

# 1 System.Globalization.DateTimeFormatInfo

## 2 Class

```
3 [ILAsm]  
4 .class public sealed serializable DateTimeFormatInfo extends System.Object  
5 implements System.ICloneable, System.IFormatProvider  
  
6 [C#]  
7 public sealed class DateTimeFormatInfo: ICloneable, IFormatProvider
```

### 8 Assembly Info:

- 9 • *Name:* mscorlib
- 10 • *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00 00]
- 11 • *Version:* 2.0.x.x
- 12 • *Attributes:*
  - 13 ○ CLSCompliantAttribute(true)

### 14 Implements:

- 15 • **System.ICloneable**
- 16 • **System.IFormatProvider**

### 17 Summary

18 Defines culture-specific formats and patterns for System.DateTime values.

### 19 Inherits From: System.Object

20  
21 **Library:** BCL

22  
23 **Thread Safety:** This type is safe for multithreaded operations.

### 24 25 Description

26 System.DateTime values are formatted by the System.DateTime.ParseExact and  
27 System.DateTime.ToString methods according to standard or custom patterns stored  
28 in the properties of a System.Globalization.DateTimeFormatInfo instance. The  
29 standard patterns can be accessed and modified through the associated  
30 System.Globalization.DateTimeFormatInfo properties. [*Note:* The format patterns  
31 and properties of a read-only System.Globalization.DateTimeFormatInfo instance  
32 cannot be changed. To determine whether a  
33 System.Globalization.DateTimeFormatInfo instance is read-only, use the  
34 System.Globalization.DateTimeFormatInfo.IsReadOnly property.]

35  
36  
37  
38 Date and time format patterns are specified using strings called *format specifiers*. A

1 string is interpreted as standard format specifier if it contains exactly one standard  
 2 format specifier character. If the string contains a single character and that character is  
 3 not one of the standard format specifiers, an exception is thrown. If the string contains  
 4 two or more characters, even if the extra characters are white spaces, the string is  
 5 interpreted as a custom format specifier. Format specifiers and format patterns are  
 6 case-sensitive; for example, 'g' and 'G' represent different patterns.

7  
 8 The following table shows the standard format specifiers and the associated format  
 9 pattern defined for the invariant culture. The exact pattern produced by a format  
 10 specifier is influenced by culture-specific date and/or time settings on the current  
 11 system; computers with different date and time settings might display different  
 12 patterns. The asterisk at the end of a format pattern indicates that the preceding  
 13 character can be repeated without changing the meaning of the pattern. For example,  
 14 the pattern "HH\*" indicates that the strings "HH", "HHH", "HHHH", and "HHHHH"  
 15 produce the same result when used with `System.DateTime.ParseExact` and  
 16 `System.DateTime.ToString` methods.

Format Specifier	Format Pattern	Description
d	MM/dd/yyyy	The full date in numeric format ( <code>System.Globalization.DateTimeFormatInfo.ShortDatePattern</code> ).
D	dddd*, dd MMMM* yyyy	The full date including the day of the week and the name of the month ( <code>System.Globalization.DateTimeFormatInfo.LongDatePattern</code> ).
f	dddd*, dd MMMM* yyyy HH*:mm*	The full date and time, including the day of the week and the name of the month ( <code>System.Globalization.DateTimeFormatInfo.LongDatePattern</code> combined with <code>System.Globalization.DateTimeFormatInfo.ShortTimePattern</code> ).
F	dddd*, dd MMMM* yyyy HH*:mm*:ss*	The full date and time, including the seconds ( <code>System.Globalization.DateTimeFormatInfo.FullDateTimePattern</code> equivalent to <code>System.Globalization.DateTimeFormatInfo.LongDatePattern</code> combined with <code>System.Globalization.DateTimeFormatInfo.LongTimePattern</code> ).
g	MM/dd/yyyy HH*:mm*	A general date pattern including the short time form ( <code>System.Globalization.DateTimeFormatInfo.ShortDatePattern</code> combined with <code>System.Globalization.DateTimeFormatInfo.ShortTimePattern</code> ).
G	MM/dd/yyyy HH*:mm*:ss*	A general date pattern including the long time form ( <code>System.Globalization.DateTimeFormatInfo.ShortDatePattern</code> combined with

