Types Of Inheritance In C#

Background

In this article we will learn about one of the most reusable object oriented features of C#, inheritance. We will learn about inheritance from the basics because I have written this article focusing on students and beginners. Before proceeding further please refer to my previous articles for a better understanding.

- Programming Methodologies
- Types of Classes in C#
- Polymorphism in C#

What Inheritance is

Acquiring (taking) the properties of one class into another class is called inheritance. Inheritance provides reusability by allowing us to extend an existing class.

The reason behind OOP programming is to promote the reusability of code and to reduce complexity in code and it is possible by using inheritance.

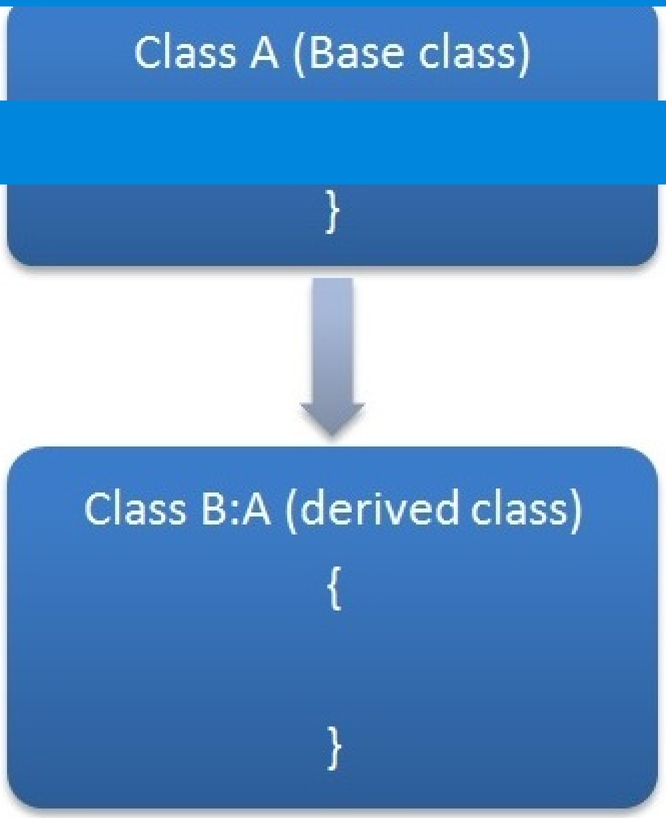
single
hierarchical
Multi level
Multiple(using interface)

The inheritance concept is based on a base class and derived class. Let us see the definition of a base and derived class.

- Base class - is the class from which features are to be inherited into another class.
- Derived class - it is the class in which the base class features are inherited.

Single inheritance

It is the type of inheritance in which there is one base class and one derived class.

