**Date And Time Format Specifiers In C#**

A date and time format specifier is a special character that enables you to display the date and time values in different formats.

**Example**

* You can display a date in mm-dd-yyyy format and time in hh:mm format.
* If you are displaying GMT time as the output, you can display the GMT time along with the abbreviation GMT using date and time format specifiers.
* The date and time format specifiers allow you to display the date and time in 12-hour and 24-hour formats.
* The following is the syntax for date and time format specifiers:

Console.WriteLine(“{format specifier}”, <datetime object>);

**Where,**

* formatspecifier: Is the date and time format specifier.
* datetime object: Is the object of the DateTime class.

|  |  |
| --- | --- |
| **Format Specifier** | **Name** |
| d | Short date |
| D | Long date |
| f | Full date/time (short time) |
| F | Full date/time (long time) |
| g | •General date/time (short time) |

**Syntax**

DateTime dt = DateTime.Now;  
Console.WriteLine("{0:D}", dt);  
Console.WriteLine("{0:f}", dt);  
Console.WriteLine("{0:F}", dt);  
Console.WriteLine("{0:g}", dt);  
Console.WriteLine("{0:d} {1:D}", dt, dt);

**Time Formats**

|  |  |
| --- | --- |
| **Format Specifier** | **Name** |
| G | General date/time (long time) |
| m or M | Month day |
| t | Short time |
| T | Long time |
| y or Y | Year month pattern |
| ddd | Represents the abbreviated name of the day of the week. |
| dddd | Represents the full name of the day of the week. |
| FF | Represents the two digits of the seconds fraction |
| HH | Represents the hour from 00 to 23 |
| MM | Represents the month as a number from 01 to 12 |
| MMM | Represents the abbreviated name of the month |
| ss | •Represents the seconds as a number from 0 to 59 |
|  |  |

**Full Date & Time Example**

Console.WriteLine("{0:HH:mm:ss tt}", dt);  
Console.WriteLine("{0:dd-MM-yyyy}", dt);

## Using Standard Date and Time Format Specifiers

The following code demonstrates the conversion of a specified date and time using the d, D, f, F, and g date and time format specifiers:

DateTimedt = DateTime.Now;

// Returns short date (MM/DD/YYYY)

Console.WriteLine(“Short date format(d): {0:d}”, dt);

// Returns long date (Day, Month Date, Year)

Console.WriteLine(“Long date format (D): {0:D}”, dt);

// Returns full date with time without seconds

Console.WriteLine(“Full date with time without seconds (f):{0:f}”, dt);

// Returns full date with time with seconds

Console.WriteLine(“Full date with time with seconds (F):{0:F}”, dt);

// Returns short date and short time without seconds

Console.WriteLine(“Short date and short time without seconds (g):{0:g}”, dt);

## Output

Short date format(d): 23/04/2007  
Long date format (D): Monday, April 23, 2007  
Full date with time without seconds (f):Monday, April 23, 2007 12:58 PM  
Full date with time with seconds (F):Monday, April 23, 2007 12:58:43 PM  
Short date and short time without seconds (g):23/04/2007 12:58 PM

The following code demonstrates the conversion of a specified date and time using the G, m, t, T, and y date and time format specifiers:

DateTimedt = DateTime.Now;

// Returns short date and short time with seconds

Console.WriteLine(“Short date and short time with seconds (G):{0:G}”, dt);

// Returns month and day - M can also be used

Console.WriteLine(“Month and day (m):{0:m}”, dt);

// Returns short time

Console.WriteLine(“Short time (t):{0:t}”, dt);

// Returns short time with seconds

Console.WriteLine(“Short time with seconds (T):{0:T}”, dt);

// Returns year and month - Y also can be used

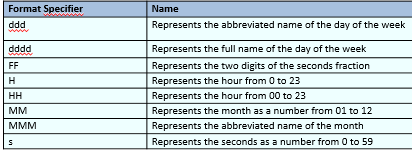
Console.WriteLine(“Year and Month (y):{0:y}”, dt);

## Output

Short date and short time with seconds (G):23/04/2007 12:58:43 PM  
Month and day (m):April 23  
Short time (t):12:58 PM  
Short time with seconds (T):12:58:43 PM  
Year and Month (y):April, 2007

**Custom DateTime Format Strings**

Any non-standard DateTime format string is referred to as a custom DateTime format string. Custom DateTime format strings consist of more than one custom DateTime format specifiers. The following table lists some of the custom DateTime format specifiers:



**The following code demonstrates the use of custom DateTime format specifiers:**

using System;

class DateTimeFormat

{

public static void Main(string[] args)

{

DateTime date = DateTime.Now;

Console.WriteLine("Date is {0:ddd MMM dd, yyyy}", date);

Console.WriteLine("Time is {0:hh:mm tt}", date);

Console.WriteLine("24 hour time is {0:HH:mm}", date);

Console.WriteLine("Time with seconds: {0:HH:mm:ss tt}", date);

Console.WriteLine("Day of month: {0:m}", date);

Console.WriteLine("Year: {0:yyyy}", date);

}

}

**In Above Code,**

* The date and time is displayed using the different DateTime format specifiers.

**The following figure displays the output of the code:**

