* What is sdlc
* Sdlc is a systematic process for building software that ensures the quality and correctness of the software built. Sdlc process aims to produce high quality software that meets customer expectations. The system development should be complete in the pre-defined time frame and cost .sdlc consists of a detailed plan which explains how to plan,build and maintain specific software. Every phase of the sdlc life cycle has its own process and deliverable that feed into the next phase.

* What is software testing
* Software testing is the process used to identify the correctness,completeness,and quality of developed computer software.

* What is agile methodology
* The agile methodology is a way to manage a project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement at every stage.once the work begins,teams cycle through a process of planning,executing,and evaluating.

* What is srs
* A software requirements specification is a document that describes what the software will do and how it will be expected to perform,it also describes the functionally the product needs to fulfill all stakeholder needs.
* What is oops
* In the object-oriented model,interaction errors can be uncovered by scenario-based testing. This form of object oriented-testing can only test against the clients specifications, so interface errors are still missed. Class testing based on method testing. This approach is the simplest approach to test classes.

* Write basic concepts of oops
* class
* Object
* Inheritance
* Polymorphism
* Abstraction
* Encapsulation
* Association
* Aggregation
* Composition

* What is object
* Objects are key to understanding object-oriented technology.look around right now and you find many example of real-world objects.your pen,your bag,your bicycle real world objects share two characteristics they all have state [name,color] and behavior [barking,fetching] bicycles also have state [current gear, current speed] and behavior[changing gear,applying brakes].
* What is class
* This approach is the simplest approach to test classes. each method of the class performs a well defined cohesive function and can,therefore,be related to unit testing of the traditional testing techniques. Therefore all the methods of a class can be involved at least once to test the class.

* What is encapsulation
* Encapsulation is defined as the wrapping up of data under a single unit. It is the mechanism that binds together code and the data it manipulates. another way to think about encapsulation is, that it is a protective shield that prevents the data from being accessed by the code outside this shield.
* What is inheritance
* Inheritance is the process by which objects of one class acquire the properties of objects of another class.it supports the concepts of hierarchical classification.polymorphism means ability to take more than on form.

* 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. The most important aspect of the object is its behavior. A behavior is initiated by sending a message to the object.

* Draw use case on online book shopping

Open the website

Choose the book

Add to cart

Select the Payment Method

Add Shipping address

Confirm the amount

Success message

* Draw use case on online bill payment system(paytm)

Open paytm app

Select the your state

Select ele. Board

Select the district

Add service no.

Confirm the amount

Pay the bill

Save the receipt

* Write sdlc phases with basic introduction
* 1] requirements–in this phase,all the requirements are collected from the customer/client.they are provided in a document called businessmen requirements specification and system requirements specification. All the details are discussed with the customer in detail.
* 2]design–it has two steps:

1. High level design[hld] —it gives the architecture of software products.

2.low level [lld]-- it describes how each and every feature in the product should work and every component.

* 3]implementation-this is the longest phase

1]This phase consists of front end+middleware

+back-end.

2] in front-end –development of coding is done even seo settings are done.

3]in middleware–they connect both the front end and back end.

4]in the back -end– a database is created.

* 4]Testing—testing is carried out to verify the entire system. the aim of the tester is to find out the gaps and defects within the system and also to check whether the system is running according to the requirements of the customer.
* 5]Deployment–after successful testing,the product is delivered to the client, and even clients are trained on how to use the product.
* 6]Maintenance–once the product has been delivered to the client a task of maintenance starts as when the client will come up with an error the issue should be fixed from time to time.

* Explain phases of the waterfall model
* Analysis- all possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
* Design - the requirement specifications from first phase are studied in this phase and the system design is prepared. this system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
* Implementation- with inputs from the system design,the system is first developed in small programs called units, which are integrated in the next phase,each unit is developed and tested for its functionally,which is referred to as unit testing.
* Testing- all the units developed in the implementation phase are integrated into a system after testing of each unit.post integration the entire system is tested for any faults and failures.
* Maintenance- there are some issues which come up in the client environment .to fix those issues,patches are released.also to enhance the

product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

* Write phases of spiral model :
* Planning- determination of objectives ,alternatives and constraints
* Risk analysis- analysis of alternative and identification /resolution of risks
* Customer evolution - assessment of the results of engineering
* Engineering- development of the next level product
* Write agile manifesto principles :
* Satisfying customers through early and continuous delivery of valuable work.
* Breaking big work down into smaller tasks that can be completed quickly.
* Recognizing motivated individuals with the environment and support they need and trusting them to get the job done.
* Creating processes that promote sustainable efforts.
* Maintaining a constant pace for completed work.
* Welcoming changing requirements ,even late in a

Project.

* Explain working methodology of agile model and also write pros and cons :
* 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 interactions .
* Each iteration typically lasts from about one to three weeks.
* Pros :
* Is a very realistic approach to software development
* Promotes teamwork and cross training.
* Functionally 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.
* 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, functionally 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 documentation use -case.

* Draw use case on online shopping products using COD
* Draw use case on online shopping product using payment gateway