

Object Oriented Programming

Object Oriented Programming

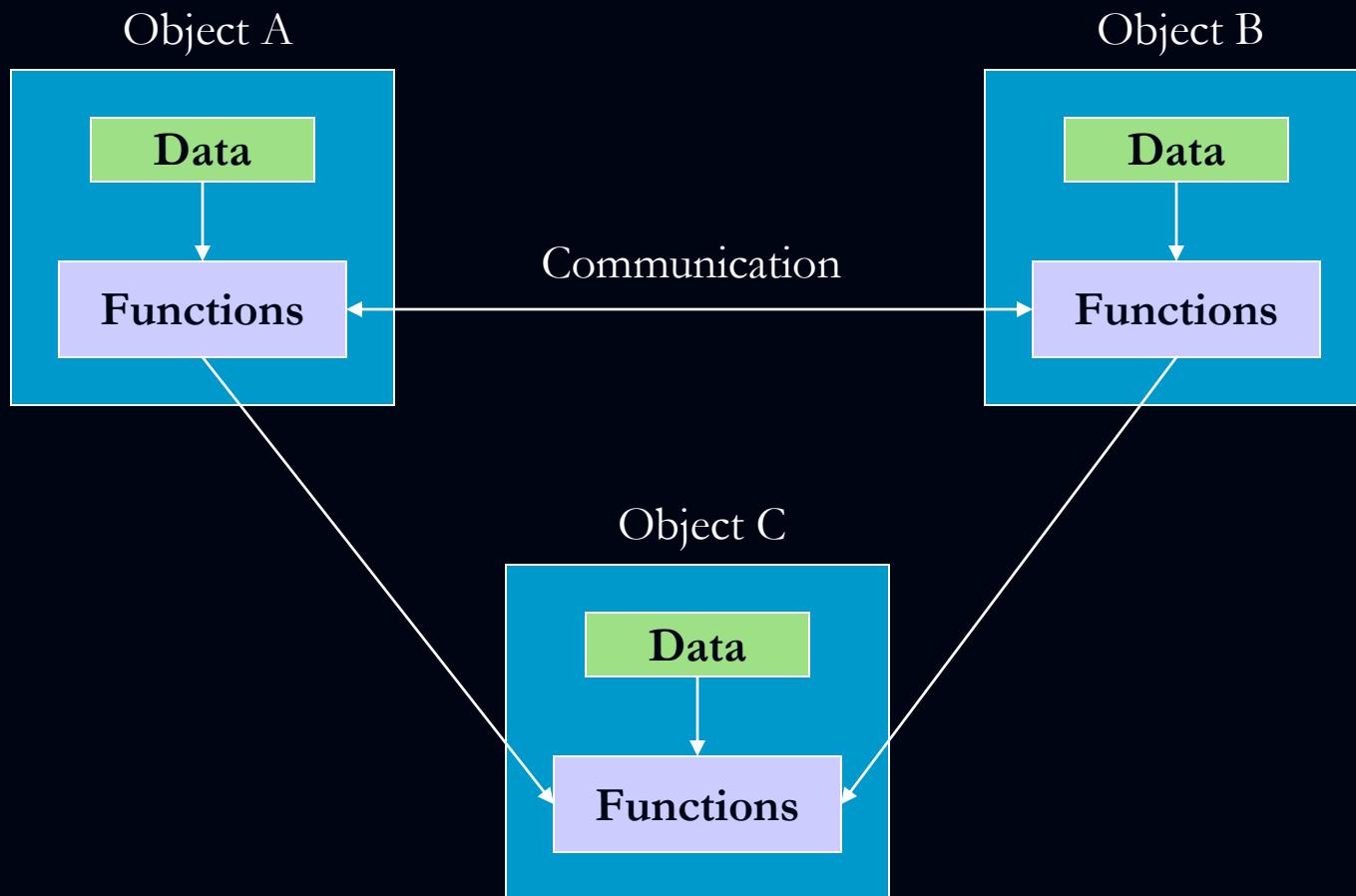
OOP is an approach to program organization and development that attempts to eliminate some of the pitfalls of conventional programming methods by incorporating the best of structured programming features with several new concepts.

