A.I.S.S.M.S
INSTITUTE OF INFORMATION TECHNOLOGY
Kennedy Road, Near R.T.O., Pune - 411 001.

* Laboratory Practice - IV (OOMD) - Experiment Number - 2.

Name :- Kaustubh Shrikant Kabra.
Class :- Fourth Year Engineering (BE)
Div :- A
Roll Number :- 37
Batch :- B-2
Department :- Computer Department
College :- AISSMS's IOIT.

Title :-

Class Digram.

Problem Definition :-

Draw basic class diagrams to identify and describe key concepts like classes, types in your system and their relationships.

Prerequisite :-

Software Analysis Skills, Object Orientation and development, Software Development life cycle, Types of Model and diagrams.

Software Required :-

Visual Paradigm 17.0 / Star UML.

Hardware Requirement :-

Windows 7 or above or Linux, RAM - 4GB and more, ROM - 128GB and more.

A.I.S.S.M.S
INSTITUTE OF INFORMATION TECHNOLOGY
Kennedy Road, Near R.T.O., Pune - 411 001.

Learning Objectives :-

① To learn and draw class diagram.
② To identify and describe key concept like classes and their relationship

Outcomes :-

① We will understand the concept of class diagram.
② We will learn to draw the class diagram for any system.

Theory :-

Class Diagram-

The class diagram is one of the types of UML diagrams which is used to represent the static diagram by mapping the structure of the systems using classes, attributes, relations, and operations between the various objects. A class diagram has various object classes; each has three-part :-

I) The first partition contains a class name which is the name of the class or entity which is participated in the activity.

II) The second partition contains class attributes that show the various properties of the class.

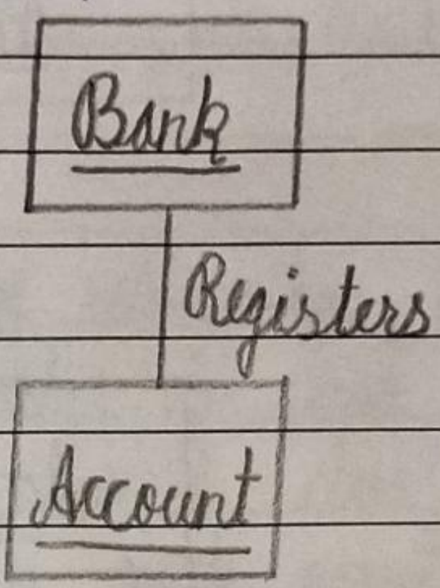
III) The third partition contains class operations which shows various operations performed by the class, relationships shows the relation between two classes.

Relationships -

In a class diagram, it is necessary that there exists a relationship between the classes

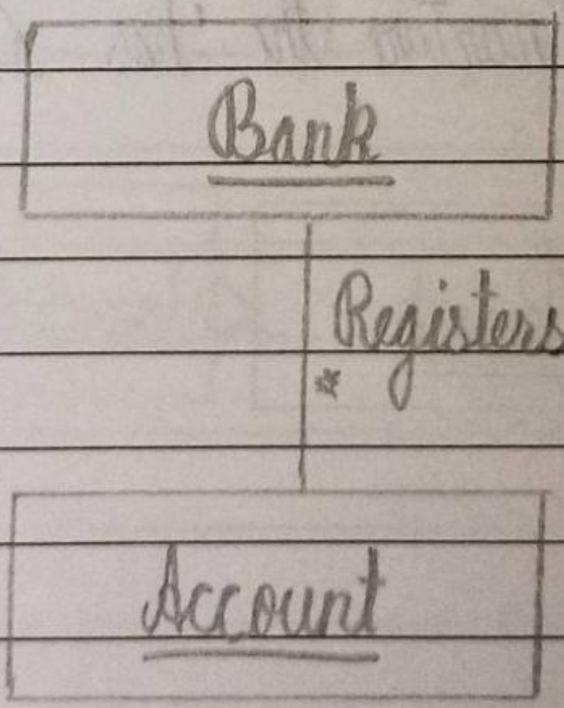
① Association -

Between two other classes in an association relationship, an association class forms a part of it. Additional information about the relationship could be obtained by attaching the association relationship with the association class.



