**1.** **What is software? What is software engineering?**

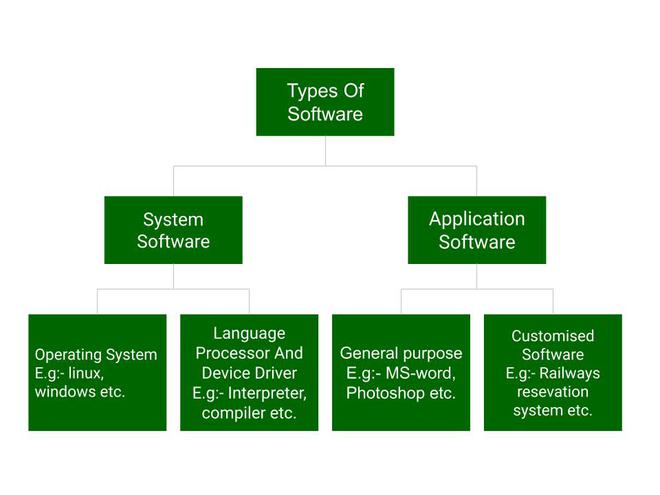
**Ans:-** the software is a computer program that provides a set of instructions to execute a user’s commands and tell the computer what to do.

**What is software engineering?**

**Ans:-** **Software Engineering** is the process of designing, developing, testing, and maintaining software.

**2. Explain types of software**

**Ans:-**



* **Tow type**

1. system software
2. Application software
3. **system software**

system software basically controls a computer’s internal functioning and controls hardware devices such as monitors, printers, and storage devices, etc. It is like an interface between hardware and user applications, it helps them to communicate with each other because hardware understands machine language (i.e. 1 or 0) whereas user applications are work in human-readable languages like English, Hindi, German, etc. so system software converts the human-readable language into machine language and vice versa.

1. **Application software**

 application software is designed to perform a specific task for end-users. It is a product or a program that is designed only to fulfil end-users’ requirements. It includes word processors, [spreadsheets](https://www.geeksforgeeks.org/introduction-to-excel-spreadsheet/), database management, inventory, payroll programs, etc.

| **System Software** | **Application Software** |
| --- | --- |
| It is designed to manage the resources of the computer system, like memory and process management, etc. | It is designed to fulfil the requirements of the user for performing specific tasks. |
| Written in a low-level language. | Written in a high-level language. |
| Less interactive for the users. | More interactive for the users. |
| System software plays vital role for the effective functioning of a system. | Application software is not so important for the functioning of the system, as it is task specific. |
| It is independent of the application software to run. | It needs system software to run. |

1. **What is SDLC? Explain each phase of SDLC**

**Ans:-**



**Stage1: Planning and requirement analysis**

**Stage2: Defining Requirements**

**Stage3: Designing the Software**

**Stage4: Developing the project**

**Stage5: Testing**

**Stage6: Deployment**

**Stage7: Maintenance**

**Requirements gathering and analysis:**

 This phase involves gathering information about the software requirements from stakeholders, such as customers, end-users, and business analysts.

**Design:**

 In this phase, the software design is created, which includes the overall architecture of the software, data structures, and interfaces. It has two steps.

* **High-level design (HLD):** It gives the architecture of software products.
* **Low-level design (LLD):** It describes how each and every feature in the product should work and every component.

**Implementation or coding:**

 The design is then implemented in code, usually in several iterations, and this phase is also called as Development.

things you need to know about this phase:

* This is the longest phase in SDLC model.
* This phase consists of Front end + Middleware + Back-end.

**Testing:**

The software is thoroughly tested to ensure that it meets the requirements and works correctly.

**Deployment:**

 After successful testing, The software is deployed to a production environment and made available to end-users.

**Maintenance:**

This phase includes ongoing support, bug fixes, and updates to the software.

**4 .What is DFD? Create a DFD diagram on Flipkart**

**Ans:-** A data flow diagram (DFD) maps out the flow of information for any process or system.

|  |  |
| --- | --- |
| **Symbol** | **Description** |
| [dfd symbol](https://meeraacademy.com/wp-content/uploads/2016/09/arro.jpg) | **Data Flow** – Data flow are pipelines through the packets of information flow. |
| [dfd process](https://meeraacademy.com/wp-content/uploads/2016/09/process.jpg) | **Process :**A Process or task performed by the system. |
| [dfd entry symbol](https://meeraacademy.com/wp-content/uploads/2016/09/Entity.jpg) | **Entity :**Entity are object of the system. A source or destination data of a system. |
| [dfd source symbol](https://meeraacademy.com/wp-content/uploads/2016/09/source.jpg) | **Data Store :** A place where data to be stored. |

1. **External entity:** an outside system that sends or receives data, communicating with the system being diagrammed. They are the sources and destinations of information entering or leaving the system. They might be an outside organization or person, a computer system or a business system. They are also known as terminators, sources and sinks or actors. They are typically drawn on the edges of the diagram.
2. **Process:** any process that changes the data, producing an output. It might perform computations, or sort data based on logic, or direct the data flow based on business rules. A short label is used to describe the process, such as “Submit payment.”
3. **Data store:** files or repositories that hold information for later use, such as a database table or a membership form. Each data store receives a simple label, such as “Orders.”
4. **Data flow:** the route that data takes between the external entities, processes and data stores. It portrays the interface between the other components and is shown with arrows, typically labeled with a short data name, like “Billing details.”

