C# Inheritance

In C#, inheritance is a process in which one object acquires all the properties and behaviors of its parent object automatically. In such way, you can reuse, extend or modify the attributes and behaviors which is defined in other class.

In C#, the class which inherits the members of another class is called **derived class** and the class whose members are inherited is called **base** class. The derived class is the specialized class for the base class.

Advantage of C# Inheritance

Code reusability: Now you can reuse the members of your parent class. So, there is no need to define the member again. So less code is required in the class.

⇧

C# Single Level Inheritance Example: Inheriting Fields

When one class inherits another class, it is known as single level inheritance. Let's see the example of single level inheritance which inherits the fields only.

```
using System;
   public class Employee
    {
       public float salary = 40000;
   }
   public class Programmer: Employee
   {
       public float bonus = 10000;
   }
   class TestInheritance{
       public static void Main(string[] args)
        {
            Programmer p1 = new Programmer();

            Console.WriteLine("Salary: " + p1.salary);
            Console.WriteLine("Bonus: " + p1.bonus);

        }
    }
```

Output:

```
Salary: 40000
Bonus: 10000
```

In the above example, Employee is the **base** class and Programmer is the **derived** class.

C# Single Level Inheritance Example: Inheriting Methods

Let's see another example of inheritance in C# which inherits methods only.

```
using System;
  public class Animal
   {
     public void eat() { Console.WriteLine("Eating..."); }
  }
  public class Dog: Animal
  {
     public void bark() { Console.WriteLine("Barking..."); }
  }
  class TestInheritance2{
     public static void Main(string[] args)
```

```
       {
          Dog d1 = new Dog();
          d1.eat();
          d1.bark();
       }
    }
```

Output:

```
Eating...
Barking...
```

C# Multi Level Inheritance Example

When one class inherits another class which is further inherited by another class, it is known as multi level inheritance in C#. Inheritance is transitive so the last derived class acquires all the members of all its base classes.

Let's see the example of multi level inheritance in C#.

```
using System;
   public class Animal
    {
       public void eat() { Console.WriteLine("Eating..."); }
   }
   public class Dog: Animal
   {
       public void bark() { Console.WriteLine("Barking..."); }
   }
   public class BabyDog : Dog
```

```
{
    public void weep() { Console.WriteLine("Weeping..."); }
}
class TestInheritance2{
    public static void Main(string[] args)
     {
         BabyDog d1 = new BabyDog();
         d1.eat();
         d1.bark();
         d1.weep();
     }
 }
```

Output:

```
Eating...
Barking...
Weeping...
```

> Note: Multiple inheritance is not supported in C# through class.

← prev next →

Help Others, Please Share

Learn Latest Tutorials

Testing	Proc*C	SMM	GDB
Fuzzy Logic	DHTML	Classroom	AutoCad
Kubernetes	Openpyxl	Tally	Godot

⇧

Preparation

Aptitude

Reasoning

Verbal A.

Interview

Company

Trending Technologies

AI

AWS

Selenium

Cloud

Hadoop

ReactJS

D. Science

Angular 7

Blockchain

GIT

Machine Learning

DevOps

Blockchain

Git

ML

DevOps

B.Tech / MCA

DBMS

DBMS

Data Structure

DS

DAA

OS

C. Network

Compiler D.

COA

Discrete Mathematics Tutorial

D. Math.

E. Hacking

Computer Graphics

C. Graphics

Software Engineering

Software E.

HTML

Web Tech.

Cyber Security tutorial

Cyber Sec.

Automata Tutorial

Automata

C Language tutorial

C

C++ tutorial

C++

Java tutorial

Java

.Net Framework tutorial

Python tutorial

Python

List of Programs

Programs

⇧

.Net

Control Systems tutorial

Control S.

Data Mining Tutorial

Data Mining

⇧