**Practical No 3**

**Aim: Study of Class & Object Diagram**

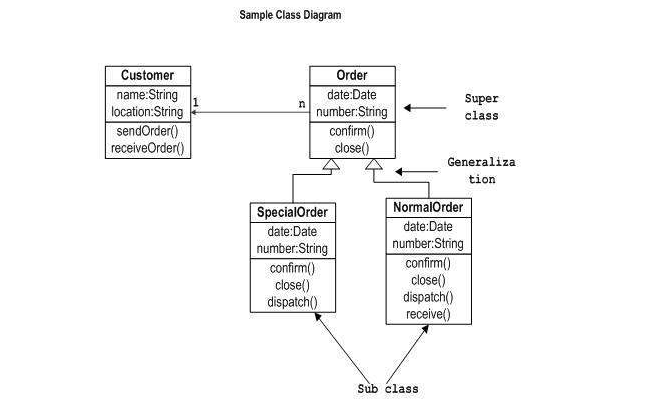
**Class Diagram**

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

The purpose of class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction. UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application, however class diagram is a bit different. It is the most popular UML diagram in the coder community.

**How to draw class diagram?**

Class diagrams are the most popular UML diagrams used for construction of software applications. It is very important to learn the drawing procedure of class diagram. Class diagrams have a lot of properties to consider while drawing but here the diagram will be considered from a top-level view. Class diagram is basically a graphical representation of the static view of the system and represents different aspects of the application. A collection of class diagrams represents the whole system.



**Where to use class diagram?**

Class diagram is a static diagram and it is used to model the static view of a system. The static view describes the vocabulary of the system.

Class diagram is also considered as the foundation for component and deployment diagrams. Class diagrams are not only used to visualize the static view of the system but they are also used to construct the executable code for forward and reverse engineering of any system.

Generally, UML diagrams are not directly mapped with any object-oriented programming languages but the class diagram is an exception.

Class diagram clearly shows the mapping with object-oriented languages such as Java, C++, etc. From practical experience, class diagram is generally used for construction purpose.

In a nutshell it can be said, class diagrams are used for −

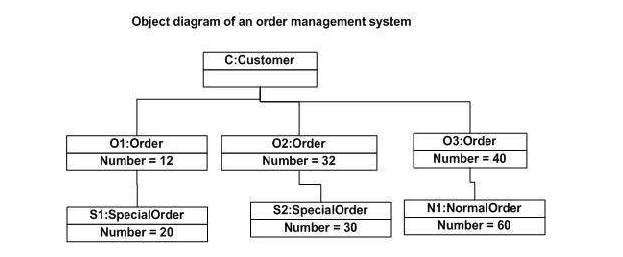
* Describing the static view of the system.
* Showing the collaboration among the elements of the static view.
* Describing the functionalities performed by the system.
* Construction of software applications using object-oriented languages.

**Object Diagram**

Object diagrams are derived from class diagrams so object diagrams are dependent upon class diagrams. Object diagrams represent an instance of a class diagram. The basic concepts are similar for class diagrams and object diagrams. Object diagrams also represent the static view of a system but this static view is a snapshot of the system at a particular moment. Object diagrams are used to render a set of objects and their relationships as an instance. The purpose of a diagram should be understood clearly to implement it practically. The purposes of object diagrams are similar to class diagrams. The difference is that a class diagram represents an abstract model consisting of classes and their relationships. However, an object diagram represents an instance at a particular moment, which is concrete in nature.

**How to draw an object diagram?**

We have already discussed that an object diagram is an instance of a class diagram. It implies that an object diagram consists of instances of things used in a class diagram. So, both diagrams are made of same basic elements but in different form. In class diagram elements are in abstract form to represent the blue print and in object diagram the elements are in concrete form to represent the real-world object. To capture a particular system, numbers of class diagrams are limited. However, if we consider object diagrams then we can have unlimited number of instances, which are unique in nature. Only those instances are considered, which have an impact on the system.



**Where to use object diagram?**

Object diagrams can be imagined as the snapshot of a running system at a particular moment. Let us consider an example of a running train

Now, if you take a snap of the running train then you will find a static picture of it having the following −

* A particular state which is running.
* A particular number of passengers. which will change if the snap is taken in a different time

Here, we can imagine the snap of the running train is an object having the above values. And this is true for any real-life simple or complex system.

In a nutshell, it can be said that object diagrams are used for −

* Making the prototype of a system.
* Reverse engineering.
* Modelling complex data structures.
* Understanding the system from practical perspective.

