1)what is SDLC and different phases in SDLC?

Ans: Software development life cycle (SDLC) is a process to develop the application

**Different phases like:**

**Requirement Analysis and planning :** Senior team members analyze the requirements/input given by customers/business users. They will check whether the requirement is feasible or not (can be done or not). They also identify the risks associated with project.

Note: this high level requirements will be written in BRD (Business Requirement document) by Business Analyst

Define/Design : in the define stage Business Analyst define more details about requirements (which are in BRD) in the form of SRS (software requirement specification) or Use Case diagram.

As part of design,

Senior Developers write High Level Design Document (HLD)

Developers write Low Level Design Document (LLD)

Seniors Tester write Test Planning document

Implementation/Development: Developers write the code for the requirements

Testers write test cases as per SRS

Testing : Execute the test cases what we prepared in previous stage

Deployment : Release the tested code to production

Maintenance : Support team monitoring the system that is running in production

2) what is waterfal in SDLC?

Ans: The **waterfall** model is a sequential design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a**waterfall**) through the phases of conception, initiation, analysis, design, construction, testing, production/implementation and maintenance.

what is the process in **agile** model

Ans: SDLC - **Agile Model**. **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 product. **Agile** Methods break the product into small incremental builds. These builds are provided in iterations.

what is scrum methodology

**Ans:Scrum** is an agile way to manage a project, usually **software**development. Agile **software** development with **Scrum** is often perceived as a **methodology**; but rather than viewing **Scrum** as**methodology**, think of it as a framework for managing a **process**.