



C++

Language

Notes

DEMO

For more PDFs and
notes visit my
telegram page..

Named "**BeingPro33**"

Handwritten Notes



@beingpro33



Object Oriented Programming System

→ Object oriented programming is a programming paradigm that revolves around the concept of object, which can contain data and functions to manipulate the data.

* Features of oops -

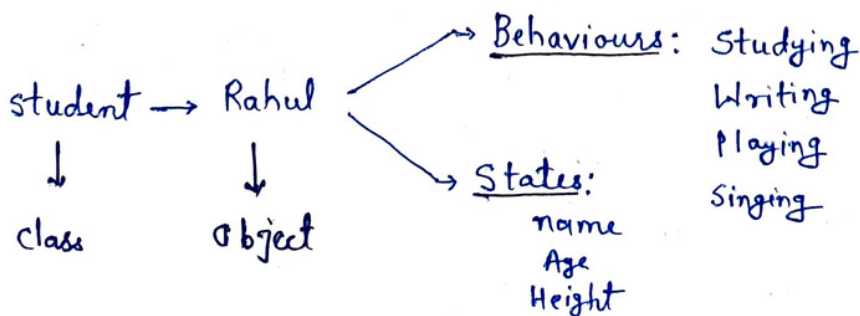
1. Inheritance
2. Abstraction
3. Encapsulation
4. Polymorphism

* Class -

A class is a blueprint or a template for creating objects that defines a set of variables, methods and properties that are common to all objects of that class.

* Object -

An object is an instance of a class or an entity which gets created using class and it represents the state and behaviour.



Inheritance

- * It is the process of acquiring features of an existing class into a new class.
- * The class that inherits properties is called the sub class or derived class or child class.
- * And the class that provides properties is called the super class or base class or parent class.

Eg:-

```
class Rectangle
{
    int length;
    int breadth;
    ---
    ---
};

class cuboid : Rectangle (Inherits the property of Rectangle class)
{
    int height;
    ---
    ---
};
```

Everything that is there in rectangle to be borrowed in cuboid class also.

Being Pro

Example -

```
class Base
{
    public:
        int x;
        void show ()
        {
            cout << x;
        }
};
```

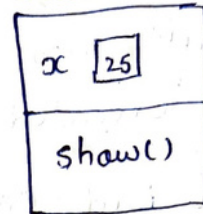
```
class Derived : public Base
{
    public:
        int y;
        void display ()
        {
            cout << x << y;
        }
};
```

```
int main ()
{
    Base b;
    b.x = 25;
    cout << b.show (); → 25
```

```
    Derived d;
    d.x = 10;
    d.y = 15;
```

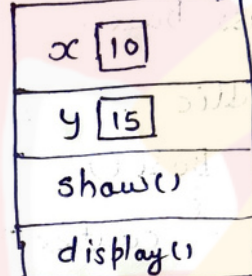
```
    cout << d.show (); → 10
    cout << d.display (); → 10, 15
```

Base Class



↑ inherits all features of base class.

Derived Class



* Constructor :

A constructor is a special member function that is automatically called when an object of a class is created.

- The purpose of a constructor is to initialize the data members of the object to specific values
- Every class will have a default constructor provided by the compiler.
- A constructor has the same name as the class name.
- It does not have any return type.
- It should be public.
- It can be declared as private also in some cases.
- Constructor is called when object is created.
- Constructor can be overloaded and it can take default arguments.

* Types of Constructor -

- i) Non-parameterised Constructor (Default Constructor)
- ii) Parameterised Constructor
- iii) Copy Constructor (Special member function of a class that creates a new object as a copy of an existing object)

Being Pro

Eg:-

```
#include <iostream>
using namespace std;

class MyClass
{
    public:
        int x;

        MyClass() // Default Constructor
        {
            x = 0;
            cout << "Default Constructor" << x << endl;
        }

        MyClass(int value) // Parameterised Constructor
        {
            x = value;
            cout << "Parameterized Constructor" << x << endl;
        }

        MyClass(MyClass &m) // Copy Constructor
        {
            x = m.x;
            cout << "Copy Constructor" << x << endl;
        }
};

int main()
{
    MyClass obj; // (It will call default constructor)
    MyClass obj1(5); // (It will call parameterised constr)
    MyClass obj2(obj1); // (It will call copy constructor)
    MyClass obj3 = obj1; // (copy constructor)
}
```