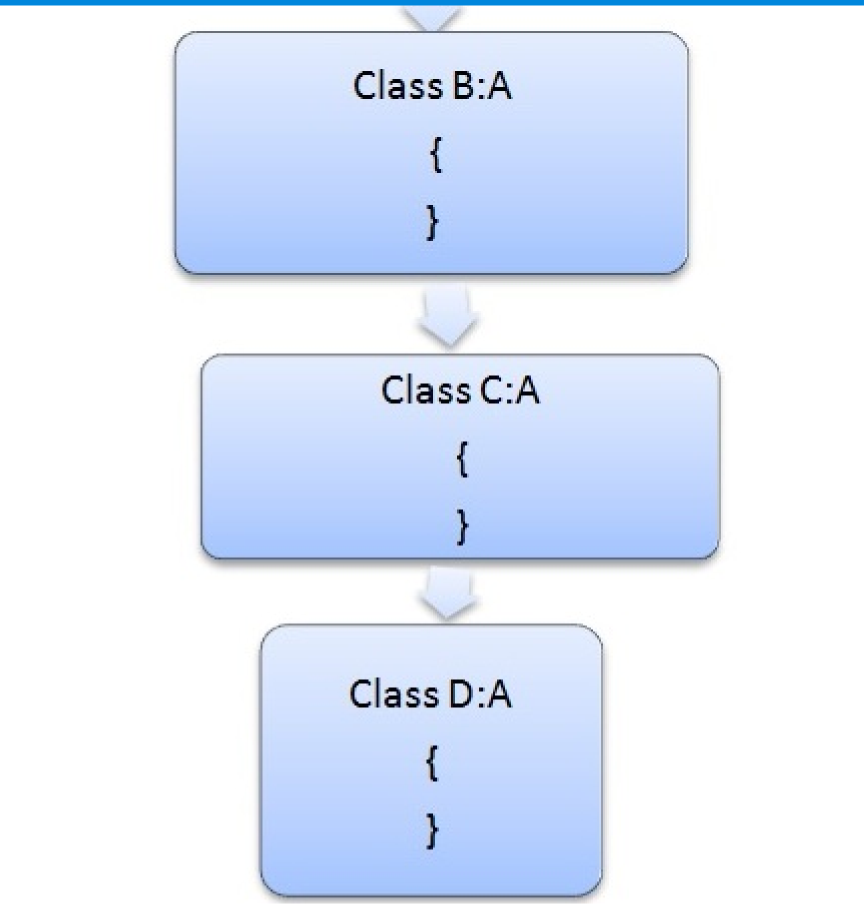


For example,

```
public class Accountcreditinfo //base class
{
    public string Credit()
    {
        return "balance is credited";
    }
}
public class debitinfo : Accountcreditinfo //derived class
{
    public string debit()
    {
        }
}
```

Hierarchical inheritance

This is the type of inheritance in which there are multiple classes derived from one base class. This type of inheritance is used when there is a requirement of one class feature that is needed in multiple classes.



For example,

```
class A  //base class
{
    public string msg()
    {
        return "this is A class Method";
```

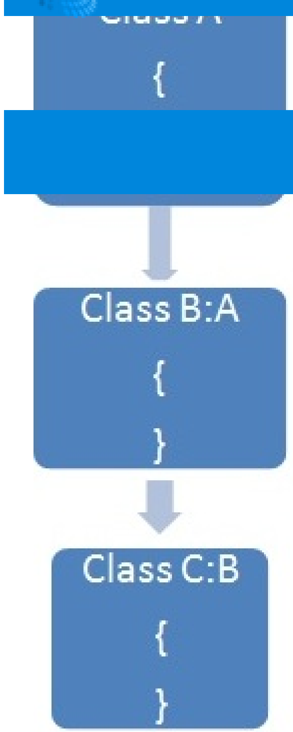
```
class B : A
{
    public string info()
```

```
        return "this is B class Method";
    }
    class C : A
    {
        public string getinfo()
        {
            msg();
            return "this is B class Method";
        }
    }
}
```

In the preceding program one base class is derived in many classes hence it is a called a Hierarchical Inheritance.

Multilevel inheritance

When one class is derived from another derived class then this type of inheritance is called multilevel inheritance.



For example,

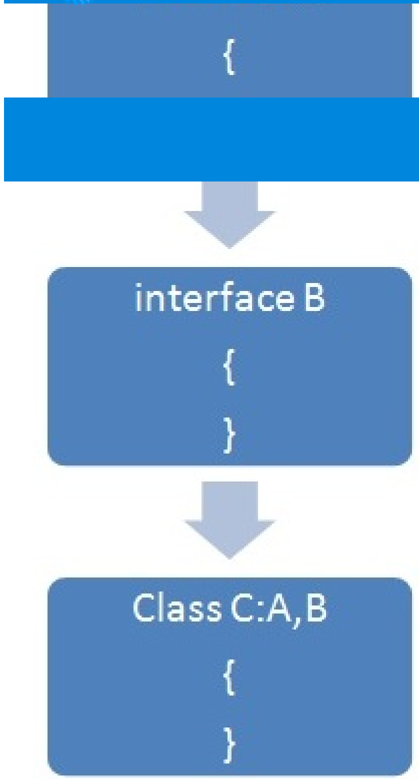
```
public class Person
{
    public string persondet()
    {
        return "this is the person class";
    }
}
public class Bird : Person
{
    public string birddet()
    {
        persondet();
        return "this is the birddet Class";
    }
```

```
{
    public string animaldet()
    {
        return "this is the Animal Class";
    }
}
```

In the preceding program, each class is derived from one class that is derived from another class hence this type of inheritance is called Multilevel Inheritance.

Multiple inheritance using Interfaces

C# does not support multiple inheritances of classes. To overcome this problem we can use interfaces, we will see more about interfaces in my next article in detail.