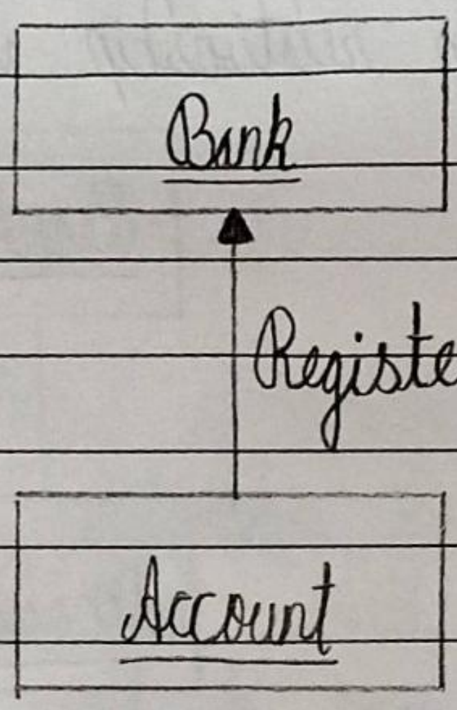
② Multiplicity -

The number of elements or cardinality could be defined by multiplicity. It is one of the most misunderstood relationship which describes the number of instances allowed for a particular element by providing an inclusive non-negative integers interval. It has both lower and upper bound.



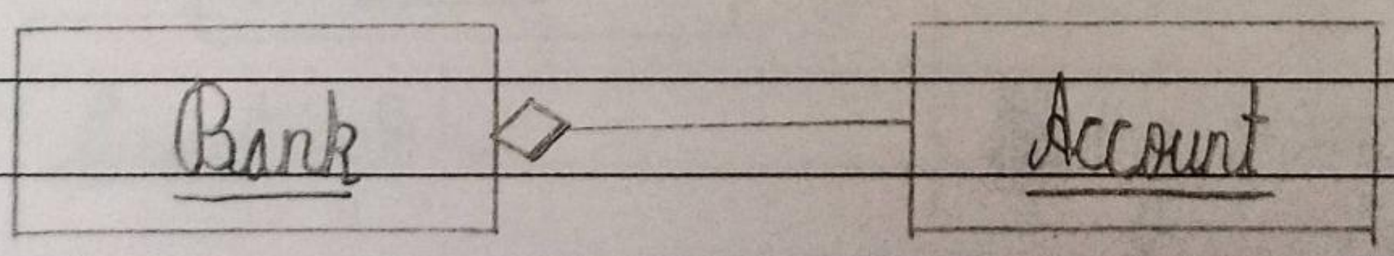
③ Directed Association -

This is a one-directional relationship in a class diagram that ensures the flow of control from one to another classifier. The navigability is specified by one of the association ends. The relationship between two classifier could be described by naming any association. An arrow indicates the direction of navigation.



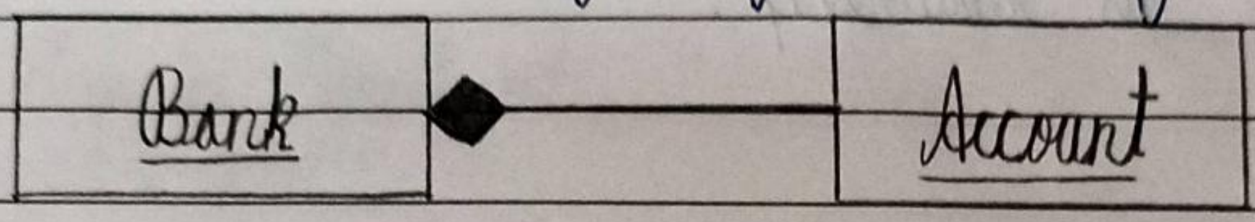
④ Aggregation -

In this type of relationship, a more complex object is created by assembling different objects together. The interaction within the different groups of objects is defined by Aggregation. The integrity of the objects is protected, and the response of the assembled objects is decided by the control object. In Aggregation, the classes nurture the 'has a' relationship.



