1.What is UML ?
UML (Unified Modeling Language) is a standard language for designing and documenting a system in an object-oriented manner, and (UML) is used by Technical Architects to communicate with the Developers.In other words,UML is a blue print of your source code.

2.What are the two basic parts of a Technical Architecture Design ?
The two basic parts of a Technical architecture design are :
(i)Static Architecture Design and
(ii)Dynamic Architecture Design

3.How are the Nine Diagrams in UML divided into ?
The Nine Diagrams in UML are divided into Static Diagrams and Dynamic Diagrams.
Static Diagrams are also called as Structural Diagrams.The Static Diagrams are :
(i)Class Diagram
(ii)Component Diagram
(iii)Deployment Diagram
(iv)Object Diagram
Dynamic Diagrams are also called as Behavioral Diagrams.The Dynamic Diagrams are :
(i) Activity Diagram
(ii)Collaboration Diagram
(iii)Sequence Diagram
(iv)StateChart Diagram
(v) Use Case Diagram

4.How to represent a Class diagram in UML ?
A Class Diagram is represented by defining a simple rectangle and then dividing this rectangle into three sections :
(i)The first section is where you define a className ;
(ii)The second section is where you define the properties of a class
(iii)The third section is where you define the methods of a class.

5.How to represent a Public property or a Public method in a Class Diagram ?
A Public property or a Public method in a Class Diagram is represented by a Plus(+) sign.

6.How to represent a Private property or a Private method in a Class Diagram ?
A Private property or a Private method in a Class Diagram is represented by a Minus(-) sign.

7.How to represent a Protected property or a Protected method in a Class Diagram ?
A Protected property or a Protected method in a Class Diagram is represented by a Hash(#) sign.

8.What is an Association in a Class Diagram ?
Association in a Class Diagram is the interaction between the classes,which (Association)can be shown by using the Arrow key between the classes.

9.What is Multiplicity in a Class Diagram ?
Multiplicity in a Class Diagram is termed as classes having multiple associations or one class can be linked to instances of many other classes.
In Multiplicity,the following notations are used :

(i) * means many instances.
(ii) (0,1) means zero or atleast one instance
(iii) (0…*) means zero or many instances
(iv) (1…*) means atleast one or many instances.

10.What is Aggregation Association and Composition Association in a Class Diagram ?
In a Class Diagram,Aggregation Association represents IS-A-Relationship whereas Composition Association represents HAS-A-Relationship.
Aggregated Association is shown by a Filled Diamond,whereas Composition Association is shown by an Empty Diamond.

11.What is Generalization and Specialization in a Class Diagram and how can we achieve ?
In Generalization and Specialization,we define the parent-child relationship between the classes.
We can achieve Generalization and Specialization by using Inheritance.

12.How to represent an abstract class ?
An Abstract class is represented by {{ abstract }}

What are the different kinds of relationships in a Class Diagram ?
The three different kinds of relationships in a Class Diagram are :
Instance Level Relationship : is classified into Association, Aggregation, and Compositon
Class Level Relationship : is classified into Generalization and Realization
General Relationship : is classified into dependency and Multiplicity

13.How to represent an Interface ?
An Interface is represented by << type >> .

14.How to represent an Interface in Action ?
An Interface in Action is represented by using " A Dotted line with a Circle".

15.What is a Use Case Diagram ?
Use Case Diagram is used in the Requirement Phase of a Software Lifecycle,which (Use Case Diagram) says "what will a system do ".

16.What are the three parts of a Use Case Diagram ?
The three parts of a Use Case Diagram are :
(i)The first part is "Scenario" ,which is a sequence of events which happens when an end user interacts with the system
(ii)The second part is "Actor",Actor is the End user of the system
(iii)The third part is "Use Case",which is a task or goal performed by an "Actor".

17.How many types an Actor is classified into in a Use Case Diagram ?
An Actor is classified into two types :
(i)Primary Actor : is an actor who actively participates in the Use Case and who is responsible for initiating the Use Case (Use Case or an Action)
(ii)Secondary Actor : is an actor who passively participates in the Use Case.

18.What is Extend and Include in a Use Case Diagram ?

Extend represents IS-A- Relationship , whereas Include represents HAS-A-Relationship.

19.What are the two views of representation in a Use Case Diagram ?
The two views of representation in a Use Case Diagram are :
(i)The use case diagrams
(ii)The details step table which describes how the use case works.

20.what is a Sequence Diagram ?
Sequence diagram shows the interaction between objects over a specific period of time.
Sequence diagram, shows the interaction between objects.
Sequence diagram shows “For what period of time the object is alive”.
Sequence diagram uses Dashed lines to represent the duration that the object is alive.
Sequence diagram uses Dark Arrow with dark Arrow Head to represent the Messages sent(request).
Sequence diagram uses Dotted Arrow to represent the Returned Messages (responses).
Sequence diagram has four types of Messages :
(i)Synchronous and Asynchronous Messages :Synchronous Messages are shown with Dark Arrow Head whereas Asynchronous Message are shown with Thin Arrow Head
(ii)Recursive Messages
(ii)Message Constraints: If you want the Message Interaction to be happened if only certain conditions are fulfilled then we make use of Message Constraints and is shown with “ [If] ”.
(iv)Message Branching: If you want the Message Interaction to be happened if only certain conditions are fulfilled deepening on the other conditions then we make use of Message Branching.

21.What is an Object Diagram ?
Object Diagram gives a pictorial representation of Class Diagram at any point of time.
Object Diagram should only be drawn to represent complicated relationships between objects.

22.What is a Collaboration Diagram ?
Collaboration Diagram provides the same information as shown by Sequence Diagram but it (Collaboration Diagram) shows it (the same information)in a different way.
In Sequence Diagram we pay more attention to the time and sequence order,but in Collaboration Diagram we pay more attention to Interaction Messages between the objects.

23.What is an Activity Diagram ?
An Activity Diagram is used to represent the complicated process flows (that is, a lot of IF conditions or a lot of parallel processings are going on)in a system.
In Activity Diagram,the start of an activity is represented by a Dark circle , the end of an activity is represented by a Dark circle inside a White circle ,and the activities are represented by oval rectangles.

24.What is a State Diagram ?
A State Diagram represents the different states that an object goes through during its lifecycle and also represents how an object responds to different events while the object goes through different states during its lifecycle.

25.What is a Component Diagram ?
A Component Diagram achieves the same objective like Package Diagram.
A Package Diagram shows logical grouping of use cases and classes while a Component Diagram shows interactions between the components.

26.What is a Deployment Diagram ?
A Deployment Diagram represents an overall static view of how software and hardware nodes in the application are

installed or configured.
Deployment diagram shows what the hardware is and which components (theses components are shown by Component Diagram) are installed on which hardware (this hardware is shown by a solid box).You can have more than one component installed on a single hardware.

27.What is a Stereo Diagram ?
Stereo Types can be used with anyone of the Nine UML Diagrams to define custom variations in the existing UML model.
Stereo Types are represneted with << Text >> sign.

28.What is a Package Diagram ?
Packages are like folders in a system which allows you to logically group UML Diagrams,which (Packages) makes complex UML Diagrams readable.
In actual projects,Packages are used to logically group “use cases” and “classes”.So we can say there are two types of Package Diagrams ; one is
Use Case Package Diagram and the other is Class Package Diagram.