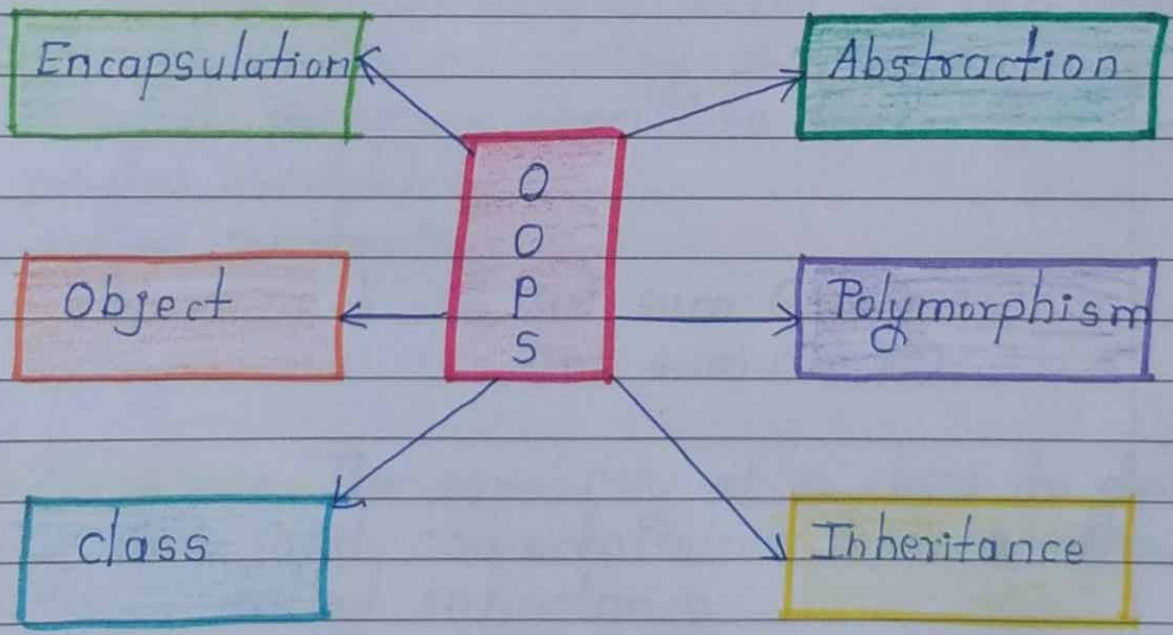
• OOPS in C++ •

• The main aim of OOP is to bind together the data and the functions the operate on them so that no other part of the code can access this data except this function.

krishna Agrawal <LinkedIn>
@coders_notes <instagram>

• Types of OOPS



• **Class :-** It is a user-defined data types, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class

• **Object :-** When a class is defined no memory is allocated but when it is instantiated. (i.e, object is created) memory is allocated

• **Encapsulation** :- In OOP, Encapsulation is defined as binding together the data and the functions that manipulates them.

• **Abstraction** - Abstraction means displaying only essential information and hiding the details
  • Abstraction using classes
  • Abstraction using Header files (math.h → pow())

• **Polymorphism** - In simple words, we can define polymorphism as an ability of a message to be displayed in more than one form
  • operator overloading
  • Function overloading
    → int sum (10, 20, 30)
      int sum (10, 20)

• **Inheritance** - The capability of a class to derive properties and characteristics from another class is called inheritance
  • subclass
  • superclass
  • reusability

• Dynamic Binding :

• In dynamic binding, the code to be executed in response to function call is decided at run time

• Constructors •

- A customer is a member function of a class which initializes objects of a class
- In c++ constructor is automatically called when the object creates
- It has same name as class itself
- Constructor don't have a return type

1. Default constructor (No parameter passed)
2. Parameterized constructor
3. copy constructor

• Destructor in c++ •

- Derived class destructor will be invoked first, then the base class destructor will be invoked

• Access Modifier •

- Public - can be accessed by any class
- Private - can be accessed only by a function in a class (inaccessible outside the class)
- Protected - It is also inaccessible outside the class but can be accessed by subclass at that class.

Note - If we do not specify any access modifier inside the class then by default the access modifier for the member will be private