1. **What is Modeling in software development?\*\*\***

**-**

**Modeling** is the process of creating a simplified version of the software system to understand how it will work.

It helps developers and users to **visualize the structure, function, and behavior** of the system before building it.  
>>Example: Like drawing a blueprint of a house before starting construction.

1. **What should a model represent?\*\*\***

**-**

A good model should represent:

The **information** that software will use and change

The **architecture** (overall structure) and **functions** that allow changes to happen

The **features** or requirements that users expect

The **behavior** of the system while it is working

>>Example: A railway booking model shows how users search, select, and book tickets through the system.

1. **What levels does modeling cover?\*\*\***

**-**

At the **customer level**, it shows what the system will do from the user’s point of view.

At the **technical level**, it helps developers understand how to build the system.  
>>Example: First, a car design is shown to the buyer; then, engineers make technical diagrams for production.

1. **What is the difference between Modeling and Design?\***

**-**

| **Modeling** | **Design** |
| --- | --- |
| Shows how the system will look or work | Plans how the system should work |
| Gives a simple picture or sample of the system | Decides the features and layout |
| Helps people understand the idea easily | Helps developers know what to build |
| Can be a prototype or example | Is the full plan for building the real system |
| Example: Demo version of a website | Example: Sketch showing button positions |

1. **What is Analysis Modeling?\*\*\***

**-**

Analysis modeling helps understand **what the system needs to do** before it is built.  
There are two main approaches:

**Structured Analysis** – Separates **data** and **processes**.  
👉 Example: A payroll system model with employee data and payment process shown separately.

**Object-Oriented Analysis** – Uses **classes** to combine data and behavior.  
👉 Example: A “Student” class with name, roll number, and attendance function.

1. **What is Data Modeling?\*\*\***

**-**

Data modeling is used when a system needs to **store, manage, or use complex data**.

It shows all the **data objects**, their **attributes**, and how they are **connected**.  
>>Example: In a hospital system, data objects like Patient, Doctor, and Appointment are connected to show how they relate.

1. **What is a Data Object?& What are Attributes of a Data Object?\*\*\***

**-**

A data object is a **collection of related information** that the system needs to understand and use.

It represents things like a file, event, person, place, or report.  
>>Example: “Order” is a data object with date, items, and total amount.

It only contains data — it does not include actions or methods.

Attributes of data of object:

**Attributes** are the details or properties that describe a data object.

>>Example: For a “Book” data object, the attributes can be title, author, and ISBN.

1. **What is the difference between Data Objects and Object-Oriented Classes?/Are Data Objects and Object-Oriented Classes the same?\*\*\***

**-**

They are not same but similar.

| **Data Object** | **Object-Oriented Class** |
| --- | --- |
| Only holds **data and attributes** | Holds **data and also functions (methods)** |
| No actions are included | Can perform actions using methods |
| Used mainly for **storing information** | Used for **storing and processing information** |
| 👉 Example: “Book” with title, author, ISBN | 👉 Example: “Book” with title + method like borrow() |

1. **What is Class-Based Modeling?& What are the elements of Class-Based Modelling?\*\*\***

**-**

Class-based modeling is used in object-oriented design. It focuses on:

**Objects** the system will manage (like users, products)

**Operations (methods)** to work with those objects

**Relationships** between objects

**Collaborations** between different classes

>>Example: In a school system, the "Teacher" class may interact with the "Classroom" class to assign subjects.

Elements of Class-Based Modelling:

**Classes and Objects** – Define real-world things in the system (e.g., Student, Teacher)

**Attributes** – Describe each class’s properties (e.g., student name, ID)

**Operations** – Actions done by or to the class (e.g., markAttendance())

**CRC Models** – Show class roles and how they interact

**Collaboration Diagrams** – Visualize how classes communicate

**Packages** – Group related classes for better organization

>>Example: In an e-commerce app, “Order”, “Cart”, and “Product” classes may be grouped in a “Shopping” package.