Module-1

1. What is SDLC

Ans: SDLC stands for Software Development Life Cycle. It is a structured approach to software development that defines the processes and stages involved in building software from conception to deployment and maintenance.

1. What is agile methodology?

Ans: Agile methodology is an approach to software development that emphasizes flexibility, collaboration, and incremental delivery of software in short iterations. Unlike traditional waterfall methods, where development is linear and each phase is completed before the next begins, agile methodologies promote adaptive planning, evolutionary development, early delivery, and continuous improvement.

Key principles and characteristics of agile methodology include:

1. **Iterative and Incremental Development**: Agile projects are divided into small iterations or sprints (usually 1-4 weeks long), where cross-functional teams work on small features or parts of the software. Each iteration results in a potentially shippable product increment.
2. **Collaboration and Communication**: Agile teams emphasize daily communication and collaboration among all team members, including developers, testers, designers, and product owners. This ensures that everyone is aligned on goals, progress, and challenges.
3. **Adaptive Planning**: Agile projects embrace changing requirements and priorities throughout the development process. Instead of a fixed plan, agile teams use adaptive planning techniques to respond to feedback and adjust their priorities and tasks accordingly.
4. **Continuous Improvement**: Agile methodologies promote a culture of continuous improvement through regular retrospectives. Team members reflect on their processes, identify areas for improvement, and implement changes to enhance productivity and quality.
5. **Customer Involvement**: Agile methodologies emphasize frequent and early delivery of working software to customers. This allows for rapid feedback, enabling teams to validate assumptions, gather requirements, and make adjustments based on real-world usage.
6. **Flexible and Responds to Change**: Agile methodologies are designed to be flexible and responsive to changes in requirements, market conditions, or technology advancements. This agility helps teams deliver value quickly and effectively.
7. What is SRS

Ans: SRS stands for Software Requirements Specification. It is a detailed document that describes the functional and non-functional requirements of a software system. The SRS serves as a contract between the development team and the customer or stakeholders, outlining what the software should do, how it should behave, and the constraints under which it must operate.

1. What is oops

Ans: OOPS stands for Object-Oriented Programming System or Object-Oriented Programming (OOP). It is a programming paradigm that organizes software design around objects and data, rather than actions and logic.

1. Write Basic Concepts of oops

Ans: Object-Oriented Programming (OOP) is built upon several fundamental concepts that help in organizing and designing software systems. Here are the basic concepts of OOP:

1. **Class**: A class is a blueprint or template that defines the attributes (data members) and behaviours (methods) that objects of the class should have. It serves as a blueprint from which objects are created.
2. **Object**: An object is an instance of a class. It represents a specific entity or instance of the class, possessing its own state (attributes) and behaviour (methods).
3. **Encapsulation**: Encapsulation is the bundling of data (attributes) and methods (functions) that operate on the data into a single unit (class). It hides the internal state of an object from the outside world and only exposes a controlled interface for interacting with the object.
4. **Inheritance**: Inheritance is a mechanism where a class (subclass or derived class) inherits attributes and behaviours from another class (superclass or base class). It promotes code reuse and allows the creation of a hierarchy of classes.
5. **Polymorphism**: Polymorphism means the ability of different objects to respond to the same message or method call in different ways. It allows objects of different classes to be treated as objects of a common superclass through method overriding and method overloading.
6. **Abstraction**: Abstraction refers to the process of simplifying complex systems by modelling classes appropriate to the problem, and working at the most relevant level of inheritance to create new classes

6. What is object

Ans: An object is an instance of a class, which serves as a blueprint or template defining its structure and behaviour.

7. What is class

Ans: A class is a blueprint or template for creating objects. It defines a data structure that encapsulates data (attributes) and behaviours (methods or functions) that operate on the data. Essentially, a class serves as a blueprint from which objects are instantiated.

8. What is encapsulation

Ans: Encapsulation is a fundamental principle in object-oriented programming (OOP) that describes the bundling of data (attributes) and methods (functions that operate on the data) into a single unit called a class. The class serves as a blueprint for creating objects (instances), which are instances of the class with their own unique data.

9. What is inheritance

Ans: Inheritance is another fundamental concept in object-oriented programming (OOP) where a new class (derived class or subclass) is created based on an existing class (base class or superclass). The derived class inherits the attributes and methods of the base class, allowing it to reuse the code of the base class and extend its functionality.

10. What is polymorphism

Ans: Polymorphism, in the context of object-oriented programming (OOP), refers to the ability of different objects to be treated as instances of a common superclass. It allows objects of different classes to be processed uniformly if they exhibit a certain behavior or share a common interface.

11. What is RDBMS

Ans: RDBMS stands for **Relational Database Management System**. It is a type of database management system (DBMS) that organizes data into tables, which are composed of rows and columns.

12. What is SQL

Ans: SQL (Structured Query Language) is a standardized programming language used for managing and manipulating relational databases. It provides a set of commands or statements for performing tasks such as querying data, inserting, updating, and deleting records, creating and modifying database schema (tables and indexes), controlling access permissions, and ensuring data integrity.

13. Write SQL Commands

Ans:

* **SELECT**: Retrieves data from a database.
* **INSERT**: Inserts new records into a table.
* **UPDATE**: Modifies existing records in a table.
* **DELETE**: Deletes records from a table.
* **CREATE**: Creates new database objects such as tables, views, indexes, etc.

14. Draw Usecase on Online book shopping

15. Draw Usecase on online bill payment system (paytm)

16. Write SDLC phases with basic introduction

17. Explain Phases of the waterfall model

18. Write phases of spiral model

19. Write agile manifesto principles

20. What is join?

21. Write type of joins.

22. Explain working methodology of agile model and also write pros and cons.

23. Draw usecase on Online shopping product using COD.

24. Draw usecase on Online shopping product using payment gateway.