

# C# Identifiers

In programming languages, identifiers are used for identification purposes. Or in other words, identifiers are the user-defined name of the program components. In C#, an identifier can be a class name, method name, variable name or label.

## Example:

```
public class GFG {  
    static public void Main ()  
    {  
        int x;  
    }  
}
```

Here the total number of identifiers present in the above example is 3 and the names of these identifiers are:

- **GFG:** Name of the class
- **Main:** Method name
- **x:** Variable name

## Rules for defining identifiers in C#:

There are certain valid rules for defining a valid C# identifier. These rules should be followed, otherwise, we will get a compile-time error.

- The only allowed characters for identifiers are all alphanumeric characters([A-Z], [a-z], [0-9]), ‘\_’ (underscore). For example “geek@” is not a valid C# identifier as it contain ‘@’ – special character.
- Identifiers should not start with digits([0-9]). For example “123geeks” is a not a valid in C# identifier.
- Identifiers should not contain white spaces.
- Identifiers are not allowed to use as [keyword](#) unless they include @ as a prefix. For example, @as is a valid identifier, but “as” is not because it is a keyword.
- C# identifiers allow Unicode Characters.
- C# identifiers are case-sensitive.
- C# identifiers cannot contain more than 512 characters.
- Identifiers does not contain two consecutive underscores in its name because such types of identifiers are used for the implementation.

## Example:

- CSharp

// Simple C# program to illustrate identifiers

**using** System;

**class** GFG {

    // Main Method

**static public void** Main()

    {

        // variable

**int** a = 10;

**int** b = 39;

**int** c;

        // simple addition

        c = a + b;

```
        Console.WriteLine("The sum of two number is: {0}", c);  
    }  
}
```

### **Output:**

The sum of two number is: 49

Below table shows the identifiers and keywords present in the above example:

Keywords	Identifiers
----------	-------------

using	GFG
-------	-----

public	Main
--------	------

static	a
--------	---

void	b
------	---

int	c
-----	---