**5. What is Flow chart? Create a flowchart to make addition of two numbers**

**Ans:-** A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy-to-understand diagrams. Flowcharts, sometimes spelled as flow charts, use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence. They can range from simple, hand-drawn charts to comprehensive computer-drawn diagrams depicting multiple steps and routes. If we consider all the various forms of flowcharts, they are one of the most common diagrams on the planet, used by both technical and non-technical people in numerous fields. Flowcharts are sometimes called by more specialized names such as [Process Flowchart](https://www.lucidchart.com/pages/process-flow-diagrams), Process Map, Functional Flowchart, Business Process Mapping, Business Process Modeling and Notation (BPMN),  or Process Flow Diagram (PFD). They are related to other popular diagrams, such as Data Flow Diagrams (DFDs) and Unified Modeling Language (UML) Activity Diagrams.

A simple flowchart representing a process for dealing with a non-functioning [lamp](https://en.wikipedia.org/wiki/Light_fixture).

A diagram of a lamp plugged in and bulb turned out

Description automatically generated

## Types of flowcharts

* **Document Flowcharts:** These “have the purpose of showing existing controls over document-flow through the components of a system. … The chart is read from left to right and documents the flow of documents through the various business units.”
* **Data Flowcharts:** These show “the controls governing data flows in a system. Data flowcharts are used primarily to show the channels that data is transmitted through the system rather than how controls flow.”
* **System Flowcharts:** These “show the flow of data to and through the major components of a system such as data entry, programs, storage media, processors, and communication networks.”
* **Program Flowcharts:** These show “the controls placed internally to a program within a system.”
* **System Flowchart:** Identifies the devices to be used.
* **General Flowchart:** Overview.
* **Detailed Flowchart**: Increased detail.

Bohl, in her 1978 book A Guide for Programmers, listed only two:

* **System Flowchart:**
* **Program Flowchart:**

But Fryman, in his 2001 book *Quality and Process Improvement*, differentiated the types in multiple ways from more of a business perspective than a computer perspective:

* **Decision Flowchart:**
* **Logic Flowchart:**
* **Systems Flowchart:**
* **Product Flowchart:**
* **Process Flowchart:**

1. **What is Use case Diagram? Create a use-case on bill payment on paytm.**

**Ans:-** Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors.

## Use Case Examples

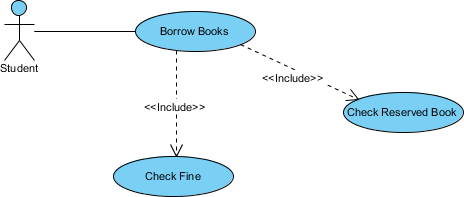
### **Use Case Example - Association Link**

A Use Case diagram illustrates a set of use cases for a system, i.e. the actors and the relationships between the actors and use cases.



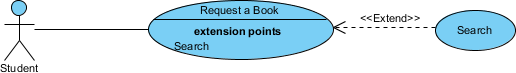
### **Use Case Example - Include Relationship**

The include relationship adds additional functionality not specified in the base use case. The <<Include>> relationship is used to include common behaviour from an included use case into a base use case in order to support the reuse of common behaviour.



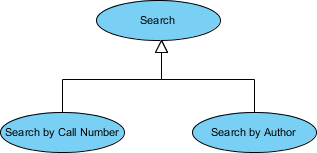
### **Use Case Example - Extend Relationship**

The extend relationships are important because they show optional functionality or system behaviour. The <<extend>> relationship is used to include optional behaviour from an extending use case in an extended use case. Take a look at the use case diagram example below. It shows an extend connector and an extension point "Search".



### **Use Case Example - Generalization Relationship**

A generalization relationship means that a child use case inherits the behaviour and meaning of the parent use case. The child may add or override the behaviour of the parent. The figure below provides a use case example by showing two generalization connectors that connect between the three use cases.



## Use Case Diagram - Vehicle Sales Systems

The figure below shows a use case diagram example for a vehicle system. As you can see even a system as big as a vehicle sales system contains not more than 10 use cases! That's the beauty of use case modeling.

The use case model also shows the use of extend and include. Besides, there are associations that connect between actors and use cases.

