry

| Broadband internet  | Fiber-optic internet   |
|---|--|
| Broadband is a general term that means high-speed internet.         | Fiber-optic internet is the fastest and most reliable type of broadband. |
| Broadband usually uses copper cables (like telephone or TV cables). | Fiber-optic internet uses glass or plastic fiber cables.                 |
| Broadband offers medium to high speed.                              | Fiber gives very high speed, often up to 1 Gbps or more.                 |
| Broadband uses electrical signals to send data.                     | Fiber-optic uses light signals, which are much faster.                   |
| Broadband uses protocols like DOCSIS, PPPoE, etc.                   | Fiber uses GPON, EPON, which are designed for optical networks.          |

### Protocols

**What are the differences between HTTP and HTTPS protocols?**

**ANS :-**

| HTTP   | HTTPS   |
|--|---|
| HTTP stands for hyper text transfer protocol.    | HTTPS stands for hyper text transfer protocol secure. |
| HTTP is not secure — data is sent in plain text. | HTTPS is secure — data is encrypted during transfer.  |
| HTTP does not encrypt the data.                  | HTTPS encrypts the data using SSL/TLS.                |
| HTTP URLs start with http://                     | HTTPS URLs start with https://                        |
| HTTP uses port 80 by default.                    | HTTPS uses port 443 by default.                       |

### Application Security

**What is the role of encryption in securing applications?**

**ANS :-**

- Encryption is the process of converting readable data (called plaintext) into unreadable code (called ciphertext) so that only authorized users can read it.

**Role of Encryption in Securing Applications:-**

**1. Protects Sensitive Data:**

Encryption hides important info like passwords and credit cards so hackers can't read it.

**2. Ensures Data Privacy:**

Only the intended user can see the data, keeping it private even on public Wi-Fi.

**3. Prevents Data Tampering:**

Encryption checks if data is changed during transfer to stop unauthorized edits.

## Module 1 – Overview of IT Industry

---

### 4. Secures Communication:

It protects data sent between users and servers, preventing spying or attacks.

### 5. Builds User Trust:

Encrypted apps make users feel safe, encouraging them to share sensitive info.

## Software Applications and Its Types

**What is the difference between system software and application software?**

| System software   | Application software   |
|---|--|
| System Software helps run and manage the computer hardware and basic functions. | Application Software is used to perform specific tasks for the user.               |
| System Software supports the system and other software.                         | Application Software helps users do activities like writing, browsing, or editing. |
| Operating System (Windows, Linux), Device Drivers, BIOS.                        | MS Word, Google Chrome, Photoshop, VLC Player.                                     |
| System Software interacts with hardware.  | Application Software interacts with the user.                                      |
| System Software is installed with the system and works in the background.       | Application Software is installed by the user as needed.                           |

## Software Architecture

**What is the significance of modularity in software architecture?**

**ANS :-**

- Modularity means dividing a large software into small, independent parts called modules. Each module is responsible for a specific task and can work independently from the rest of the system.
- It is important because it makes software easier to maintain. If there is an error or change needed, only that module needs to be updated without affecting the whole software. It also allows different developers or teams to work on different modules at the same time, which saves development time.
- Modularity makes testing and debugging easier because you can test each module separately. It also increases code reusability, meaning once a module is created, it can be used in other software without writing the same code again. It gives flexibility to add, remove, or update features without changing the entire system.

## Layers in Software Architecture

**Why are layers important in software architecture?**

1. Layers divide the software into different parts based on their functions.
2. Each layer has a specific role, like user interface, logic, or data management.
3. This makes the software easier to understand and work with.

4. Developers can work on one layer without affecting the others.
5. It saves time and reduces the chances of errors.
6. Testing and debugging become easier when done layer by layer.
7. Changes or updates can be made in one layer without disturbing the rest.
8. In short, layers make the software more organized, flexible, and maintainable.

### **Software Environments**

**Explain the importance of a development environment in software production**

**ANS :-**

- A development environment provides the tools and setup needed to write, test, and debug code.
- It helps developers build software in a structured and efficient way.
- It includes editors, compilers, debuggers, and other tools in one place.
- It allows code to be written and tested safely before it is released.
- It helps identify and fix errors early in the development process.
- It improves productivity by automating common tasks like compiling or testing.
- It supports version control so that changes in code can be tracked easily.
- It ensures that the software works the same way on all developer systems.
- In short, a good development environment speeds up software development and improves quality.

