1. What is SDLC?

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

2. What is agile methodology?

Agile methodologies aim to deliver the right product, with incremental and frequent delivery of small chunks of functionality, through small cross-functional self-organizing teams, enabling frequent customer feedback and course correction as needed.

3. What is SRS?

A software requirements specification (SRS) is a document explaining how and what the software/system will do. It defines the features and functionality that the product requires to satisfy all stakeholders’ (business, users) needs.

The best SRS documents describe how the program communicates with the embedded hardware or specific software with unique coding culture. The chosen real-life users also account for nice SRS documents.

4. What is oops?

Object Oriented Programming is the principle of design and development of programs using modular approach.

5. Write Basic Concepts of oops.

[**Class**](https://www.topperskills.com/tutorials/oop/object-oriented-programming-classes-concepts.html)

The class is a model or blueprint or prototype of an object that defines or specifies all the properties of the objects.

[**An Object**](https://www.topperskills.com/tutorials/oop/object-oriented-programming-objects-concepts.html)

An object is an entity or instance of a class. The objects are mostly the physical entity but it can be a logical entity as well. Each object has state and behaviours.

[**Abstraction**](https://www.topperskills.com/tutorials/oop/object-oriented-programming-abstraction-concepts.html)

Abstraction means hiding the implementation and showing only functionality to the user.

[**Encapsulation**](https://www.topperskills.com/tutorials/oop/object-oriented-programming-encapsulation-concepts.html)

Encapsulation is the process of binding the data by making it private and functions to process that data into a single unit. Encapsulation provides security to the data.

[**Inheritance**](https://www.topperskills.com/tutorials/oop/object-oriented-programming-inheritance-concepts.html)

Inheritance is the process of acquiring or sharing the properties and behaviours of the base class into the sub class to achieve reusability.

[**Polymorphism**](https://www.topperskills.com/tutorials/oop/object-oriented-programming-polymorphism-concepts.html)

Polymorphism mean same name having different functionality.

6. What is SQL?

SQL is a language to operate databases; it includes database creation, deletion, fetching rows, modifying rows. It is a computer language for storing, manipulating and retrieving data stored in a relational database.

7. Write SQL Commands.

**Create:** Creates a new table in the database. It allows to specify the name of the table and the name of each column in the table.

CREATE TABLE table\_name (

column\_1 datatype,

column\_2 datatype,

column\_3 datatype

);

**Insert:** Used to add a new row to a table.

INSERT INTO table\_name (column\_1, column\_2, column\_3)

VALUES (value\_1, 'value\_2', value\_3);

**Alter:** Add columns to a table in a database.

ALTER TABLE table\_name

ADD column\_name datatype;

**Select:** Used to fetch data from a database.

SELECT column\_name

FROM table\_name;

**Where:** Filter the result set to include only rows where the following condition is true.

SELECT column\_name(s)

FROM table\_name

WHERE column\_name operator value;

**Delete:** Used to remove rows from a table.

DELETE FROM table\_name

WHERE some\_column = some\_value;

