



Hire with us!

## Difference between Class and Structure in C#

A **class** is a user-defined blueprint or prototype from which objects are created. Basically, a class combines the fields and methods(member function which defines actions) into a single unit.

### Example:

```
// C# program to illustrate the
// concept of class
using System;

// Class Declaration
public class Author {

    // Data members of class
    public string name;
    public string language;
    public int article_no;
    public int improv_no;

    // Method of class
    public void Details(string name, string language,
                       int article_no, int improv_no)
    {
        this.name = name;
        this.language = language;
        this.article_no = article_no;
        this.improv_no = improv_no;

        Console.WriteLine("The name of the author is : " + name
                          + "\nThe name of language is : " + language
                          + "\nTotal number of article published "
                          + article_no + "\nTotal number of Improvements:"
                          + " done by author is : " + improv_no);
    }

    // Main Method
    public static void Main(String[] args)
    {

        // Creating object
        Author obj = new Author();

        // Calling method of class
        // using class object
        obj.Details("Ankita", "C#", 80, 50);
    }
}
```

```
}  
}
```

### Output:

```
The name of the author is : Ankita  
The name of language is : C#  
Total number of article published 80  
Total number of Improvements: done by author is : 50
```

A **structure** is a collection of variables of different data types under a single unit. It is almost similar to a class because both are user-defined data types and both hold a bunch of different data types.

### Example:

```
// C# program to illustrate the  
// concept of structure  
using System;  
  
// Defining structure  
public struct Car  
{  
  
    // Declaring different data types  
    public string Brand;  
    public string Model;  
    public string Color;  
}  
  
class GFG {  
  
    // Main Method  
    static void Main(string[] args)  
    {  
  
        // Declare c1 of type Car  
        // no need to create an
```

```

// instance using 'new' keyword
Car c1;

// c1's data
c1.Brand = "Bugatti";
c1.Model = "Bugatti Veyron EB 16.4";
c1.Color = "Gray";

// Displaying the values
Console.WriteLine("Name of brand: " + c1.Brand +
                  "\nModel name: " + c1.Model +
                  "\nColor of car: " + c1.Color);
}
}

```

### Output:

```

Name of brand: Bugatti
Model name: Bugatti Veyron EB 16.4
Color of car: Gray

```

### Difference between Class and Structure

CLASS	STRUCTURE
Classes are of reference types.	Structs are of value types.
All the reference types are allocated on heap memory.	All the value types are allocated on stack memory.
Allocation of large reference type is cheaper than allocation of large value type.	Allocation and de-allocation is cheaper in value type as compare to reference type.
Class has limitless features.	Struct has limited features.
Class is generally used in large programs.	Struct are used in small programs.
Classes can contain constructor or destructor.	Structure does not contain parameter less constructor or destructor, but can contain Parameterized constructor or static constructor.
Classes used new keyword for creating instances.	Struct can create an instance, with or without new keyword.
A Class can inherit from another class.	A Struct is not allowed to inherit from another struct or

	class.
The data member of a class can be protected.	The data member of struct can't be protected.
Function member of the class can be virtual or abstract.	Function member of the struct cannot be virtual or abstract.
Two variable of class can contain the reference of the same object and any operation on one variable can affect another variable.	Each variable in struct contains its own copy of data(except in ref and out parameter variable) and any operation on one variable can not effect another variable.

## Recommended Posts:

[Top 10 Programming Languages for Blockchain Development](#)

[Basic CRUD \(Create, Read, Update, Delete\) in ASP.NET MVC Using C# and Entity Framework](#)

[Program to swap numbers using XOR operator in C#](#)

[Uri.HexUnescape\(\) Method in C# with Examples](#)

[Uri.Fragment Property in C# with Examples](#)

[Uri.DnsSafeHost Property in C# with Examples](#)

[Uri.IsLoopback Property in C# with Example](#)

[File.Replace\(String, String, String\) Method in C# with Examples](#)

[File.WriteAllText\(String, String, Encoding\) Method in C# with Examples](#)

[File.WriteAllText\(String, String\) Method in C# with Examples](#)

[File.ReadAllText\(String, Encoding\) Method in C# with Examples](#)

[File.ReadAllText\(String\) Method in C# with Examples](#)

[File.ReadLines\(String, Encoding\) Method in C# with Examples](#)

[File.ReadLines\(String\) Method in C# with Examples](#)

[File.ReadAllLines\(String, Encoding\) Method in C# with Examples](#)



**ankita\_saini**

Check out this Author's [contributed articles](#).

If you like GeeksforGeeks and would like to contribute, you can also write an article using [contribute.geeksforgeeks.org](https://contribute.geeksforgeeks.org) or mail your article to [contribute@geeksforgeeks.org](mailto:contribute@geeksforgeeks.org). See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please Improve this article if you find anything incorrect by clicking on the "Improve Article" button below.

**Improved By :** [zeuss](#)

**Article Tags :** [C#](#) [CSharp-OOP](#)



2

**1.5**

☐ To-do ☐ Done

Based on 2 vote(s)

[Feedback/ Suggest Improvement](#)

[Improve Article](#)

Please write to us at [contribute@geeksforgeeks.org](mailto:contribute@geeksforgeeks.org) to report any issue with the above content.

[Previous](#)

[C# | Get an enumerator that iterates through the ListDictionary](#)

[Next](#)

[C# | Get an enumerator that iterates through the SortedSet](#)

[>](#)

Writing code in comment? Please use [ide.geeksforgeeks.org](https://ide.geeksforgeeks.org), generate link and share the link here.

Load Comments

GeeksforGeeks

NOT  
~~AN AD~~

JUST A  
REMINDER



THAT  
YOU NEED TO  
**LEARN DSA**

**Learn Now**

## Most popular in C#

[File.WriteAllText\(String, String\) Method in C# with Examples](#)

[File.ReadAllBytes\(\) Method in C# with Examples](#)

[Uri.Fragment Property in C# with Examples](#)

[How to Pad an Integer Number With Leading Zeroes in C#?](#)

[Program to swap numbers using XOR operator in C#](#)



## More related articles in C#

[sizeof\(\) Operator in C#](#)

[File.WriteAllText\(String, String, Encoding\) Method in C# with Examples](#)

[Different Methods to Read a Character in C#](#)

[Difference between Int32 and UInt32 in C#](#)

[Difference between UInt16, UInt32 and UInt64 in C#](#)

# GeeksforGeeks

A computer science portal for geeks

5th Floor, A-118,  
Sector-136, Noida, Uttar Pradesh - 201305  
[feedback@geeksforgeeks.org](mailto:feedback@geeksforgeeks.org)

## COMPANY

[About Us](#)  
[Careers](#)  
[Privacy Policy](#)  
[Contact Us](#)

## PRACTICE

[Courses](#)  
[Company-wise](#)  
[Topic-wise](#)  
[How to begin?](#)

## LEARN

[Algorithms](#)  
[Data Structures](#)  
[Languages](#)  
[CS Subjects](#)  
[Video Tutorials](#)

## CONTRIBUTE

[Write an Article](#)  
[Write Interview Experience](#)  
[Internships](#)  
[Videos](#)



@geeksforgeeks, Some rights reserved