		<code>System.Globalization.DateTimeFormatInfo.LongTimePattern</code> ).
m, M	<code>MMMM* dd</code>	The full name of the month and the date ( <code>System.Globalization.DateTimeFormatInfo.MonthDayPattern</code> ).
t	<code>HH*:mm*</code>	The time in short format ( <code>System.Globalization.DateTimeFormatInfo.ShortTimePattern</code> ).
T	<code>HH*:mm*:ss*</code>	The time in long format ( <code>System.Globalization.DateTimeFormatInfo.LongTimePattern</code> ).
U	<code>dddd*, dd MMMM* yyyy HH*:mm*:ss*</code>	The full date and time, including the seconds, in the Gregorian calendar ( <code>System.Globalization.DateTimeFormatInfo.FullDateTimePattern</code> ).
y, Y	<code>yyyy MMMM*</code>	The full name of the month and the year in four-digit format ( <code>System.Globalization.DateTimeFormatInfo.YearMonthPattern</code> ).

1  
2 The following table lists custom format specifiers that can be combined to construct  
3 custom patterns. If the custom pattern contains white space characters, characters  
4 enclosed in single or double quotation marks, or characters not defined in the following  
5 table, these characters are considered literals and are included in the output string  
6 unchanged. [*Note:* See the `System.String` class for the list of white space characters.]

Format Pattern	Description	Examples
d	The day of the month as a value in the range 1-31, inclusive. Single-digit days do not have a leading zero.	1 22
dd	The day of the month as a value in the range 1-31, inclusive. Single-digit days have a leading zero.	01 22
ddd	The abbreviated name of the day of the week, as defined in <code>System.Globalization.DateTimeFormatInfo.AbbreviatedDayNames</code> .	Mon
dddd*	The full name of the day of the week, as defined in <code>System.Globalization.DateTimeFormatInfo.DayNames</code> .	Monday
M	The numeric month as a value in the range 1-12, inclusive. Single-digit	2

	months do not have a leading zero.	11
MM	The numeric month as a value in the range 1-12, inclusive. Single-digit months have a leading zero.	02 11
MMM	The abbreviated name of the month, as defined in <code>System.Globalization.DateTimeFormatInfo.AbbreviatedMonthNames</code> .	Feb
MMMM*	The full name of the month, as defined in <code>System.Globalization.DateTimeFormatInfo.MonthNames</code> .	February
y	The year without the century (two-digit). If the value is less than 10, the year is displayed with no leading zero.	0 3
yy	The year without the century (two-digit). If the year without the century is less than 10, the year is displayed with a leading zero.	00 03
yyyy	The year including the century in four digits.	2000 2003
g*	The name of a period or era (such as "A.D." or "B.C."). This pattern is ignored if the date to be formatted does not have an associated period or era string.	A.D.
h	The hour within a 12-hour range as a value in the range 1-12, inclusive. Single-digit hours do not have a leading zero. [ <i>Note:</i> The value represents whole hours passed since either midnight (12) or noon (12). To distinguish between values occurring before and after noon, include the "t" or "tt*" custom format specifier.]	3 11
hh*	The hour within a 12-hour range as a value in the range 1-12, inclusive. Single-digit hours have a leading zero. [ <i>Note:</i> The value represents whole hours passed since either midnight (12) or noon (12). To distinguish between values occurring before and after noon, include the "t" or "tt*" custom format specifier.]	03 11
H	The hour as a value in the range 0-23, inclusive. Single-digit hours do not have a leading zero. [ <i>Note:</i> The value represents whole hours passed since midnight.]	3 13

HH*	The hour as a value in the range 0 and 23, inclusive. Single-digit hours have a leading zero. [ <i>Note:</i> The value represents whole hours passed since midnight.]	03 13
m	The minute as a value in the range 0-59, inclusive. Single-digit minutes do not have a leading zero. [ <i>Note:</i> The value represents whole minutes passed since the last hour.]	5 15
mm*	The minute as a value in the range 0-59, inclusive. Single-digit minutes have a leading zero. [ <i>Note:</i> The value represents whole minutes passed since the last hour.]	05 15
s	The second as a value in the range 0-59, inclusive. Single-digit seconds do not have a leading zero. [ <i>Note:</i> The value represents whole seconds passed since the last minute.]	1 30
ss*	The second as a value in the range 0-59, inclusive. Single-digit seconds have a leading zero. [ <i>Note:</i> The value represents whole seconds passed since the last minute.]	01 30
f	Displays fractional seconds represented in one digit.	1
ff	Displays fractional seconds represented in two digits.	01
fff	Displays fractional seconds represented in three digits.	001
ffff	Displays fractional seconds represented in four digits.	0001
fffff	Displays fractional seconds represented in five digits.	00001
ffffff	Displays fractional seconds represented in six digits.	000001
fffffff	Displays fractional seconds represented in seven digits.	0000001
t	The first character of the AM/PM designator defined in the <code>System.Globalization.DateTimeFormatInfo</code> property <code>System.Globalization.DateTimeFormatInfo.AMDesignator</code> or <code>System.Globalization.DateTimeFormatInfo.PMDesignator</code> . [ <i>Note:</i> If the total number of hours passed since midnight is less than 12, the A.M. designator is used; otherwise the P.M. designator is used.]	A P
tt*	The AM/PM designator defined in the <code>System.Globalization.DateTimeFormatInfo</code> property <code>System.Globalization.DateTimeFormatInfo.AMDesignator</code> or	AM