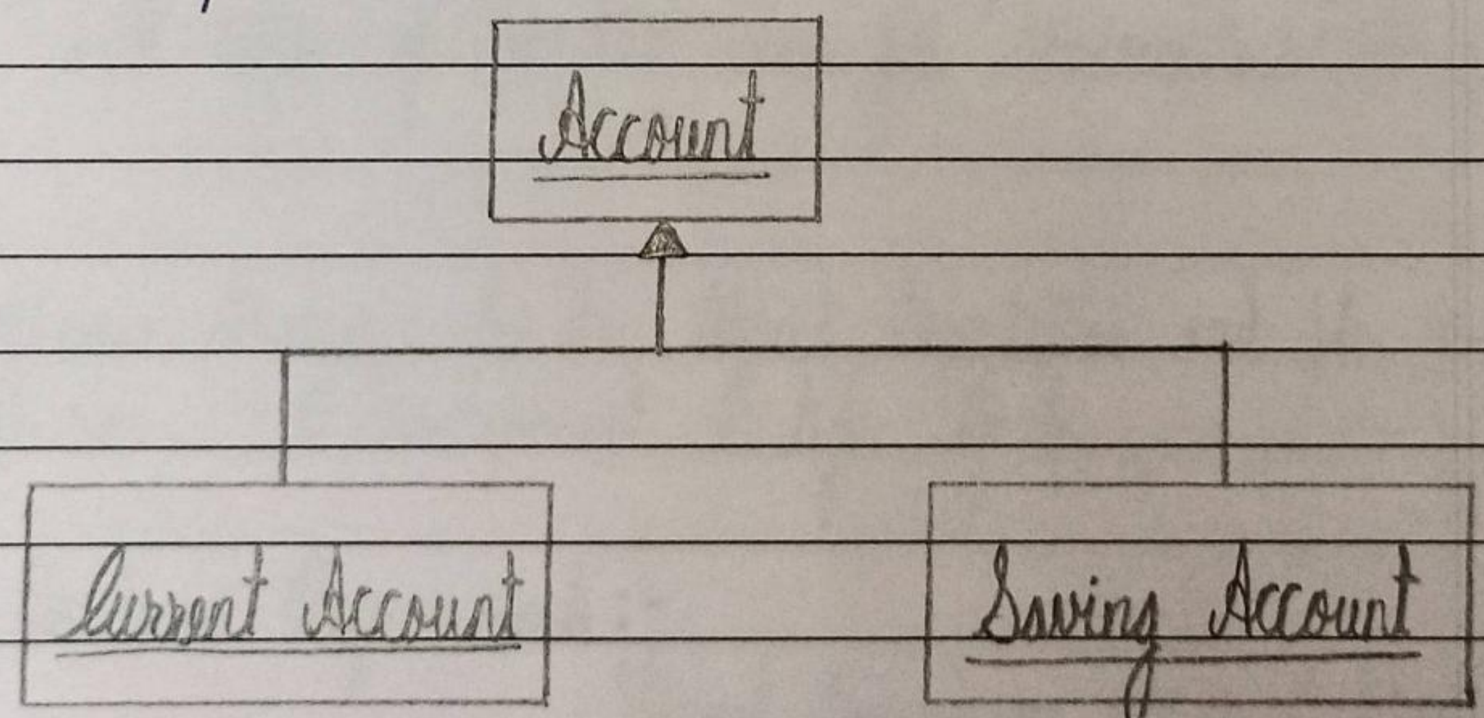
⑤ Composition-

It is a form of aggregation which represents the whole-part relationship. Here, the part classifier lifetime is dependent on the whole classifier lifetime. In a class, a strong life-cycle is represented by the composition relationship. There is usually a one-direction flow of data here. It is generally indicated by a solid line.



⑥ Generalization-

In this kind of relationship, the child model is based on the parent model. The relationship is used to describe various use-case diagrams and ensures that child class receives the properties present in the parent. There could be single parent, multiple children, or multiple parents, single child characteristics in this relationship. There are no names in the generalization relationship. It is also known as the 'is a relationship'.



⑦ Realization :-

The behavior of one model element is realized by the specified behavior

Conclusion :-

In this way, we learned how to draw a class diagram and its relationship.