Object

Object

An object is an instance of a Class. It is an identifiable entity with some characteristics and behavior. To access the members defined inside the class, we need to create the object of that class. Objects are the basic units of object-oriented programming. It may be any real-world object like a person, chair, table, pen, animal, car, etc.

Code in object-oriented programming is organized around objects. Once you have your objects, they can interact with each other to make something happen.

Syntax to create an object in C++:

```
class_name objectName;
```

Syntax to create an object dynamically in C++:

```
class_name * objectName = new class_name();
```

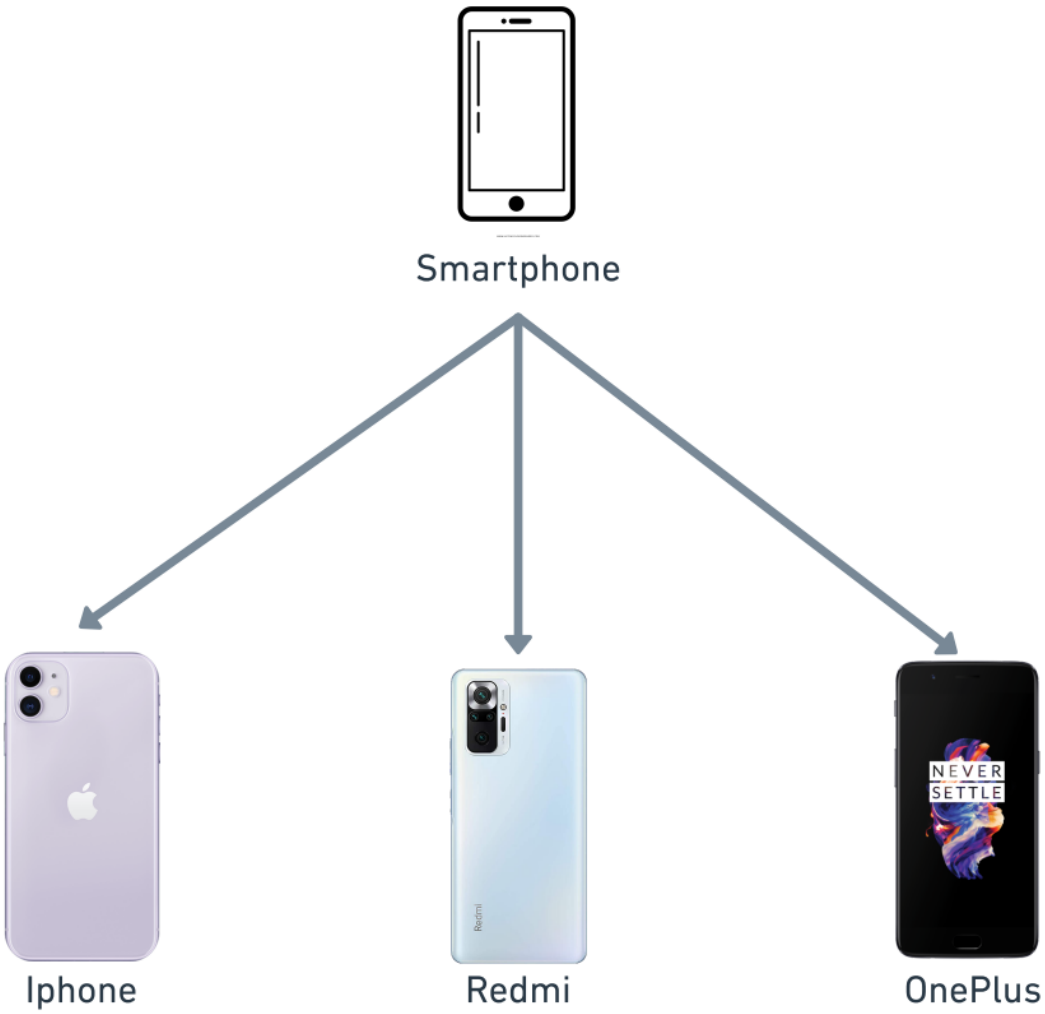
Here,

- objectName: It is the name of the object created by class_name.

The class's default constructor is called, and it dynamically allocates memory for one object of the class. The address of the memory allocated is assigned to the pointer, i.e., objectName.

Explaining object using Smartphone example with code:-

A simple example of an object would be a smartphone. Logically, you would expect a smartphone to have a model number or name. This would be considered the property of the smartphone. You could also expect a smartphone to do something, such as to send SMS, etc. This would be considered a method of the smartphone.



We have created a smartphone class earlier in the class module, and Now we will use that same class to make objects.

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

//creating class
class smartphone{
    //class body

        //Data Members(Properties)
        string model;
        int year_of_manufacture;
```

```
        bool _5g_supported;

        //Constructor
        smartphone(string model_string, int manufacture, bool _5g_){
            //initialzing data members
            model = model_string;
            year_of_manufacture = manufacture;
            _5g_supported = _5g_;
        }

        //methods
        void print_details(){
            cout << "Model : " << model << endl;
            cout << "Year of Manufacture : " << year_of_manufacture << endl;
            cout << "5g Supported : " << _5g_supported << endl;
        }
};


int main(){
        //creating objects of smartphone class
        smartphone iphone("iphone 11", 2019, false );
        smartphone redmi("redmi note 11 t", 2021, true );
        smartphone oneplus("oneplus nord", 2020, true );

        //accessing class variables
        int iphone_manufacture_date = iphone.year_of_manufacture;
        bool redmi_support_5g = redmi._5g_supported;
        string oneplus_model = oneplus.model;

        //calling methods on objects
        iphone.print_details();
        redmi.print_details();
        oneplus.print_details();
}
```

- ★ To create an object of a smartphone, specify the class name, followed by the object name.
- ★ To access the class attributes or data members (like model), use the dot syntax (.) on the object followed by the attribute name.
- ★ To call any method (print_details()) of class, use the dot syntax (.) on the object followed by the method name.

You need to have a class before you can create an object. When a class is defined, no memory is allocated, but memory is allocated when it is instantiated (i.e., an object is created).

Graphical Representation of smartphone class and its object:-