	System.Globalization.DateTimeFormatInfo.PMDesignator. [ <i>Note:</i> If the total number of hours passed since midnight is less than 12, the A.M. designator is used; otherwise the P.M. designator is used.]	PM
z	The time zone offset (hour only) from the universal time coordinate (UTC) time (Greenwich Mean Time) as a value in the range -12 to +13, inclusive. Single-digit hours do not have a leading zero. [ <i>Note:</i> The value always includes a leading sign (zero is '+0'), indicating hours ahead of UTC time (+) or hours behind UTC time (-). The offset takes Daylight Savings Time into account.]	-8
zz	The time zone offset (hour only) from the UTC time (Greenwich Mean Time) as a value in the range -12 to +13, inclusive. Single-digit hours have a leading zero. [ <i>Note:</i> The value always includes a leading sign (zero is '+0'), indicating hours ahead of UTC time (+) or hours behind UTC time (-). The offset takes Daylight Savings Time into account.]	-08
zzz*	The full time zone offset (hour and minutes) from the UTC time (Greenwich Mean Time) as a value in the range -12:00 to +13:00, inclusive. Single-digit hours and minutes have leading zeros. [ <i>Note:</i> The value always includes a leading sign (zero is '+0'), indicating hours ahead of UTC time (+) or hours behind UTC time (-). The offset takes Daylight Savings Time into account.]	-08:00
:	The invariant culture time separator defined in System.Globalization.DateTimeFormatInfo.TimeSeparator.	:
/	The invariant culture date separator defined in the System.Globalization.DateTimeFormatInfo.DateSeparator.	/
%c	c represents a single custom format character. Produces the custom format pattern associated with the format character c. The %c specifier provides a mechanism for specifying a single custom format character and having it recognized as a custom specifier. This format is intended for characters that define both a custom and a standard format. Note that a format string containing exactly one such character will be interpreted as a standard format specifier unless prefaced with the %. [ <i>Note:</i> For example, for the invariant culture, "%d" produces the single or double digit date, while "d" produces the date in "MM/dd/yyyy" format. Without the %, a format string containing one character would have to include leading or trailing white space to be interpreted as a custom specifier because custom formats are required to have two or more characters.]	"%y" produces a two digit year without a leading zero, and not the "MMMM, yyyy" pattern.

<p>lc</p>	<p>c represents any character predefined as part of a format specifier. Prevents the character from being interpreted as a format specifier (the character is treated as a literal). [<i>Note:</i> In programming languages where the backslash ('\') character is used to specify control sequences such as newline (\n), the backslash character is required to be specified twice. For example, in C#, "\d" is coded as "\\d".]</p>	<p>"\d" produces the character 'd', and not the day of the month.</p>
<p>'xx' or "xx"</p>	<p>xx represents a string of characters of any length. The characters are treated as literals.</p>	<p>"'d'" produces the character 'd', and not the day of the month.</p>

1

2

# 1 DateTimeFormatInfo() Constructor

```
2 [ILAsm]  
3 public rtspecialname specialname instance void .ctor()  
  
4 [C#]  
5 public DateTimeFormatInfo()
```

## 6 Summary

7 Constructs and initializes a new instance of the  
8 System.Globalization.DateTimeFormatInfo class that is culture-independent  
9 (invariant).

## 10 Description

11 The new instance of System.Globalization.DateTimeFormatInfo is not read-only, and  
12 its properties can be modified with user-defined patterns.

13

# 1 DateTimeFormatInfo.Clone() Method

```
2 [ILAsm]  
3 .method public final hidebysig virtual object Clone()  
4 [C#]  
5 public object Clone()
```

## 6 Summary

7 Creates a copy of the current instance.

## 8 Return Value

9 A new `System.Globalization.DateTimeFormatInfo` instance with property values  
10 equal to the property values of the original  
11 `System.Globalization.DateTimeFormatInfo` instance.