For example,

```
public interface IA //ineterface  1
{
    string setImgs(string a);
}
public interface IB  //Interface 2
{
    int getAmount(int Amt);
}
public class ICar : IA, IB //implementatin
{
    public int getAmount(int Amt)
    {
        return 100;
    }
```

```
        return "this is the car";
    }
}
```

Multiple Inheritance.

The following are some key points about inheritance:

1. C# does not support multiple inheritances of classes, the same thing can be done using interfaces.
2. Private members are not accessed in a derived class when one class is derived from another.

Summary

I hope you now have an overview of inheritance and its types. In the next article we will learn about each inheritance type in detail. If you have any suggestions regarding this article then please contact me.

Further recommended readings on C# inheritance

Here are more articles on inheritance and object oriented programming in C#.

Types of Inheritance in C#
In this article, we will learn about C# inheritance and types of inheritance in C# and .NET with code examples.

Overview of Inheritance in C#
In this article, you will learn about inheritance in C#. Inheritance is one of the major pillars of object-oriented programming. Here is an overview of inheritance.

Inheritance with Example in C#
Here are some example of inheritance using C#.

Learn all about Inheritance in C#
In this article, you will learn about Inheritance in C#.

This article explains the various behaviors of inheritance in various cases.

[Multiple Inheritance in C#](#)
Can you inherit from multiple classes in C#? Simply put, this cannot be done. However there are ways around it. From a design perspective you must ask yourself, will a Class fully represent an object?

[Multiple Inheritance in C# using Interfaces](#)
This is the simple mathematical operation program demonstrating how multiple inheritance can be achieved in C# using Interface Concept.

[Inheritance in C#](#)
Interface inheritance defines a new interface in terms of one or more existing interfaces. Implementation inheritance defines a new implementation in terms of one or more existing implementations.

Next Recommended Article
Simulating Multiple Inheritance in C#: Part I

C# Inheritance In C#

Vithal Wadje TOP 50

Vithal is a Developer, Software Architect, Microsoft MVP (since 2016) , C# Corner MVP (8 times), Microsoft C# TechNet Guru (March 2015), Author, Blogger, Speaker, C# Corner Mumbai User Group Lead and a Students' Ment... Read more

C# Corner ecome a member Login

Post ▼ Ask Question

👍 17 💬 5

Type your comment here and press Enter Key (Minimum 10 characters)

Nice article very helpful..thanks

Mohan Harne Sep 14, 2019

• 1650 • 267 • 0 👍 1 ↩ 0 Reply

Nicely explained. Please also add explanation for the reason why C# does not support Multiple inheritance with classes.

AKshay Raut Aug 12, 2018

• 1833 • 84 • 0 👍 3 ↩ 2 Reply

@Akshay - I guess you meant, " Why C# does not support multiple inheritance with classes". Here's the reason - It is because of the "deadly diamond problem". You can read more here. https://en.wikipedia.org/wiki/Multiple_inheritance

Mandar Hadke Aug 27, 2018

• 1905 • 12 • 0 👍 2

Mandar Hadke Yeah thanks. I just corrected it.

AKshay Raut Sep 08, 2018

• 1833 • 84 • 0 👍 1

This is very good But do small changes in that pls also display OUTPUT in that to

Arjun Walmiki Feb 18, 2018

• 1293 • 646 • 55.8k 👍 4 ↩ 0 Reply

C#Corner

ecome a member Login

Post ▼ Ask Question

FEATURED ARTICLES

CRUD Operations In PostgreSQL With EF Core And ASP.NET Core Web API

Views In Snowflake

PnP Date Time Picker Control In SharePoint Framework

TRENDING UP

01 Getting Started With Microsoft Fluent UI React

02 Build A FAQ Chatbot With Power Virtual Agents

03 What is Software-Defined Networking and Virtual Networks in Physical Networks

04 What are Virtualized Data Centers and VMware's SDDC Approach

05 What are Data Center Building Blocks and Network Virtualization Services?

06 Introduction to Network Virtualization

07 What is Docker?

ecome a member Login

Post ▼ Ask Question

09 Insert Data Into Azure Table Storage Using ASP.NET Core Application

About Us Contact Us Privacy Policy Terms Media Kit Sitemap Report a Bug FAQ Partners