**A. Keeping the classes ‘Worker’, ‘Activity’, ‘Skills’, ‘Project’ and ‘Work’ in mind, how will you depict the following information: -**

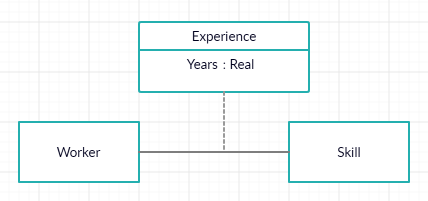
**1. When a worker has a skill, the years of experience is maintained for the relationship using the association class.**

**2. An activity may have no more than one previous activity and any number of next activities. Using these role names (previous and next activity), we can show how activities are ordered. Given an activity, you can only determine its next activities (if it has any) but not its previous activity (if it has any).**

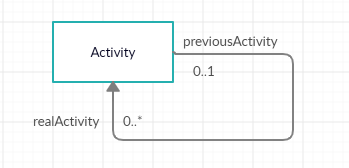
**3. A worker is not simply associated with a set of skills, but a worker has skills. Specially, a worker must have 3 or more skills, and any numbers of workers may have same skill.**

**4. Projects and activities are specific type of work.**

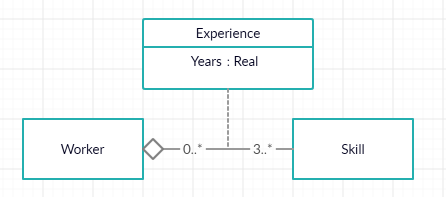
**Ans 1:**



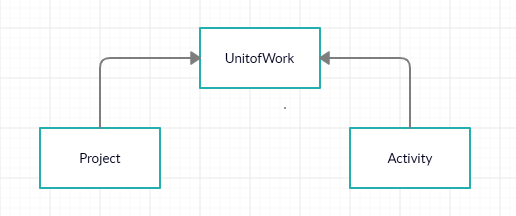
**Ans 2:**



**Ans 3:**

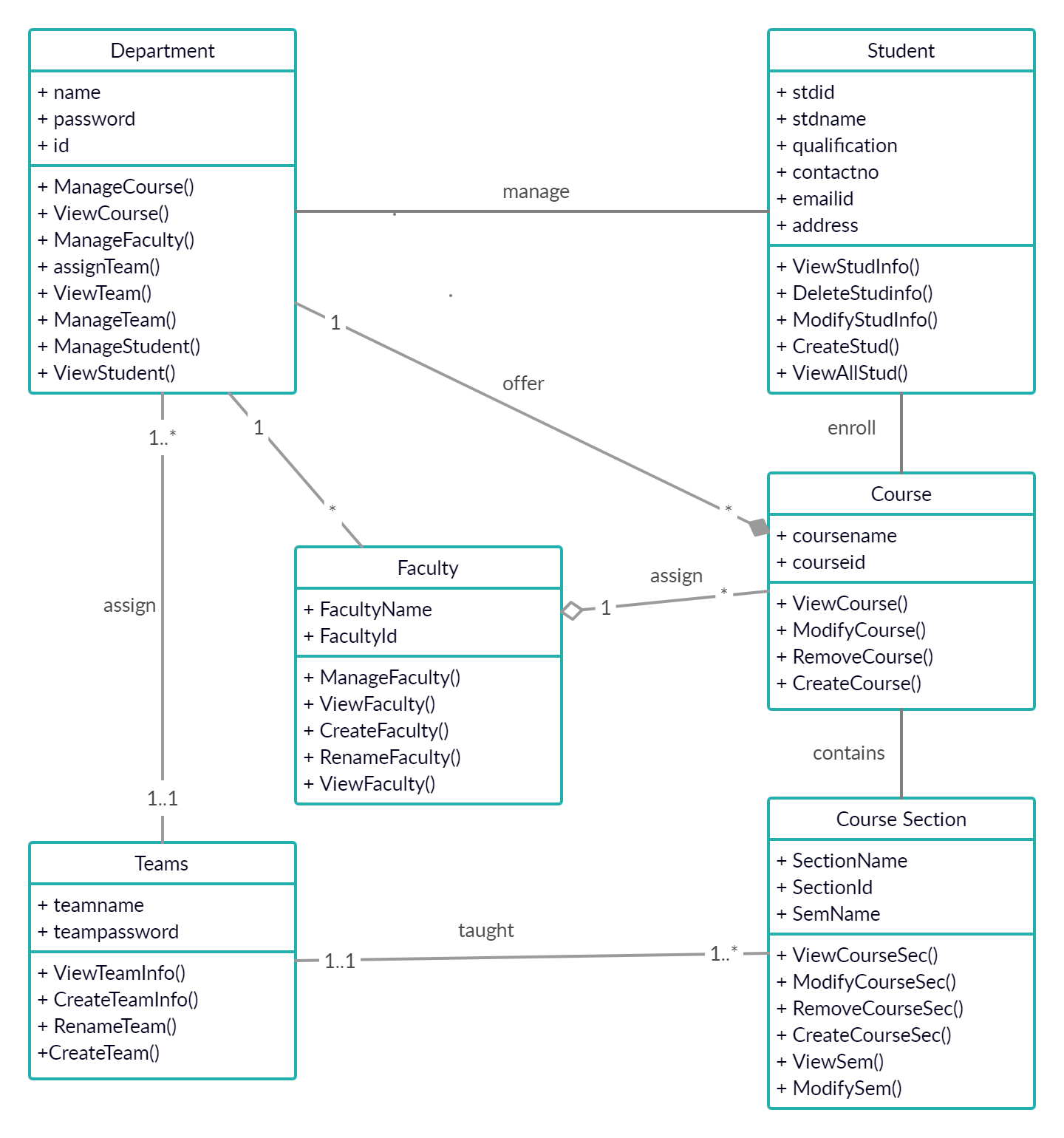


**Ans 4:**

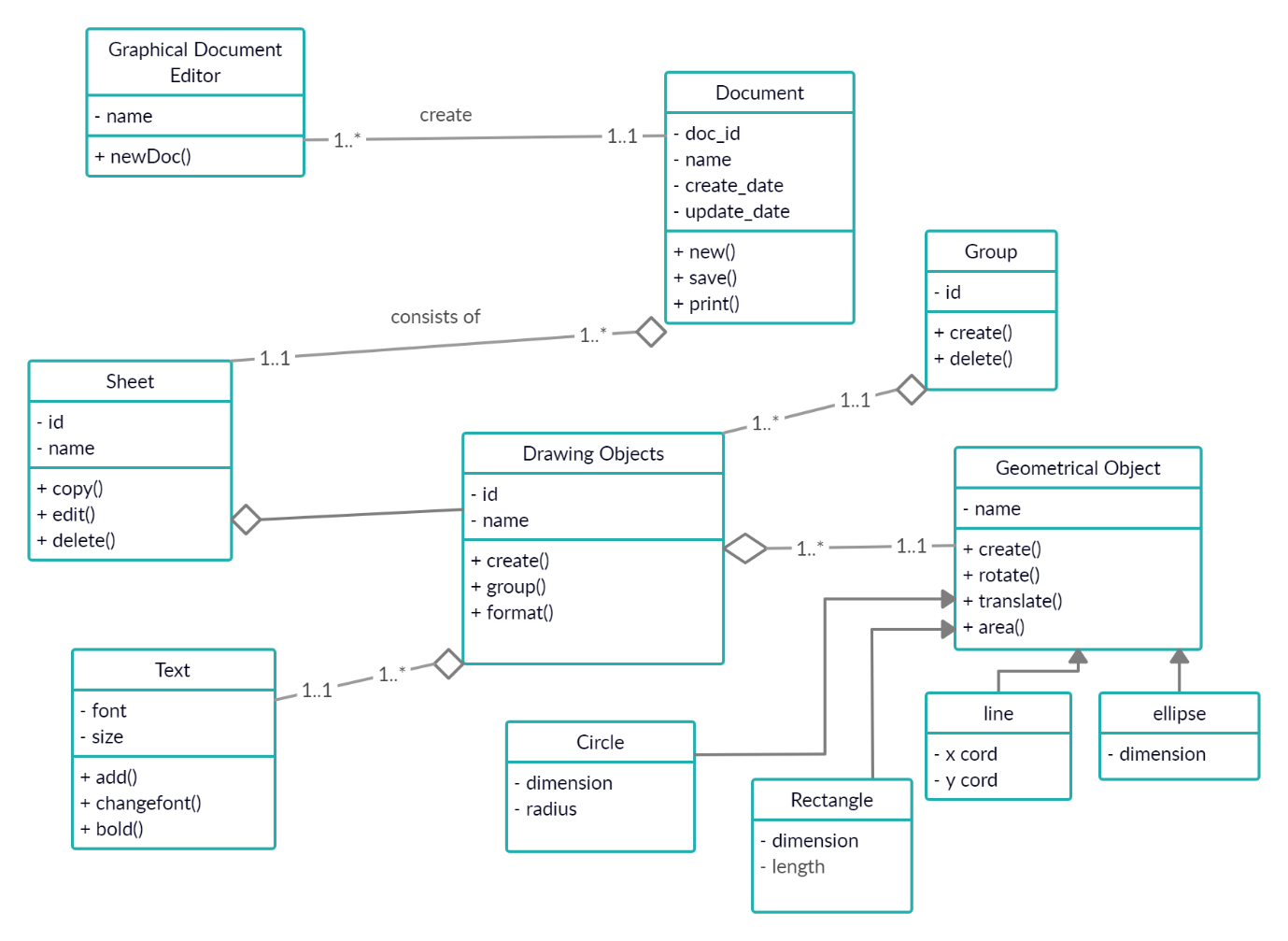


**B. Draw a class diagram for following:**

**A department offers several courses. A faculty member of the department usually teaches many course sections. But some semesters they do not. Each course section must have at least one faculty teaching it. But sometimes course sections are taught by teams. Furthermore, to make sure all course sections are similar, one faculty member is assigned as course coordinator. A faculty member can be coordinator for many courses. The students are allowed to enrol in more than one course section at a time.**

****

**C. Prepare a class diagram for a Graphical document editor, which supports grouping which is a concept used in a variety of graphical editors. Assume that a document is composed of several sheets each sheet contains drawing objects including text, Geometrical objects and groups. A group is simply a set of drawing objects possibly including other groups. A group must contain at least 2 drawing objects. A drawing object may or may not be a member of group. If it is a member then it can be a member of only 1 group. Geometrical objects include circular, elliptical, rectangular, lines and squares. You may consider circle as special case of ellipse and square has a special case of rectangle.**

****

**Conclusion: We have studied the details about the class and object diagram.**