Object-Oriented Programming

continue ...

- The data of an object can be accessed only by the functions associated with that object.
- Functions of one object can access the functions of another objects.

Organization of data and functions in OOP



Characteristics of Object-Oriented Programming

- Emphasis is on data rather than procedure.
- Programs are divided into objects.
- Data structures are designed such that they characterize the objects.
- Functions that operate on the data of an object are tied together in the data structure.
- Data is hidden and can not be accessed by external functions.

Characteristics of Object-Oriented Programming

continue ...

- Objects may communicate with each other through functions.
- New data and functions can be added easily whenever necessary.
- Follows bottom-up approach in program design.

Object-Oriented Programming

■ Definition:

It is an approach that provides a way of modularizing programs by creating partitioned memory area for both data and functions that can be used as templates for creating copies of such modules on demand. *Thus the object is considered to be a partitioned area of computer memory that stores data and set of operations that can access that data.*

Basic Concepts of Object-Oriented Programming

- Objects
- Classes
- Data Abstraction and Encapsulation
- Inheritance
- Polymorphism
- Dynamic Binding
- Message Passing

Basic Concepts of OOP

continue ...

■ Objects

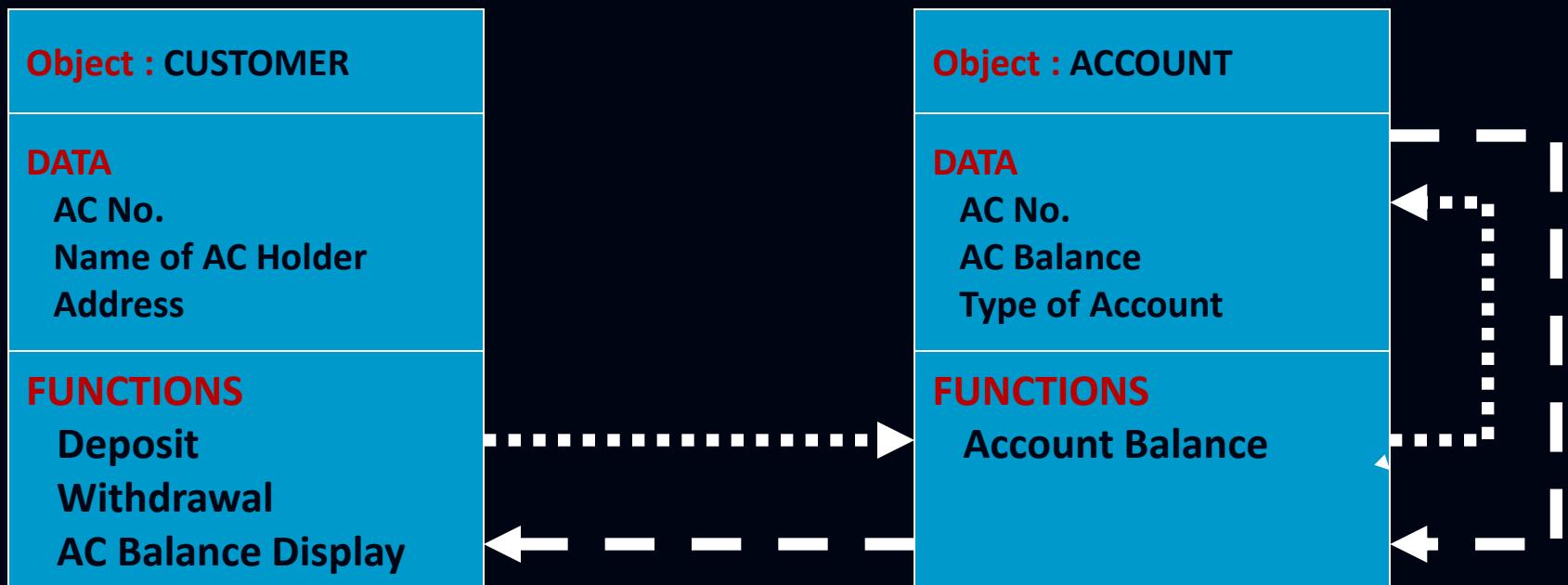
Objects are the basic run-time entities in an object-oriented system. They may represent a person, a place, a bank account, etc. Objects take up space in the memory and have an associated address like a structure in C.

