

**Topic 1: Class Diagram – Structure Overview (Any 4)**

**Problem Statement 1:**

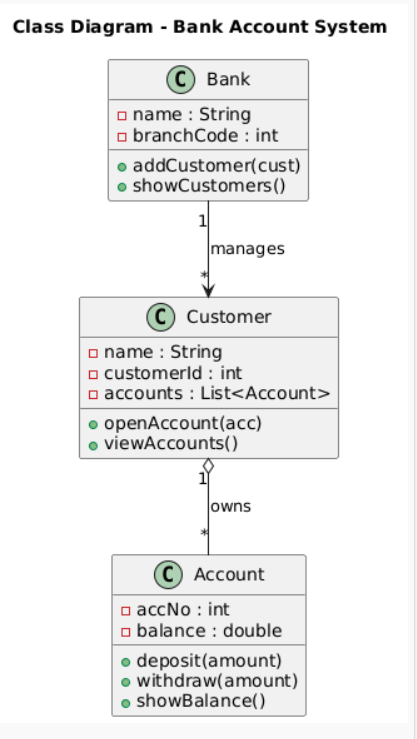
Draw a **Class Diagram** for a simple **Bank Account System** with classes such as Account, Customer, and Bank. Show attributes and methods for each class, and represent the relationships between them.

**Hints:**

● Use **association** between Bank and Customer.

● Represent **aggregation** between Customer and Account.

● Include **visibility** symbols (+, -, #) for attributes and methods.



**Topic 2: Object Diagram – Runtime Instances**

**Problem Statement 2:**

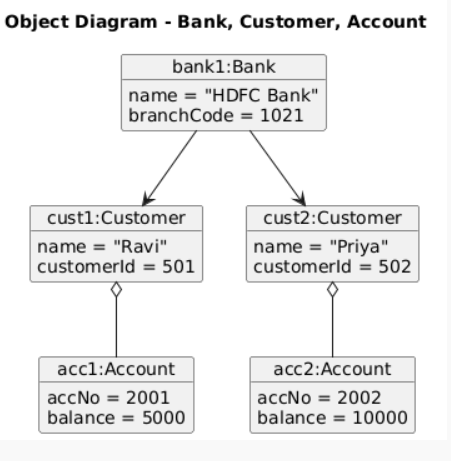
Create an **Object Diagram** showing instances of Customer, Account, and Bank classes. Display how specific objects are linked during runtime (e.g., one bank has multiple customers, each with an account).

**Hints:**

● Show object names like cust1:Customer, acc1:Account.

● Include attribute values in objects (e.g., balance = 5000).

● Demonstrate **links** between instantiated objects.



1



**Topic 3: Sequence Diagram – Method Interaction Over Time**

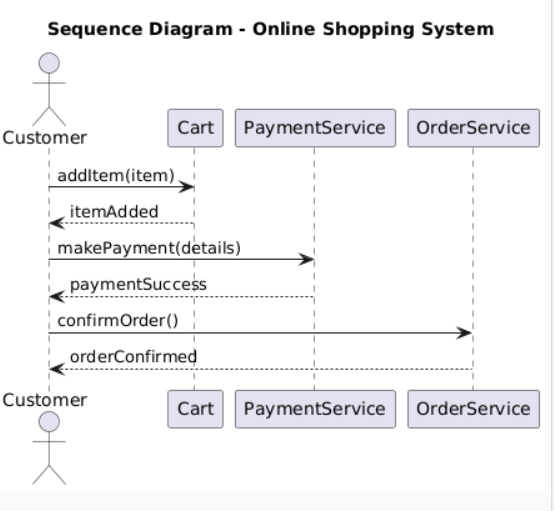
**Problem Statement 3:**

Draw a **Sequence Diagram** for an **Online Shopping System** showing interactions among Customer, Cart, PaymentService, and OrderService when a customer places an order.

**Hints:**

● Use **lifelines** for each participant.

● Show **method calls** (e.g., addItem(), makePayment(), confirmOrder()). ● Include **return messages** and activation boxes.



**Topic 4: Use Case Diagram – Functional Overview**

**Problem Statement 4:**

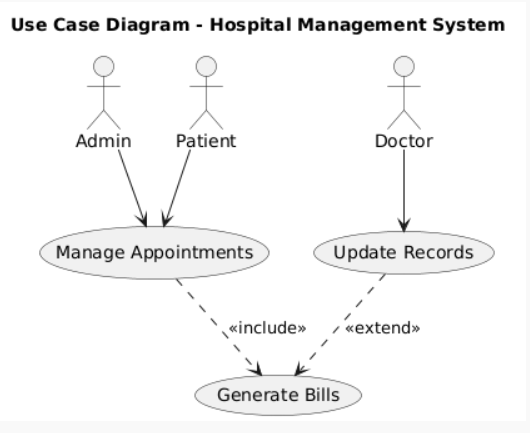
Create a **Use Case Diagram** for a **Hospital Management System** where an Admin, Doctor, and Patient interact with features like Manage Appointments, Update Records, and Generate Bills.

**Hints:**

● Use **actor symbols** for external users.

● Draw **include** and **extend** relationships between use cases.

● Label all relationships clearly.



2



3