**Order By:** Sort the result set by a particular column either alphabetically or numerically.

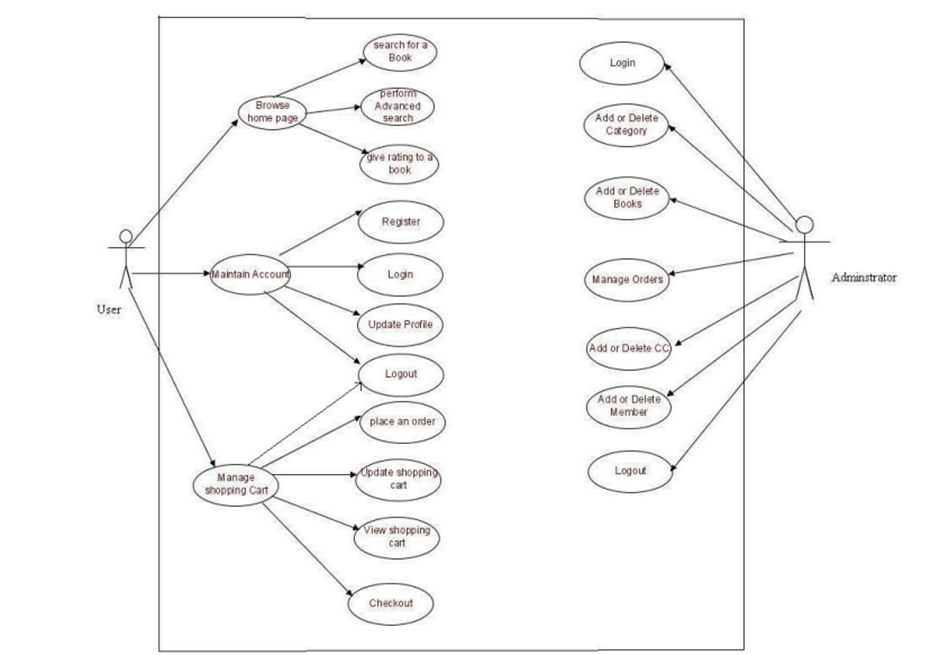
SELECT column\_name(s)

FROM table\_1

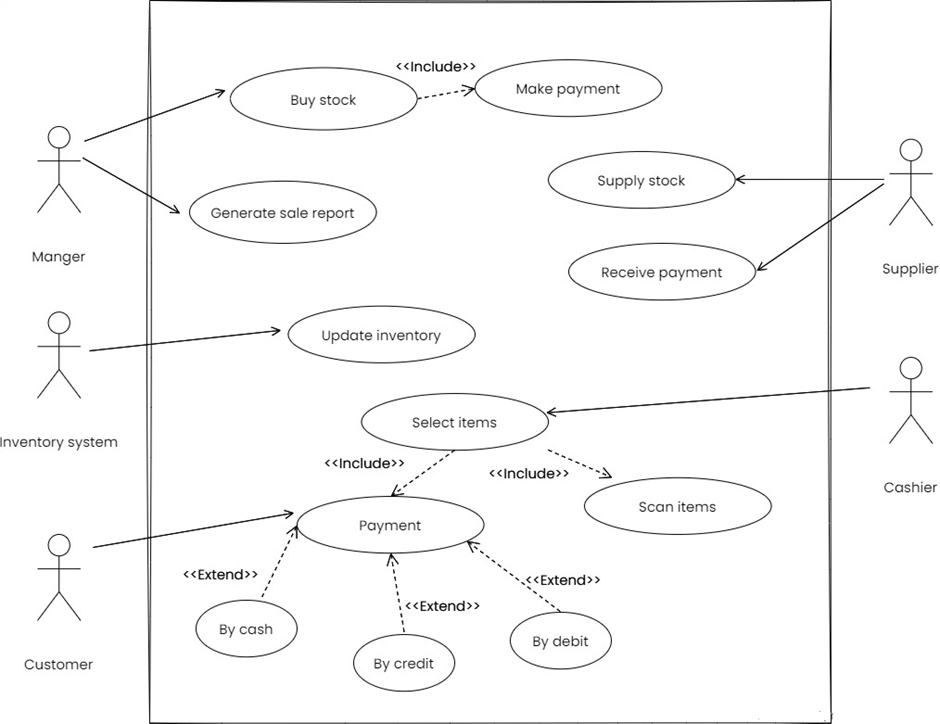
LEFT JOIN table\_2

ON table\_1.column\_name = table\_2.column\_name;

8. Draw Usecase on Online book shopping.



9. Draw Usecase on online bill payment system



10. Write SDLC phases with basic introduction.

Software Development Life Cycle (SDLC) is a framework that defines the steps involved in the development of software at each phase. It covers the detailed plan for building, deploying and maintaining the software.