When a program is executed, the objects interact by sending messages to one another.

Basic Concepts of OOP

continue ...

■ Objects



Basic Concepts of OOP

continue ...

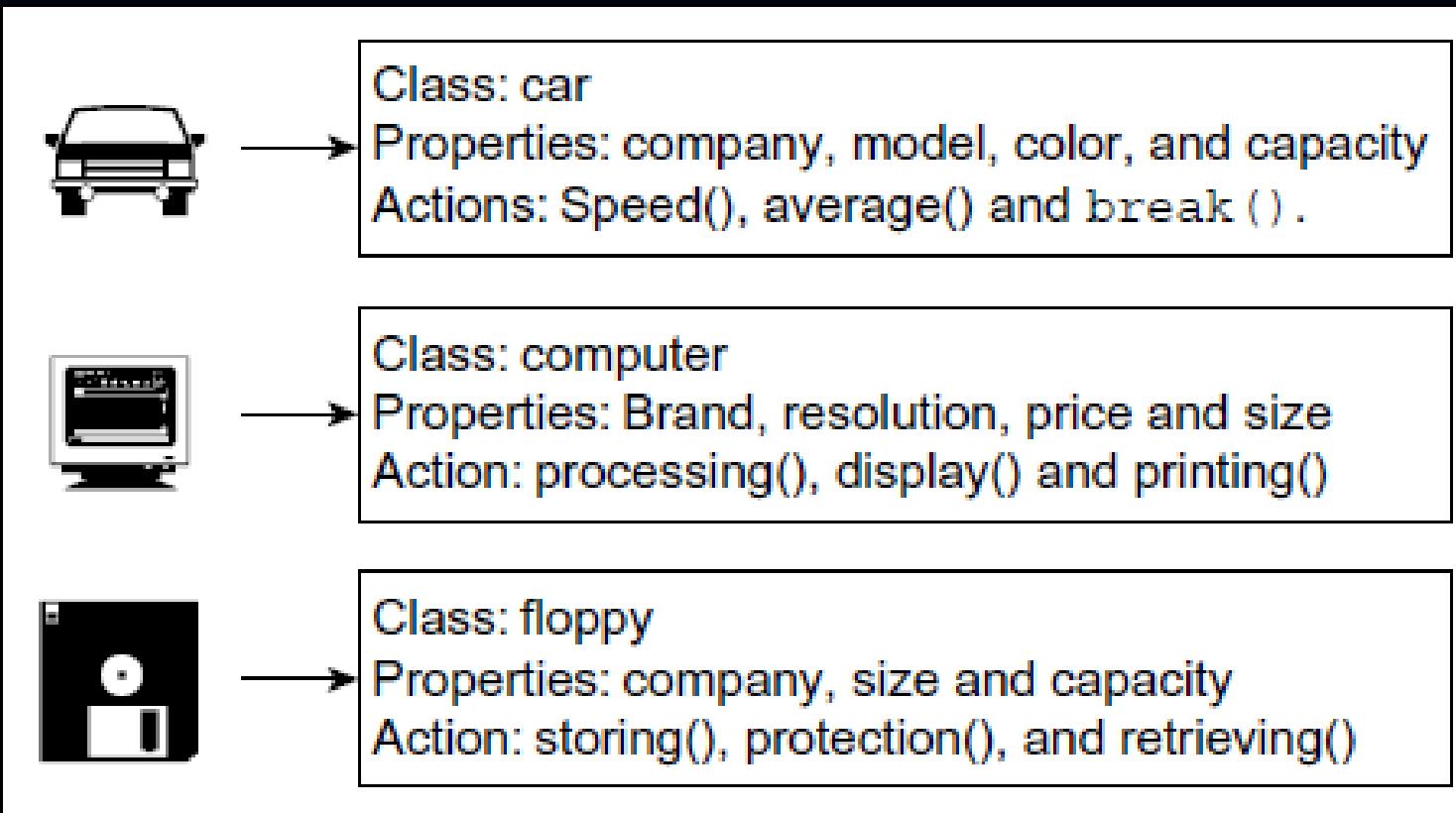
■ Classes

Classes are user-defined data types.

The entire set of data and code of an object can be made a user-defined data type with the help of a class. Objects are variables of the type class. Once a class has been defined, we can create any number of objects belonging to that class. Each object is associated with the data of type class with which they are created.

A class is a collection of objects of similar type.

A class is grouping of objects that have the identical properties, common behaviour, and shared relationship. A class binds the data and its related functions together.



Class contains

- data members
- member functions

it's used to implement encapsulation, data abstraction, and data hiding

Basic Concepts of OOP

continue ...

■ Classes

If fruit has been defined as a class, then the statement

```
fruit mango;
```

will create an object **mango** belonging to the class **fruit**.

Method

- An operation required for an object or entity when coded in a class is called a method.
- An operation required for an object or entity when coded in a class is called a method. The operations that are required for an object are to be defined in the class.
- All objects in a class carry out certain common actions or operations. Each action needs an object that becomes a function in the class that defines it and is referred to as a method.

- The class A contains private data members and public methods or member functions. Usually, the data members are declared private and methods or member functions are declared as public and they are available outside the class. The data member of any class uses its member functions or methods to perform operations.

Basic Concepts of OOP

continue ...

■ Data Abstraction and Encapsulation

- Abstraction refers to the procedure of representing essential features without including the background details.
- The wrapping up of data and functions into a single unit is known as encapsulation.
- The data is not accessible to the outside world, and only those functions which are wrapped in the class can access it.

Basic Concepts of OOP

continue ...

■ Data Abstraction and Encapsulation

These functions provide the interface between the object's data and the program. This insulation of the data from direct access by the program is called data hiding or information hiding.

The attributes wrapped in the classes are called data members and the functions that operate on these data are called methods or member functions.

Since the classes use the concept of data abstraction, they are known as Abstracted Data Types (ADT). It consists of a set of values and a set of operations.

Basic Concepts of OOP

continue ...