## 12 Description

13 The `System.Globalization.DateTimeFormatInfo.Clone` method creates a new  
14 instance of the same type as the current instance, and then copies the contents of each  
15 of the current instance's non-static fields.

16  
17 The new instance is not read-only, and its properties can be modified with user-defined  
18 patterns.

19  
20 [*Note:* This method is implemented to support the `System.ICloneable` interface.]  
21  
22

23

1  
2 **DateTimeFormatInfo.GetAbbreviatedMonthName(System.Int32) Method**  
3

```
4 [ILAsm]  
5 .method public hidebysig instance string GetAbbreviatedMonthName(int32  
6 month)  
7 [C#]  
8 public string GetAbbreviatedMonthName(int month)
```

9 **Summary**

10 Gets the abbreviated name of the specified month based on the culture of the current  
11 thread.

12 **Parameters**

Parameter	Description
<i>month</i>	A System.Int32 from 1 through 13 representing the month name to retrieve.

13  
14 **Return Value**

15 A System.String containing the abbreviated name of the month represented by *month*.  
16 For cultures with 12-month calendars, the empty string is returned as the name of the  
17 13th month.

18 **Description**

19 For the default (culture-invariant) System.Globalization.DateTimeFormatInfo  
20 instance, this method returns one of the following strings:

	Return Value
1	"Jan"
2	"Feb"
3	"Mar"
4	"Apr"
5	"May"

6	"Jun"
7	"Jul"
8	"Aug"
9	"Sep"
10	"Oct"
11	"Nov"
12	"Dec"
13	""

1  
2  
3  
4

[*Note:* This method supports calendars with 13 months.]

#### 5 Exceptions

Exception	Condition
<b>System.ArgumentOutOfRangeException</b>	<i>month</i> is less than 1 or greater than 13.

6  
7

# 1 DateTimeFormatInfo.GetEra(System.String) 2 Method

```
3 [ILAsm]  
4 .method public hidebysig instance int32 GetEra(string eraName)  
5 [C#]  
6 public int GetEra(string eraName)
```

## 7 Summary

8 Gets a `System.Int32` representing the specified era.

## 9 Parameters

Parameter	Description
<i>eraName</i>	A <code>System.String</code> containing the name of the era.

## 10 11 Return Value

12 A `System.Int32` representing the era. If *eraName* is invalid, returns -1.

## 13 Description

14 The value specified for *eraName* is case-insensitive.

15  
16 [Note: An era name is a culturally specific name for a period of time marked by  
17 distinctive characters or reckoned from a fixed point or event. For example "A.D." and  
18 "B.C." are two eras of the Gregorian calendar.]  
19  
20

## 21 Exceptions

Exception	Condition
<code>System.ArgumentNullException</code>	<i>eraName</i> is a null reference.

22

23

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22

# DateTimeFormatInfo.GetEraName(System.Int32) Method

```
[ILAsm]  
.method public hidebysig instance string GetEraName(int32 era)  
  
[C#]  
public string GetEraName(int era)
```

## Summary

Gets the System.String containing the name of the specified era.

## Parameters

Parameter	Description
<i>era</i>	A System.Int32 representing the era.

## Return Value

A System.String containing the name of the era.

## Description

[Note: An era name is a culturally specific name for a period of time marked by distinctive characters or reckoned from a fixed point or event. For example "A.D." and "B.C." are two eras of the Gregorian calendar.]

## Exceptions

Exception	Condition
<b>System.ArgumentOutOfRangeException</b>	<i>era</i> does not represent a valid era in calendar for the current thread.

1  
2 **DateTimeFormatInfo.GetFormat(System.Type**  
3 **) Method**

```
4 [ILAsm]  
5 .method public final hidebysig virtual object GetFormat(class System.Type  
6 formatType)  
7 [C#]  
8 public object GetFormat(Type formatType)
```

9 **Summary**

10 Returns an object of the specified type that provides System.DateTime formatting  
11 services.

12 **Parameters**

Parameter	Description
<i>formatType</i>	The System.Type of the formatting object to be returned.

13  
14 **Return Value**

15 The current instance, if *formatType* is of type  
16 System.Globalization.DateTimeFormatInfo; otherwise, a null reference.

