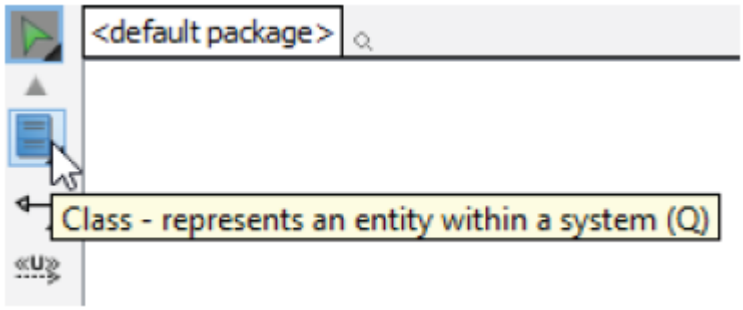
Lab 6: Design Class Diagram

- A class diagram shows the objects that are required and the relationships between them. Since it provides detailed information about the properties and interfaces of the classes, it can be considered as the main model.

Creating a class diagram

1. To create a class diagram, select **Diagram > New** from the toolbar. In the **New Diagram** window, select **UML Diagrams > Class Diagram**. Click **OK** to confirm.
2. To create class, click **Class** on the diagram toolbar and then click on the diagram. Finally, name the newly created class.



3. To create attribute, right click the class and select **Add** > **Attribute** from the pop-up menu.
4. Similarly, to create operation, right click the class and select **Add** > **Operation** from pop-up menu.
5. To create association from class, click the **Association** -> **Class** resource beside it and drag.

6. Drag to empty space of the diagram to create a new class or drag to an existing class to connect to it. Release the mouse button to create the association.
7. Common UML relations supported in a class diagram are:
   - Association
   - Generalization
   - Aggregation
   - Composition
   - Multiplicity

Case Study: All Phones Shop (APS)

Student Name: **Student ID:**

System description: All Phones Shop (APS) is a company that sells three types of phones: *smartphone*, *normal mobile phone*, and *land phone*. APS currently uses a small computer-based system called PoB that manages the payroll of its employees, all payments made by clients, and the financial information of the APS client. PoB is actually a standard, commercial-off-the-shelf application. APS has two types of **clients**: *individual*, and *corporate*.

To **buy** or **sell** a phone, the client must first register if he/she does not have any registration with APS. In order to register, the client has to provide name and address. APS first checks if the client already exists. If not, the system creates a registration, assigns a unique registration number, PoB is advised by APS with the registration number. PoB then opens a financial account of the client with the registration number. All payments of the client are handled by PoB and recorded in the financial account. However, APS keeps only the registration information such as name, address, the registration number, the invoice(s) if the client has bought any item from APS, and the item information if the client sells any item. For all future selling and buying activities with APS, the client has to use the same registration number.
A **registration** can only be removed by corporate clients. In this case, APS checks if there is any unpaid invoice attached with the client registration. PoB is informed if a client registration is removed. The individual client can only create registration but cannot remove any registration. APS does not handle any login operation; it is assumed all login functions are checked before any operation is executed.

To sell an item, the client first selects the type of item that he/she wants to sell. Under smartphone category, there are two sub-categories: *Apple iPhone* and *Samsung Galaxy*. The client then provides model, make, capacity, and color. The client has the option to sell the item either on **auction** (bidding), or on **fixed price sale**. For the auction, a starting price and the end of the auction time is specified; and for the fixed price sale, a price is given. For both types of sale (fixed price and auction), the system assigns an ID to the item, records and attaches the item information with the client registration. For the catalog entry, APS then automatically creates a display image based on the item description, asks the client (seller) to approve it, and displays the item for sale if approved by the client. Otherwise, the system re-creates another image, and displays it to the client for approval. Once the client approves the display image, APS includes it in its product catalog. Clients can also remove their items after display but before sale; in that case, a **penalty** of 10% of the starting price (for auction) or the fixed price of the item is charged to the client, and an advice on this penalty will be sent to PoB.

Clients can also buy items on auction, or on fixed price sale. For the fixed price items, the client selects item(s). The system creates a shopping basket for the selected item(s). The system saves the shopping basket and attaches this with the client registration. The client then checks out, and each item in the basket is considered "sold" and removed from the product catalog. For auction, clients can bid a price for the selected item before the end of auction time. A bid submitted by a client must be greater than the last offered bid (price). All bids submitted by a client are recorded in the registration of the client. When the auction time ends, the client with the last bid will be notified as the winner. The item is considered "sold" and removed from the product catalog.

For either type of sale, APS prepares an **invoice** (bill) based on the item(s). Once the client confirms the invoice, it is attached with the client registration with the status "unpaid".
For all payments, PoB informs APS with the client registration number and the invoice number. The system finds the invoice and assigns the invoice as "paid". If a client fails to pay to PoB within 7 days, a

penalty of 10% of the price is charged to the client, an advice on this penalty is sent to PoB, and the sale is cancelled by APS.

Task: Develop design class diagram of the system.
***Lab topic(s)**:* Class model and operations
***Lab objective(s)**:* Understanding of

- Object
- Association
- Generalization
- Aggregation
- Composition
- Attribute
- Multiplicity
- Object behavior
- Method
- Parameters of methods

***Lab activities*:**

Activity	**Resources and notes**	**Estimated time**
1. The lab instructor explains the lab activities.	• Lab-6	10 minutes
2. Students write their name and SID (See first page on where to write these)	• First page of this document	5 minutes
3. **Task 1**: Each student reviews the system description again. Identify five major classes and two attributes for each identified class for the system.	• System description • Use Table 6.1: Template for Task 1 to record the identified classes and attributes.	10 Minutes
4. **Task 2**: Develop a class diagram with major classes, their key attributes, multiplicity, and associations, generalization, and aggregation. Do not include methods.	• Visual Paradigm	20 minutes
5. **Task 3**: Each student reviews the system description and class diagram developed earlier. Identify two methods with parameters for three major classes as enlisted in template 6.1 for Task 1 in this document.	• System description • Use Table 6.2: Template for Task 1 • UML Visual paradigm	30 Minutes
6. **Task 4:** Identify at least one method with parameters for the two classes in table 6.3.	• System description • Use Table 6.3: Template for Task 3 to record the identified methods for the two classes.	10 minutes

7. **Task 5:** Update your class diagram developed earlier with the defined methods in Task 1 and Task 2.	• Visual Paradigm • Class diagram developed earlier	30 minutes
1. **Task 6**: Find more relevant methods to the system for the classes in your class diagram. Answer the following questions: • Does your class diagram with those methods tell the story? • Is your class diagram consistent with the use case specifications? 2. Does your class diagram capture the main properties of the system description?	• System description • Visual Paradigm • Updated Class diagram	15 minutes
3. Save your domain model. Copy the model and paste it at the end of this document.	• Visual paradigm • This document.	5 minutes
4. Submit this document to the lab instructor and leave the lab clean and tidy.	• This document with solutions to 6 tasks.	5 minutes

Table 6.1: Template for Task1

Class name	Attribute 1	Attribute 2

Task-2: Paste your class diagram <<Here>>:

Table 6.2: Template for Task 3.

Class	Methods
Client	
Item	
Bid	

Table 6.3: Template for Task 4:

Class	Method
Invoice	
Catalog	

Task 5 and 6: Paste your updated class diagram with the update methods <<Here>>: