**Practical No 1:-**

**Study and implementation of class diagrams.**

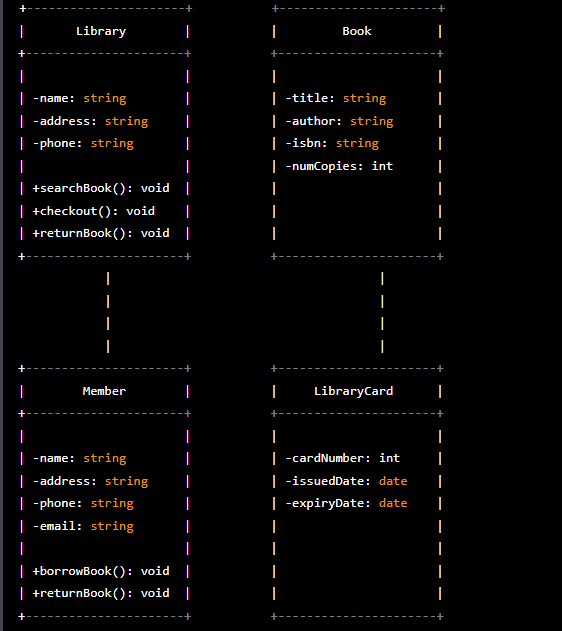
Class diagrams are a type of UML (Unified Modeling Language) diagram used to model object-oriented systems. Class diagrams represent the structure of a system by showing the classes, their attributes, and their relationships with other classes.

A typical class diagram consists of the following components:

1. Class: A class is a blueprint for creating objects. It has attributes (data) and methods (functions) that define its behavior.
2. Attributes: Attributes are the data members of a class that define its state.
3. Operations: Operations are the methods of a class that define its behavior.
4. Visibility: Visibility is the degree to which a class member is accessible from other classes. It can be public, private or protected.
5. Inheritance: Inheritance is a relationship between two classes where one class (the child or derived class) inherits the attributes and methods of another class (the parent or base class).
6. Association: Association is a relationship between two classes where one class uses or depends on another class.
7. Aggregation: Aggregation is a type of association where one class contains a reference to another class. The contained class is not dependent on the container class and can exist independently.
8. Composition: Composition is a type of aggregation where one class is composed of one or more instances of another class. The contained class is dependent on the container class and cannot exist independently.

To implement a class diagram, you can use a programming language that supports object-oriented programming, such as Java or C++. You can create a class for each element in the class diagram and define its attributes and methods. You can also define the relationships between classes using inheritance, association, aggregation, and composition.

Here is an example of a class diagram for a library management system:



In this example, there are four classes: Library, Book, Member, and LibraryCard. The Library class has attributes for the name, address, and phone number of the library, as well as methods for searching for books, checking out books, and returning books. The Book class has attributes for the title, author, ISBN, and number of copies of each book, as well as a method for searching for books.

The Member class has attributes for the name, address, phone number, and email of each library member, as well as methods for borrowing and returning books. The LibraryCard class has attributes for the card number