



# EXPERIMENT - 5

## Software Engineering Lab

### Aim

To draw the structural view diagram for the system: Class diagram, object diagram.

Syeda Reeha Quasar

14114802719

4C7

## EXPERIMENT – 5

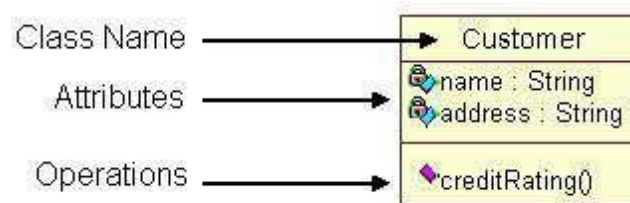
### Aim:

To draw the structural view diagram for the system: Class diagram, object diagram.

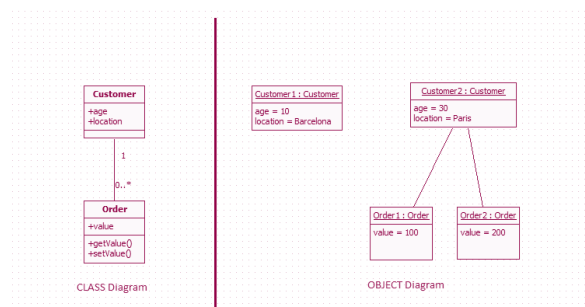
### Theory:

**Class diagrams** are widely used to describe the types of objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagrams describe three different perspectives when designing a system, conceptual, specification, and implementation.

Classes are composed of three things: a name, attributes, and operations. Below is an example of a class:



