

C# Strings

In C#, string is an object of **System.String** class that represent sequence of characters. We can perform many operations on strings such as concatenation, comparison, getting substring, search, trim, replacement etc.

string vs String

In C#, *string* is keyword which is an alias for *System.String* class. That is why string and String are equivalent. We are free to use any naming convention.

```
string s1 = "hello";//creating string using string keyword
String s2 = "welcome";//creating string using String class
```

C# String Example

```
using System;
public class StringExample
{
    public static void Main(string[] args)
    {
        string s1 = "hello";

        char[] ch = { 'c', 's', 'h', 'a', 'r', 'p' };
        string s2 = new string(ch);

        Console.WriteLine(s1);
        Console.WriteLine(s2);
    }
}
```

Output:

```
hello
csharp
```

C# String methods

Method Name	Description
<code>Clone()</code>	It is used to return a reference to this instance of String.
<code>Compare(String, String)</code>	It is used to compares two specified String objects. It returns an integer that indicates their relative position in the sort order.
<code>CompareOrdinal(String, String)</code>	It is used to compare two specified String objects by evaluating the numeric values of the corresponding Char objects in each string..
<code>CompareTo(String)</code>	It is used to compare this instance with a specified String object. It indicates whether this instance precedes, follows, or appears in the same position in the sort order as the specified string.
<code>Concat(String, String)</code>	It is used to concatenate two specified instances of String.
<code>Contains(String)</code>	It is used to return a value indicating whether a specified substring occurs within this string.
<code>Copy(String)</code>	It is used to create a new instance of String with the same value as a specified String.
<code>CopyTo(Int32, Char[], Int32, Int32)</code>	It is used to copy a specified number of characters from a specified position in this instance to a specified position in an array of Unicode characters.

EndsWith(String)	It is used to check that the end of this string instance matches the specified string.
Equals(String, String)	It is used to determine that two specified String objects have the same value.
Format(String, Object)	It is used to replace one or more format items in a specified string with the string representation of a specified object.
GetEnumerator()	It is used to retrieve an object that can iterate through the individual characters in this string.
GetHashCode()	It returns the hash code for this string.
GetType()	It is used to get the Type of the current instance.
GetTypeCode()	It is used to return the TypeCode for class String.
IndexOf(String)	It is used to report the zero-based index of the first occurrence of the specified string in this instance.
Insert(Int32, String)	It is used to return a new string in which a specified string is inserted at a specified index position.
Intern(String)	It is used to retrieve the system's reference to the specified String.
IsInterned(String)	It is used to retrieve a reference to a specified String.
IsNormalized()	It is used to indicate that this string is in Unicode normalization form C.
IsNullOrEmpty(String)	It is used to indicate that the specified string is null or an Empty string.
IsNullOrWhiteSpace(String)	It is used to indicate whether a specified string is null , empty, or consists only of white-space characters.
Join(String, String[])	It is used to concatenate all the elements of a string array, using the specified separator between each element.
LastIndexOf(Char)	It is used to report the zero-based index position of the last occurrence of a specified character within String.

LastIndexOfAny(Char[])	It is used to report the zero-based index position of the last occurrence in this instance of one or more characters specified in a Unicode array.
Normalize()	It is used to return a new string whose textual value is the same as this string, but whose binary representation is in Unicode normalization form C.
PadLeft(Int32)	It is used to return a new string that right-aligns the characters in this instance by padding them with spaces on the left.
PadRight(Int32)	It is used to return a new string that left-aligns the characters in this string by padding them with spaces on the right.
Remove(Int32)	It is used to return a new string in which all the characters in the current instance, beginning at a specified position and continuing through the last position, have been deleted.
Replace(String, String)	It is used to return a new string in which all occurrences of a specified string in the current instance are replaced with another specified string.
Split(Char[])	It is used to split a string into substrings that are based on the characters in an array.
StartsWith(String)	It is used to check whether the beginning of this string instance matches the specified string.
Substring(Int32)	It is used to retrieve a substring from this instance. The substring starts at a specified character position and continues to the end of the string.
ToCharArray()	It is used to copy the characters in this instance to a Unicode character array.
ToLower()	It is used to convert String into lowercase.
ToLowerInvariant()	It is used to return convert String into lowercase using the casing rules of the invariant culture.
ToString()	It is used to return instance of String.
ToUpper()	It is used to convert String into uppercase.

Trim()	It is used to remove all leading and trailing white-space characters from the current String object.
TrimEnd(Char[])	It Is used to remove all trailing occurrences of a set of characters specified in an array from the current String object.
TrimStart(Char[])	It is used to remove all leading occurrences of a set of characters specified in an array from the current String object.