### **Source Code**

**What is the difference between source code and machine code?**

**ANS :-**

| <b>Source code</b>   | <b>Machine code</b>   |
|--|---|
| Source code is written by programmers using human-readable programming languages like Python, C++, or Java.        | Machine code is written in binary (0s and 1s) that the computer's processor understands directly. |
| Source code is easy for humans to read and understand.   | Machine code is difficult for humans to read because it is in binary form.                        |
| Source code must be converted into machine code using a compiler or interpreter for the computer to understand it. | Machine code is executed directly by the processor.   |
| Source code is used for writing and developing software.   | Machine code is used for executing instructions on the computer.                                  |
| Source code can be easily modified by programmers.   | Machine code is very hard to modify because it is in complex binary form.                         |
| Source code is in text form, readable and editable.  | Machine code is in binary form, understood only by the processor.                                 |



## Module 1 – Overview of IT Industry

---

### Github and Introductions

**Why is version control important in software development?**

**ANS :-**

- Version control keeps track of every change made to the code over time.
- It allows multiple developers to work on the same project without overwriting each other's work.
- If a mistake happens, version control helps easily revert to a previous correct version.
- It helps manage different versions or features of the software simultaneously.
- Version control improves collaboration and communication among team members.

### Student Account in Github

**What are the benefits of using Github for students?**

**ANS :-**

- GitHub helps students learn how to save and manage their code safely.
- It makes working with friends on projects easy.
- Students can show their work to teachers and future employers.
- GitHub gives free tools to help students learn coding.
- It helps find and fix mistakes in the code quickly.
- Students can join and help with real coding projects online.
- It teaches good habits like saving work often and writing clear notes.

### Types of Software

**What are the differences between open-source and proprietary software?**

| Open-source software   | Proprietary software   |
|--|--|
| Open-source software is free and anyone can see and change its code. | Proprietary software is owned by a company and its code is secret. |
| Open-source software is usually free to use.                         | Proprietary software you have to buy or subscribe to use.          |
| Open-source lets users change and improve the software.              | Proprietary software limits what users can change.                 |
| Users of open-source software can customize it to fit their needs.   | Users of proprietary software have limited customization options.  |

### GIT and GITHUB Training

**How does GIT improve collaboration in a software development team?**

**ANS :-** Git improves collaboration in a software development team by allowing multiple developers to work on the same project simultaneously without overwriting each other's changes. It tracks all changes made by team members, making it easy to merge work from different people using branches. Git also helps revert to earlier versions if needed and shows who made each change, which improves accountability. Overall, Git makes teamwork smoother, organized, and more efficient.

### **Application Software**

**What is the role of application software in businesses?**

**ANS :-**

1. Application software helps businesses do specific tasks like managing accounts and customers.
2. It saves time by automating routine work and reducing mistakes.
3. Businesses use it to organize data and track sales or inventory.
4. It provides reports to help in making better decisions.
5. It improves communication and overall productivity.
6. In short, application software makes business work easier and faster.

### **Software Development Process**

**What are the main stages of the software development process?**

**ANS :-**

**1) planning / requirement gathering :** The developer collects Requirements from the client or Company to plan the project.

**2) Analysis :** Developer Analysis the requirements and creates a Software Requirements (SRC) , ensuring no changes.

**3) Designing :** developer work from designing , different3d model , ER diagram , User Case diagram , flowchart , data flow diagram etc..

**4) Implementation/development/building :** developer work from backend side a project implementation or development.

**5) Testing :**The Developer meets the client to tested the project and client check the work is correctly success.

**6)Maintenance :** maintain the project according to the developer Time prior.

### **Software Requirement**

**Why is the requirement analysis phase critical in software development?**

**ANS :-** Requirement analysis is important because it tells developers what the users want. It helps avoid mistakes and saves time and money. Without it, the software might not work as needed. It sets a clear plan for the project.

### **Software Analysis**

#### **What is the role of software analysis in the development process?**

**ANS :-** Software analysis helps understand what the software needs to do by studying user requirements and problems. It finds out the exact features and functions needed. This helps plan the design and development correctly. Good analysis prevents mistakes and ensures the final software meets user needs. It acts as a bridge between what users want and what developers build.

### **System Design**

#### **What are the key elements of system design?**

**ANS :-**

**Architecture Design:** Defines the overall structure and how different parts interact.

**Interface Design:** Plans how users will interact with the system.

**Data Design:** Organizes how data will be stored and managed.

**Component Design:** Details the functions of each module or part.

**Security Design:** Ensures the system is protected from threats.

**Performance Design:** Makes sure the system works efficiently and quickly.

### **Software Testing**

#### **Why is software testing important?**

Software testing is a very important part of the software development process. It helps make sure that the final product works correctly, is safe to use, and gives a good experience to the user.

##### **1. Finds Bugs Early**

Testing helps developers find and fix mistakes (bugs) in the code before the software is released. Catching these problems early saves time and avoids big issues later.

##### **2. Improves Quality**

Good testing makes sure that the software does what it is supposed to do. It checks if all features are working properly and if the software meets the requirements.

### 3. Saves Time and Money

Finding and fixing bugs after the software is released can be very expensive. Testing during development helps avoid this by catching problems early.

### 4. Gives Better User Experience

If the software is tested well, users will have fewer issues. It will be smooth, fast, and easy to use, which keeps users happy and satisfied.

### 5. Protects from Security Risks

Testing also checks for security problems, like whether hackers can steal data. This is very important for apps that handle personal or sensitive information.