**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.



## Performance Instruction:

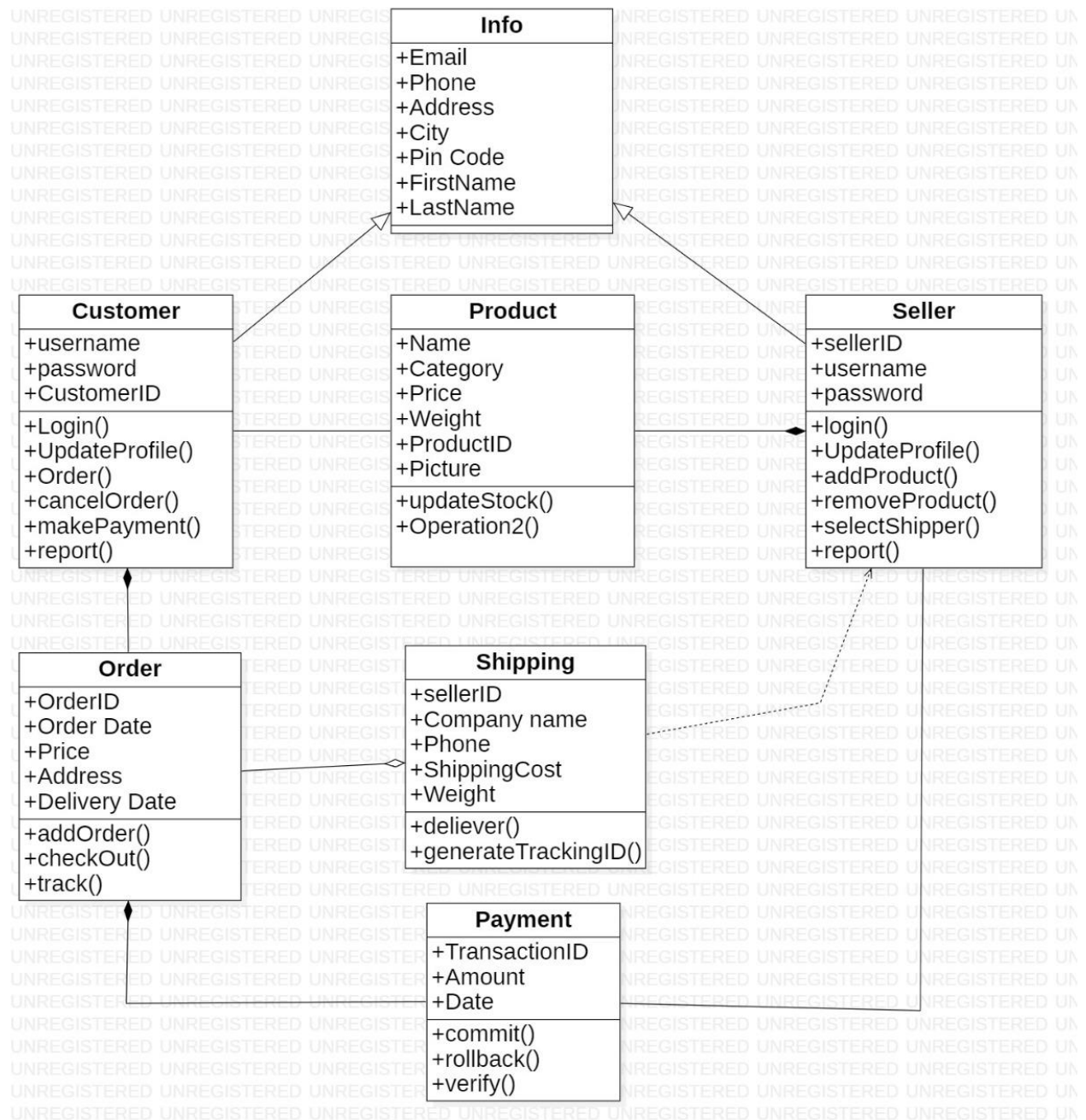
### To draw class diagram

1. Identify various elements such as classes, member variables, member functions etc. of the class diagram
2. Draw the class diagram as per the norms

