

### ASSIGNMENT NO: 3 and 4

**Aim:** Object Oriented Analysis and design using UML diagrams: Draw Use case and Class Diagram using Open-Source Tool.

Problem Statement:

<<Students should write the problem statement chosen by them>>>

The tasks we have to do are:

1. You will have to identify the main entities (objects) for this system.
2. You will have to find out the relationships between these objects.
3. You will have to find the necessary attributes and functions that need to be associated with each object to implement the functionality mentioned above.
4. You will make a final comprehensive diagram show and all objects and their relations along with their attributes and functions.

#### Objectives

1. To learn the relationships and notions of Use case diagram.
2. To learn the relationships and notions of Class diagram.

#### Theory:







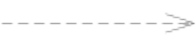
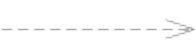
##### 1. Use Case Diagram:

A view of a system that emphasizes the behavior as it appears to outside users. A use case model partitions system functionality into transactions ('use cases') that are meaningful to users ('actors').

- a) Create Actors to represent classes of people, organizations, other systems, software or devices that interact with your system or subsystem.
- b) Create Use Cases for each of the goals that each actor seeks to achieve with the system.
- c) Use Associations to link actors to use cases.
- d) Use <<include >> , <<extend>> and generalization to show the relationships among the use cases

Following are the notations used to draw UML diagrams. It is added here for reference.

Construct	Description	Syntax
-----------	-------------	--------

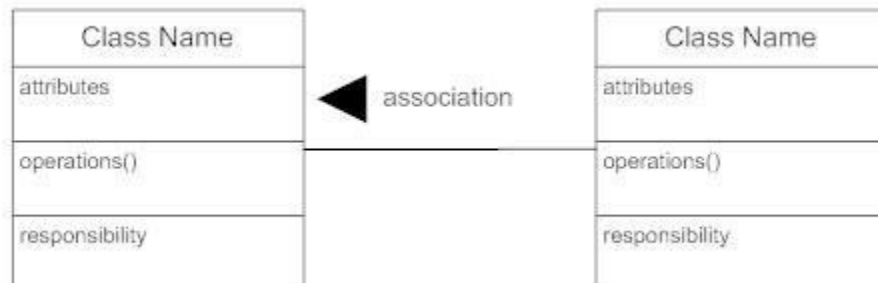
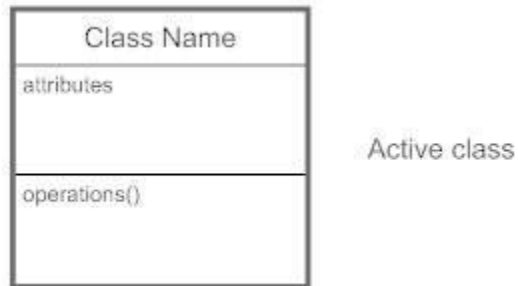
<b>Use case</b> 	A sequence of actions, including variants, that a system (or other entity) can perform, interacting with actors of the system.	
<b>Actor</b> 	A coherent set of roles that user of use cases play when interacting with these use cases.	
<b>System boundary</b> 	Represents the boundary between the physical system and the actors who interact with the physical system.	
<b>Association</b>	The participation of an actor in a use case. i.e., instance of an actor and instances of a use case communicate with each other.	
<b>Generalization</b>	A relationship between a more general use case and a more specific use case.	
<b>Extend</b> 	A relationship from an <i>extension</i> use case to a <i>base</i> use case, specifying how the behavior for the extension use case can be inserted into the behavior defined for the base use case.	<<extend>>
<b>Include</b> 	A relationship from a <i>base</i> use case to an <i>inclusion</i> use case, specifying how the behavior for the inclusion use case is inserted into the behavior defined for the base use case.	<<include>>