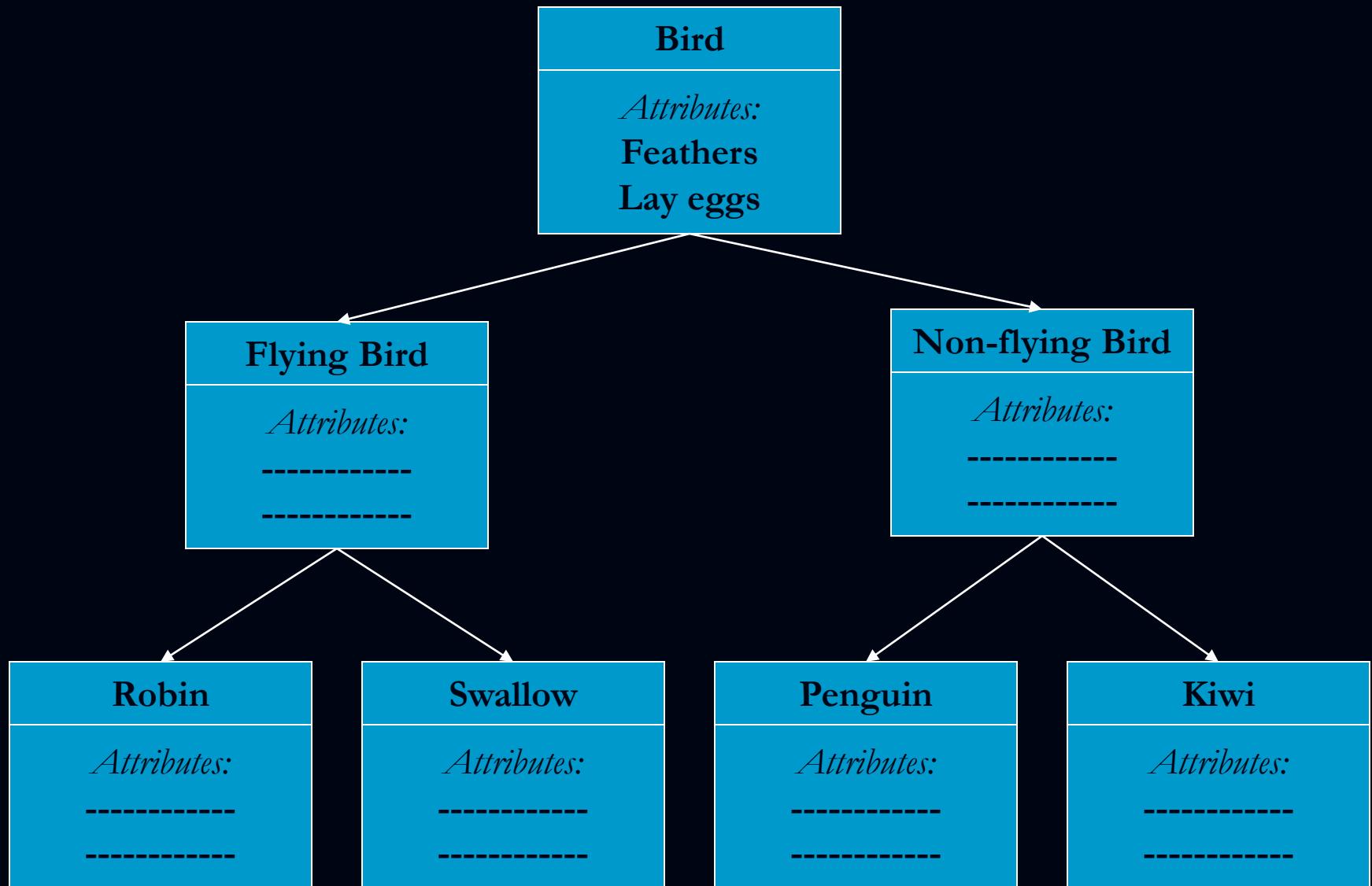
■ Inheritance

- Inheritance is the process by which objects of one class acquire the properties of objects of another class.

- It supports the concept of hierarchical classification.

- Each derived class shares common characteristics with the class from which it is derived.

Property Inheritance



Basic Concepts of OOP

continue ...

■ Inheritance

- Inheritance provides the idea of reusability.
- We can add additional features to an existing class without modifying it.

(By deriving new class from existing one. The new class will have the combined features of both the classes.)

Basic Concepts of OOP

continue ...