## Maintenance

**What types of software maintenance are there?**

**ANS :-**

1. **Corrective Maintenance:**  
This involves fixing errors and bugs in the software system.
2. **Patching:**  
It is an emergency fix implemented mainly due to pressure from management. Patching is done for corrective maintenance, but it gives rise to unforeseen future errors due to lack of proper impact analysis.
3. **Adaptive Maintenance:**  
This involves modifying the software system to adapt it to changes in the environment, such as changes in hardware or software, government policies, and business rules.
4. **Perfective Maintenance:**  
This involves improving functionality, performance, and reliability, and restructuring the software system to improve changeability.
5. **Preventive Maintenance:**  
This involves taking measures to prevent future problems, such as optimization, updating documentation, reviewing and testing the system, and implementing preventive measures such as backups.

## Development

**What are the key differences between web and desktop applications?**

**ANS :-**

## Module 1 – Overview of IT Industry

---

| Web application   | Desktop application  |
|---|--|
| Web apps are hosted on a web server and accessed via a browser. | Desktop apps are installed on a specific computer (like Windows OS). |
| Web apps can be accessed from anywhere with internet.           | Desktop apps work only on the installed system.                      |
| Web apps are more complex to develop.                           | Desktop apps are generally easier to build.                          |
| Web apps need a web server like IIS to run.                     | Desktop apps run directly on the OS.                                 |
| Web apps are more vulnerable due to internet exposure.          | Desktop apps offer better security control.                          |
| Web app maintenance is done centrally by developers.            | Users maintain desktop apps individually.                            |
| Web: Google Chrome, Firefox, Gmail                              | Desktop: MS Word, Adobe Photoshop, Excel.                            |

### Web Application

**What are the advantages of using web applications over desktop applications?**

**ANS :-**

- **Accessible Anywhere:**  
You can use web applications from any device with an internet connection, no matter where you are.
- **No Installation Required:**  
Web apps run directly in your browser, so you don't need to download or install anything on your computer.
- **Automatic Updates:**  
The latest version is always available because updates happen on the server, so users don't have to update manually.
- **Works on All Devices:**  
Web applications work on different operating systems like Windows, Mac, Linux, and even mobile phones.
- **Easier Maintenance:**  
Developers handle all maintenance and updates on the server side, so users don't have to worry about it.
- **Cost-Effective:**  
It is usually cheaper to develop and maintain web applications because you don't need separate versions for each platform.
- **Better for Collaboration:**  
Web apps allow multiple users to work together easily in real-time from different places

### 28. Designing

**What role does UI/UX design play in application development?**

**ANS :-**

- **Improves User Experience (UX):**  
Good UX design makes the app easy to use, intuitive, and enjoyable, helping users achieve their goals smoothly.
- **Enhances Visual Appeal (UI):**  
Attractive and consistent user interfaces make the app look professional and inviting, which increases user trust and engagement.
- **Increases User Satisfaction:**  
When users find the app simple and pleasant, they are more likely to keep using it and recommend it to others.
- **Reduces Errors and Confusion:**  
Clear design and smooth workflows prevent user mistakes and frustration.
- **Supports Brand Identity:**  
UI/UX design reflects the brand's personality and values, creating a stronger connection with users.
- **Boosts Business Success:**  
Apps with great UI/UX tend to have higher user retention, better reviews, and more sales or conversions.

### **29. Mobile Application**

**What are the differences between native and hybrid mobile apps?**

**ANS :-**

| <b>Native mobile apps</b>                                   | <b>Hybrid mobile apps</b>                               |
|---|---|
| Native apps are made for one device like Android or iPhone. | Hybrid apps work on many devices using the same code.   |
| Native apps run very fast and use phone features well.      | Hybrid apps can be slower and don't use features fully. |
| Native apps look and feel better on their device.           | Hybrid apps may look a bit different on each device.    |
| Native apps take more time and money to make.               | Hybrid apps are faster and cheaper to build.            |
| Native apps need separate coding for each platform.         | Hybrid apps use one code for all platforms.             |

### **DFD (Data Flow Diagram)**

**What is the significance of DFDs in system analysis?**

- ANS :- DFD also known as data flow diagram.
- A graphical tool, useful for communicating with users, managers, and other personnel.
- DFD also known as Bubble Chart.

**Components of DFD:**

- Process
- Data Flow
- Data Store
- External Entity

### Flow Chart

**How do flowcharts help in programming and system design?**

**ANS :-** A Flow Chart is a Diagram that Graphical representation of algorithm.

- A Flow Chart is use to symbols , Shapes and Arrows in process.
- Flow chart is step by step process and solve a task to easily understand and make a decision.
- Flowcharts are very useful in programming and system design because they show the process in a clear and visual way.
- They break down complex tasks into simple steps using shapes and arrows, which makes it easier to understand how the program works.
- Before writing code, programmers use flowcharts to plan the sequence of actions and decisions, which helps prevent mistakes later.
- Flowcharts also help teams and clients communicate better by providing a common picture of how the system will function.
- Additionally, they make it easier to find and fix problems and serve as good documentation for future updates.