

base keyword in C#

The base keyword is used to access members of the base class from within a derived class.

We can use base keyword for following things as

- Access characteristics of base class
- Call a method on the base class that has been overridden by another method.
- Specify which base-class constructor should be called when creating instances of the derived class.

A base class access is permitted only in a constructor, an instance method, or an instance property accessor.

It is an error to use the base keyword from within a static method.

Application :

Application to demonstrate all scenarios in which we can use base keyword.

using System;

// base keyword is used to access characteristics of base class

```
public class base1
```

```
{  
    public string name = "Marvellous";  
}
```

```
public class Derived1 : base1
```

```
{  
    public string name = "Infosystems";  
  
    public void Display()  
    {  
        Console.WriteLine(base.name);  
        Console.WriteLine(name);  
    }  
}
```

```
}
```

// base keyword is used to access behaviour of base class

```
public class base2
```

```
{
```

```
    public void fun()
```

```
    {
```

```
        Console.WriteLine("Inside base fun");
```

```
    }
```

```
}
```

```
public class Derived2 : base2
```

```
{
```

```
    public void fun()
```

```
    {
```

```
        Console.WriteLine("Inside derived fun");
```

```
        base.fun();
```

```
    }
```

```
}
```

// base keyword is used to call base class constructor explicitly.

```
public class base3
```

```
{
```

```
    public base3(int i)
```

```
    {
```

```
        Console.WriteLine("Base class constructor");
```

```
    }
```

```
}
```

```
public class Derived3 : base3
```

```
{
```

```
    public Derived3(int x):base(x)
```

```
    {
```

```
        Console.WriteLine("Derived class constructor");
```

```
    }
```

```
}
```

```
class Marvellous
```

```
{
```

```
    static void Main(string[] args)
```

```
    {
```

```
        Derived1 dobj1 = new Derived1();
```

```
        dobj1.Display();
```

```
        Derived2 dobj2 = new Derived2();
```

```
        dobj2.fun();
```

```
        Derived3 doj3 = new Derived3(11);
```

```
    }
```

```
}
```