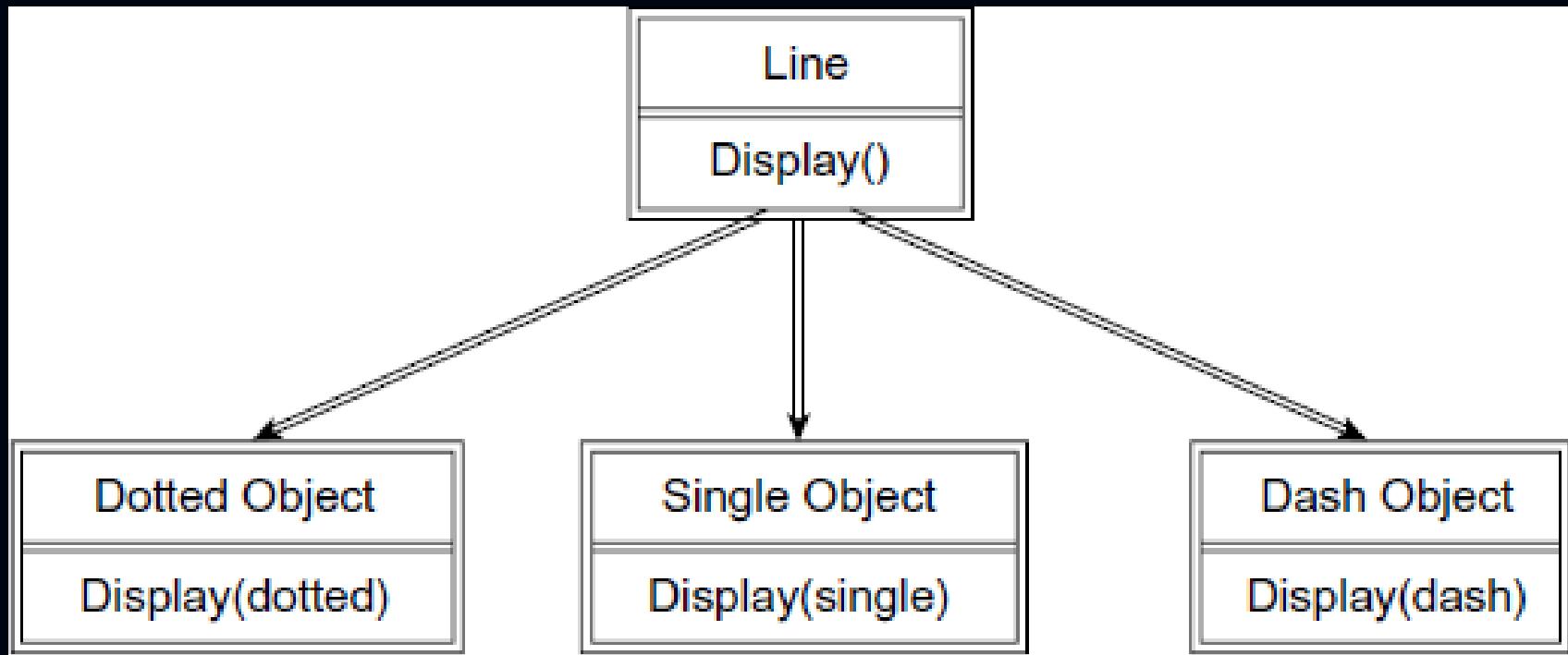
- Polymorphism - *ability to take more than one form*
 - An operation may exhibit different behaviours in different instances.
 - The behaviour depends upon the types of data used in the operation.
 - add(3, 5) gives 8
 - Add("hello", "-world") gives "hello-world"

Basic Concepts of OOP

continue ...

■ Polymorphism - *ability to take more than one form*

- The process of making an operator to exhibit different behaviours in different instances is known as operator overloading.
- << Insertion Operator
- << Left-shift bit-wise operator
- Using a single function name to perform different types of tasks is known as function overloading.
- add(3, 5) gives 8
- Add("hello", "-world") gives "hello-world"



Benefits of OOP

- Inheritance – eliminate redundant code and extend the use of existing classes.
- We can build programs from the standard working module, no need of starting from the scratch.
- Data hiding helps the programmer to build secure programs that can not be invaded by code in other parts of the program.

Benefits of OOP

continue ...

- Multiple instances of an objects can co-exists with out any interference.
- It is easy to partition the work in a project based on objects.
- Object-oriented system can be easily upgraded from small to large systems.
- Message passing techniques for communication between objects makes the interface descriptions with external systems much simpler.
- Software complexity can be easily managed.