## 2. Class Diagram:

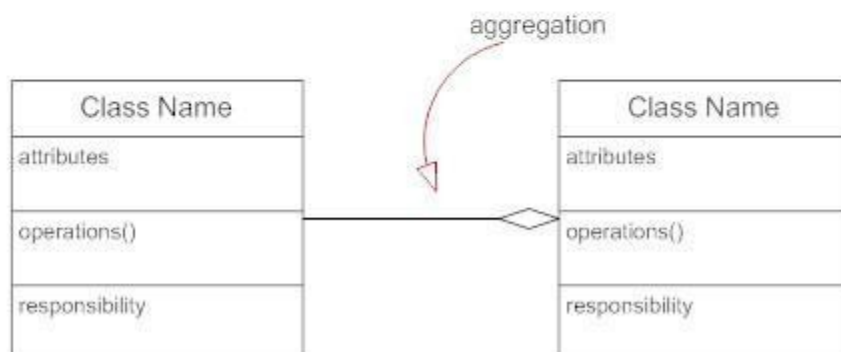
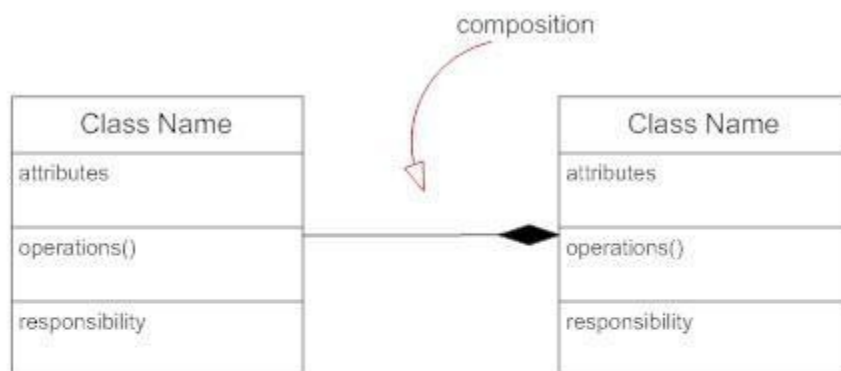
A class model captures the static structure of a system by characterizing the objects in the system.

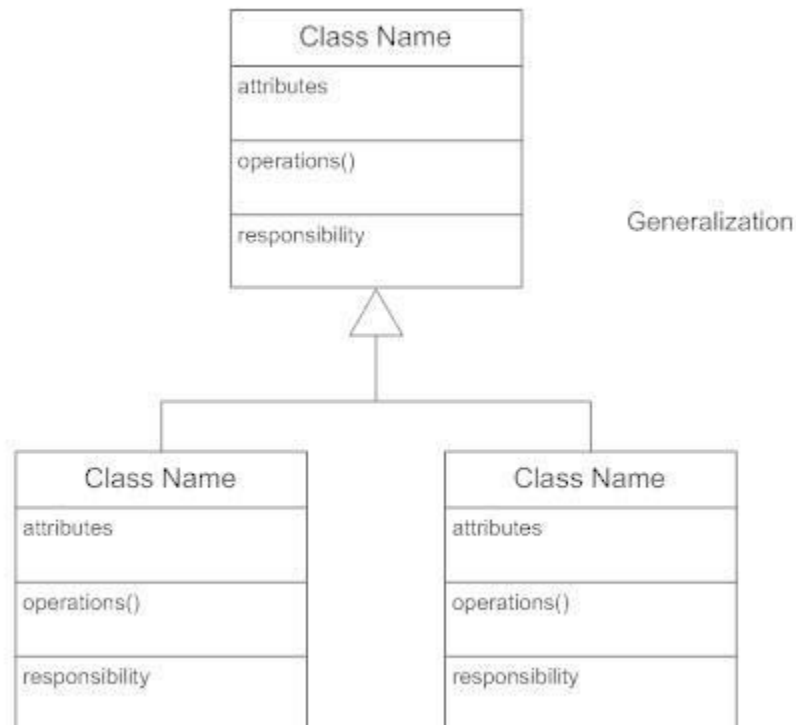
### Procedure:

- Write a problem statement, clearly defining the scope of the system.
- List out nouns as the probable classes from the problem statement.
- Draw the classes of the nouns focus should be on understanding relationships.
- Draw model of the system- the class model. Show relationships, attributes and multiplicity in the class model. Data types not required.



### Composition and Aggregation





**Platform:** Microsoft Windows 7, Open Source draw.io tool or any UML design tool.

**Input:** Problem statement scenario.

**Output:** Soft copy of Class diagram, Use Case Diagram.

**Conclusion:**

**FAQs:**

1. What kind of relationships classes have? Explain all relationships with examples.
2. Explain any 2 terminologies used in Use case diagrams.
3. Explain the aggregation and composition in diagram?