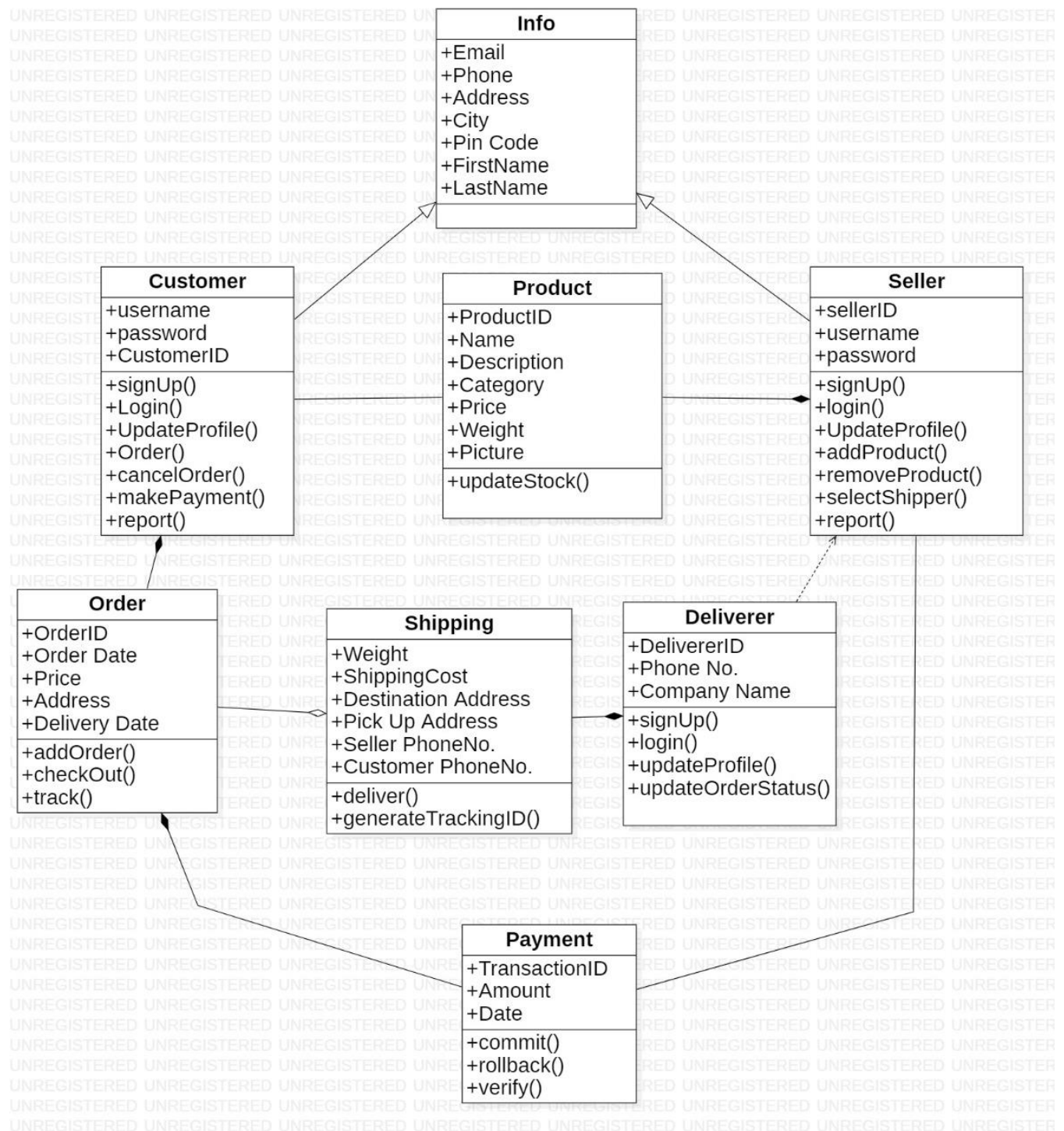
### To draw object diagram

1. First, analyse the system and decide which instances have important data and association.
2. Second, consider only those instances, which will cover the functionality.
3. Third, make some optimization as the number of instances are unlimited.

Class Diagram vs Object Diagram		
More Information Online: <a href="http://WWW.DIFFERENCEBETWEEN.COM">WWW.DIFFERENCEBETWEEN.COM</a>		
	Class Diagram	Object Diagram
DEFINITION	A type of static structural diagram that describes structure of the system by showing classes, their attributes, methods and relationship among classes.	A type of static structural diagram that shows a complete or partial view of the structure of a modeled system at a specific time.
MAIN REPRESENTATION	Defines classes or blue prints and show how they relate to each other.	Shows the objects and their relationships.
METHOD OF WRITING	In a class diagram, the class name starts with uppercase. e.g. - Student	In an object diagram, the object name is in lowercase, and it is underlined. e.g., s1: Student

**Output:****Class Diagram**

## Object Diagram





## Viva Questions

### 1. Explain class diagram?

Ans.

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

### 2. Explain four types of relationship used in class diagram?

Ans.

#### **Inheritance (or Generalization):**

A generalization is a taxonomic relationship between a more general classifier and a more specific classifier. Each instance of the specific classifier is also an indirect instance of the general classifier.

#### **Association**

Associations are relationships between classes in a UML Class Diagram. They are represented by a solid line between classes. Associations are typically named using a verb or verb phrase which reflects the real-world problem domain.

#### **Realization**

Realization is a relationship between the blueprint class and the object containing its respective implementation level details. This object is said to realize the blueprint class.

#### **Dependency**

An object of one class might use an object of another class in the code of a method. If the object is not stored in any field, then this is modelled as a dependency relationship.

### 3. Explain terms classes, interfaces, collaborations and dependency?

Ans.

#### **Classes**

A template for creating objects and implementing behavior in a system. In UML, a class represents an object or a set of objects that share a common structure and behavior. They're represented by a rectangle that includes rows of the class name, its attributes, and its operations

#### **Interfaces**

Interfaces are model elements that define sets of operations that other model elements, such as classes, or components must implement. An implementing model element realizes an interface by overriding each of the operations that the interface declares.

#### **Collaborations**

It is used to show the relationship between the objects in a system. Both the sequence and the collaboration diagrams represent the same information but differently. Instead of showing the flow of messages, it depicts the architecture of the object residing in the system as it is based on object-oriented programming. An object consists of several features.

#### **Dependency**

An object of one class might use an object of another class in the code of a method. If the object is not stored in any field, then this is modeled as a dependency relationship.

### 4. Explain object diagram?

Ans.

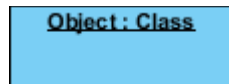
Object diagrams represent an instance of a class diagram. The basic concepts are similar for class diagrams and object diagrams. Object diagrams also represent the static view of a system but this static view is a snapshot of the system at a particular moment.

## 5. Explain symbols used in it?

Ans.

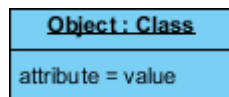
### Object Names:

- Every object is actually symbolized like a rectangle, that offers the name from the object and its class underlined as well as divided with a colon.



### Object Attributes:

- Similar to classes, you are able to list object attributes inside a separate compartment. However, unlike classes, object attributes should have values assigned for them.



### Links:

- Links tend to be instances associated with associations. You can draw a link while using the lines utilized in class diagrams.