**SDCL Phases**

**1.** **Planning and Requirement Analysis**

All the relevant information is collected from the customer to develop a product as per their expectation. Any ambiguities must be resolved in this phase only.

**2.** **Designing**

The requirement gathered in the SRS document is used as an input and software architecture that is used for implementing system development is derived.

**3.** **Implementation or Coding**

Implementation/Coding starts once the developer gets the Design document. The Software design is translated into source code. All the components of the software are implemented in this phase.

**4.** **Testing**

Testing starts once the coding is complete and the modules are released for testing. In this phase, the developed software is tested thoroughly and any defects found are assigned to developers to get them fixed.

**5.** **Deployment**

Once the product is tested, it is deployed in the production environment.

**6.** **Maintenance**

After the deployment of a product on the production environment, maintenance of the product.

11. Explain Phases of the waterfall mode.

The outcome of one phase is the input for the next phase. Development of the next phase starts only when the previous phase is complete.

* · First, Requirement gathering and analysis is done. Once the requirement is freeze then only the System Design can start. Herein, the SRS document created is the output for the Requirement phase and it acts as an input for the System Design.
* · In System Design Software architecture and Design, documents which act as an input for the next phase are created i.e. Implementation and coding.
* · In the Implementation phase, coding is done and the software developed is the input for the next phase i.e. testing.
* · In the testing phase, the developed code is tested thoroughly to detect the defects in the software. Defects are logged into the defect tracking tool and are retested once fixed. Bug logging, Retest, Regression testing goes on until the time the software is in go-live state.
* · In the Deployment phase, the developed code is moved into production after the sign off is given by the customer.
* · Any issues in the production environment are resolved by the developers which come under maintenance.

12. Write phases of spiral model.

It includes iterative and prototype approach.

**1.** **Planning:**

The planning phase includes requirement gathering wherein all the required information is gathered from the customer and is documented. Software requirement specification document is created for the next phase.

**2.** **Risk Analysis:**

In this phase, the best solution is selected for the risks involved and analysis is done by building the prototype.

**3.** **Engineering:**

Once the risk analysis is done, coding and testing are done.

**4.** **Evaluation:**

Customer evaluates the developed system and plans for the next iteration.

13. Write agile manifesto principles.

1. Satisfying customers through early and [continuous delivery](https://www.techtarget.com/searchitoperations/definition/continuous-delivery-CD) of valuable work.

2. Breaking big work down into smaller tasks that can be completed quickly.

3. Recognizing that the best work emerges from self-organized teams.

4. Providing motivated individuals with the environment and support they need and trusting them to get the job done.

5. Creating processes that promote sustainable efforts.

6. Maintaining a constant pace for completed work.

7. Welcoming changing requirements, even late in a project.

8. Assembling the project team and business owners on a daily basis throughout the project.

9. Having the team reflect at regular intervals on how to become more effective, then tuning and adjusting behaviour accordingly.

10. Measuring progress by the amount of completed work.

11. Continually seeking excellence.

12. Harnessing change for a competitive advantage.

14. What is join?

Combine two or more tables

15. Write type of joins.

1. inner join,

2. left outer join,

3. right outer join,

4. full outer join,

5. cross join.

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

Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning.

**Pros:**

1. In Agile methodology the delivery of software is unremitting.

2. The customers are satisfied because after every Sprint working feature of the software is delivered to them.

3. Customers can have a look of the working feature which fulfilled their expectations.

4. If the customers has any feedback or any change in the feature then it can be accommodated in the current release of the product.

5. In this methodology attention is paid to the good design of the product.

6. Changes in the requirements are accepted even in the later stages of the development.

**Cons:**

1. Sometimes in Agile methodology the requirement is not very clear hence it’s difficult to predict the expected result.

2. In few of the projects at the starting of the software development life cycle it’s difficult to estimate the actual effort required.

3. For complex projects, the resource requirement and effort are difficult to estimate

4. In Agile methodology the documentation is less.

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

