Sequence Model for Hotel Management System

Name	Roll Number	Gr. Number	Div/Batch
Atharva Parikh	332044	21910716	B/B2
Sidhant Khamankar	332029	21910598	B/B2
Namrata Kadav	332040	21911024	B/B2

Objective

Identify at least 5 major scenarios (sequence flow) for your system.
Draw Sequence Diagram for every scenario by using advanced notations using UML 2.0

Theory

- ### What are Sequence Diagrams ?

  Sequence diagrams describe interactions among classes in terms of an exchange of messages over time and are used to explore the logic of complex operations, functions or procedures.
  In this diagram, the sequence of interactions between the objects is represented in a step-by-step manner.

- ### Purpose of Sequence Diagrams ?

  These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process.

- ### Benefits of Sequence Model ?

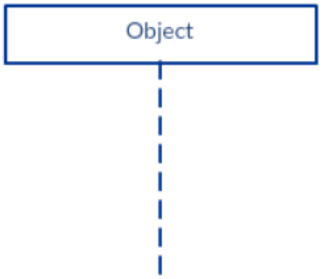
- Represent the details of a UML use case.
- Model the logic of a sophisticated procedure, function, or operation.
- See how objects and components interact with each other to complete a process.
- Plan and understand the detailed functionality of an existing or future scenario.

- How to draw ?
  - A sequence diagram has two dimensions: The vertical dimension shows the sequence of messages in the chronological order that they occur; the horizontal dimension shows the object instances to which the messages are sent.
  - A sequence diagram is straightforward to draw. Across the top of diagram, identify the class instances (objects) by putting each class instance inside a box .
  - If a class instance sends a message to another class instance, draw a line with an open arrowhead pointing to the receiving class instance and place the name of the message above the line.
  - Optionally, for important messages, we can draw a dotted line with an arrowhead pointing back to the originating class instance; label the returned value above the dotted line.

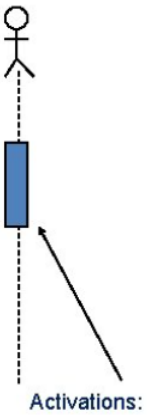
- Basic Sequence Diagram Notations

- **Actor :** A type of role played by an entity that interacts with the subject (e.g., by exchanging signals and data)

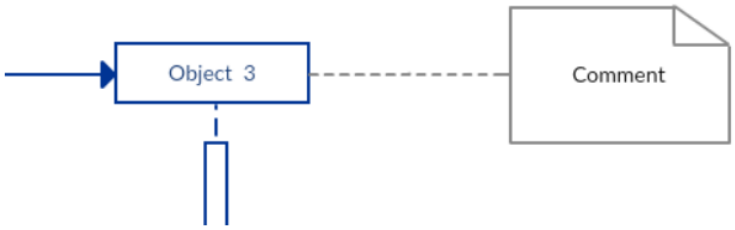
- **Lifeline :** A lifeline represents an individual participant in the Interaction.



- **Activations :** A thin rectangle on a lifeline) represents the period during which an element is performing an operation.

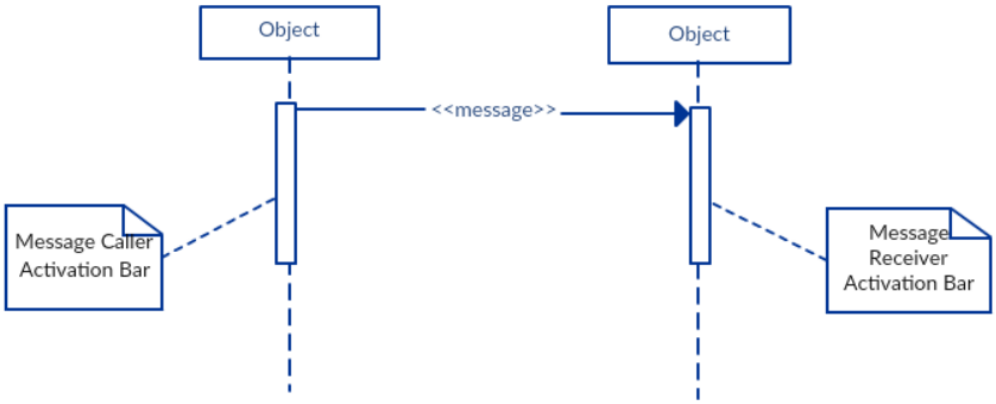


- **Comment :** UML diagrams generally permit the annotation of comments in all UML diagram types. The comment object is a rectangle with a folded-over corner as shown below. The comment can be linked to the related object with a dashed line.

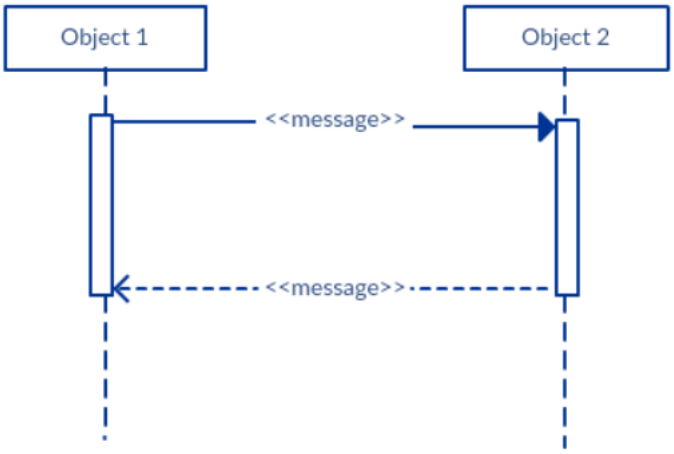


Types of Messages in Sequence Diagrams

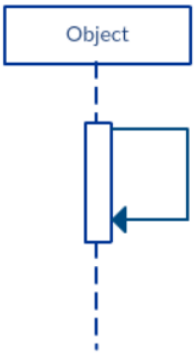
- **Call Message :** A message defines a particular communication between Lifelines of an Interaction.



- **Return Message :** A message defines a particular communication between Lifelines of an Interaction.

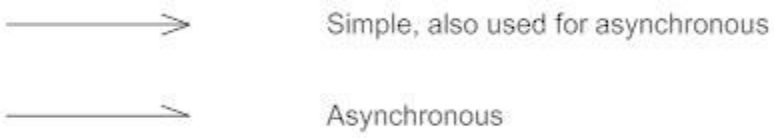


- **Self Message :** A message defines a particular communication between Lifelines of an Interaction.



- **Synchronous Message :** A synchronous message requires a response before the interaction can continue. It's usually drawn using a line with a solid arrowhead pointing from one object to another.

- **Asynchronous Message :** Asynchronous messages don't need a reply for interaction to continue. Like synchronous messages, they are drawn with an arrow connecting two lifelines; however, the arrowhead is usually open and there's no return message depicted.



- **Delete Message :** This is a message that destroys an object. It can be shown by an arrow with an x at the end.

Sequence Diagram