—> **What is SDLC**

SDLC is a structure imposed on the development of a software product that defines the process for planning, implementation, testing, documentation, deployment, and ongoing maintenance and support.

—> **What is agile methodology?**

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software products.

—> **What is SRS**

SRS :- Software Requirements Specification, A software requirements specification (SRS) is a complete description of the behaviour of the system to be developed.

—> **What is oops**

Identifying objects and assigning responsibilities to these objects. Objects communicate to other objects by sending messages. Messages are received by the methods of an object An object is like a black box. The internal details are hidden. Object is derived from abstract data type Object-oriented programming has a web of interacting objects, each house-keeping its own state. Objects of a program interact by sending messages to each other.

—> **Write Basic Concepts of oops**

Object, Class, Encapsulation, Inheritance, Polymorphism, Overriding, Overloading, Abstraction

—> **What is object**

Tangible Things as a car, printer, Roles as employee, boss, Incidents as flight, overflow, Interactions as contract, sale, Specifications as color, shape.

—> **What is class**

When you define a class, you define a blueprint for an object. This doesn't actually define any data, but it does define what the class name means, that is, what an object of the class will consist of and what operations can be performed on such an object. A class represents an abstraction of the object and abstracts the properties and behaviour of that object. Class can be considered as the blueprint or definition or a template for an object and describes the properties and behaviour of that object, but without any actual existence. An object is a particular instance of a class which has actual existence and there can be many objects (or instances) for a class. In the case of a car or laptop, there will be a blueprint or design created first and then the actual car or laptop will be built based on that. We do not actually buy these blueprints but the actual objects.

—> **What is encapsulation**

Encapsulation is the practice of including in an object everything it needs hidden from other objects. The internal state is usually not accessible by other objects.

—> **What is inheritance**

Inheritance means that one class inherits the characteristics of another class. This is also called a “is a” relationship

—> **What is polymorphism**

Polymorphism means “having many forms”. · It allows different objects to respond to the same message in different ways, the response specific to the type of the object.

—> What is RDBMS

RDBMS stands for Relational Database Management System. RDBMS is the basis for SQL, and for all modern database systems like MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.

—> **What is SQL**

SQL is Structured Query Language, which is a computer language for storing, manipulating and retrieving data stored in relational databases.

—> **Write SQL Commands**

DDL – Data Definition Language, DML – Data Manipulation Language, DCL – Data Control Language, DQL – Data Query Language

—> **Write SDLC phases with basic introduction**

Requirements Collection/Gathering Establish Customer Needs

Analysis Model And Specify the requirements- “What”

Design Model And Specify a Solution – “Why”

Implementation Construct a Solution In Software Testing Validate the solution against the requirements.

Maintenance Repair defects and adapt the solution to the new requirements.

—> **Explain Phases of the waterfall model**

The waterfall is unrealistic for many reasons, especially, Requirements must be “frozen” too early in the life cycle, Requirements are validated too late.

—> **Write phases of spiral model**

Spiral Model is very widely used in the software industry as it is in sync with the natural development process of any product i.e. learning with maturity and also involves minimum risk for the customer as well as the development firms. Following are the typical uses of the Spiral model.

—> **Write agile manifesto principles**

Individuals and interactions - in agile development, self-organisation and motivation are important, as are interactions like co-location and pair programming, Working software - Demo working software is considered the best means of communication with the customer to understand their requirement, instead of just depending on documentation, Customer collaboration - As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to get proper product requirements, Responding to change - agile development is focused on quick responses to change and continuous development.

—> **Write type of joins.**

INNER JOIN: returns rows when there is a match in both tables, LEFT JOIN: returns all rows from the left table, even if there are no matches in the right table, RIGHT JOIN: returns all rows from the right table, even if there are no matches in the left table, FULL JOIN: returns rows when there is a match in one of the tables. DDL - Data Definition Language

—> **Explain the working methodology of the agile model and also write pros and cons.**

Pros

Is a very realistic approach to software development, Promotes teamwork and cross training, Functionality can be developed rapidly and demonstrated, Resource requirements are minimum, Suitable for fixed or changing requirements, Delivers early partial working solutions, Good model for environments that change steadily, Minimal rules, documentation easily employed, Enables concurrent development and delivery within an overall planned context, Little or no planning required, Easy to manage, Gives flexibility to developers.

Cons

Not suitable for handling complex dependencies, More risk of sustainability, maintainability and extensibility, An overall plan, an agile leader and agile PM practice is a must without which it will not work, Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines, Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction, There is very high individual dependency, since there is minimum documentation generated, Transfer of technology to new team members may be quite challenging due to lack of.