17 **Description**

18 [*Note:* This method is implemented to support the System.IFormatProvider interface.]  
19  
20

21

# 1 2 DateTimeFormatInfo.GetMonthName(System. 3 Int32) Method

```
4 [ILAsm]  
5 .method public hidebysig instance string GetMonthName(int32 month)  
6 [C#]  
7 public string GetMonthName(int month)
```

## 8 Summary

9 Gets the full name of the specified month based on the culture of the current thread.

## 10 Parameters

Parameter	Description
<i>month</i>	A <code>System.Int32</code> from 1 through 13 representing the month name to retrieve.

## 11 12 Return Value

13 A `System.String` containing the full name of the month represented by *month*. For  
14 cultures with 12-month calendars the empty string is returned as the name of the 13th  
15 month.

## 16 Description

17 For the default (culture invariant) `System.Globalization.DateTimeFormatInfo`  
18 instance, this method returns one of the following strings:

	Return Value
1	"January"
2	"February"
3	"March"
4	"April"
5	"May"
6	"June"

7	"July"
8	"August"
9	"September"
10	"October"
11	"November"
12	"December"
13	""

1  
2  
3  
4

[*Note:* This method supports calendars with 13 months.]

5 **Exceptions**

Exception	Condition
<b>System.ArgumentOutOfRangeException</b>	<i>month</i> is less than 1 or greater than 13.

6  
7

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25

# DateTimeFormatInfo.ReadOnly(System.Globalization.DateTimeFormatInfo) Method

```
[ILAsm]  
.method public hidebysig static class  
System.Globalization.DateTimeFormatInfo ReadOnly(class  
System.Globalization.DateTimeFormatInfo dtfi)  
  
[C#]  
public static DateTimeFormatInfo ReadOnly(DateTimeFormatInfo dtfi)
```

## Summary

Returns a read-only copy of the specified instance of System.Globalization.DateTimeFormatInfo.

## Parameters

Parameter	Description
<i>dtfi</i>	The System.Globalization.DateTimeFormatInfo to copy.

## Return Value

A read-only instance of System.Globalization.DateTimeFormatInfo that is a copy of *dtfi*.

## Description

[Note: Use a read-only System.Globalization.DateTimeFormatInfo copy to prevent modifications to the specified instance.]

## Exceptions

Exception	Condition
System.ArgumentNullException	<i>dtfi</i> is a null reference.

# 1 DateTimeFormatInfo.AbbreviatedDayNames 2 Property

```
3 [ILAsm]  
4 .property string[] AbbreviatedDayNames { public hidebysig specialname  
5 instance string[] get_AbbreviatedDayNames() public hidebysig specialname  
6 instance void set_AbbreviatedDayNames(string[] value) }  
7 [C#]  
8 public string[] AbbreviatedDayNames { get; set; }
```

## 9 Summary

10 Gets or sets a one-dimensional array of type `System.String` containing the culture-  
11 specific abbreviated names of the days of the week.

## 12 Property Value

13 A one-dimensional array of type `System.String` containing the culture-specific  
14 abbreviated names of the days of the week.

## 15 Description

16 The array specified in a set operation is required to be one-dimensional and have  
17 exactly seven elements. The first element of the array is the abbreviated day name for  
18 Sunday, and the last element is the name for Saturday.

19  
20 The property value of the culture invariant  
21 `System.Globalization.DateTimeFormatInfo.AbbreviatedDayNames` is a  
22 `System.String` array that contains "Sun", "Mon", "Tue", "Wed", "Thu", "Fri" and "Sat".

## 23 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.ArgumentException</b>	The value specified for a set operation is not an array with exactly 7 elements.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

24

25

# 1 2 DateTimeFormatInfo.AbbreviatedMonthName 3 s Property

```
4 [ILAsm]  
5 .property string[] AbbreviatedMonthNames { public hidebysig specialname  
6 instance string[] get_AbbreviatedMonthNames() public hidebysig specialname  
7 instance void set_AbbreviatedMonthNames(string[] value) }  
8 [C#]  
9 public string[] AbbreviatedMonthNames { get; set; }
```

## 10 Summary

11 Gets or sets a one-dimensional array of type `System.String` containing the culture-  
12 specific abbreviated names of the months.

## 13 Property Value

14 A one-dimensional array of type `System.String` containing the abbreviated names of  
15 the months. For cultures with 12-month calendars the 13th element of the array is an  
16 empty string.

## 17 Description

18 The `System.Array` specified in a set operation is required to be one-dimensional and  
19 have exactly 13 elements.

20  
21 The property value of the culture invariant  
22 `System.Globalization.DateTimeFormatInfo.AbbreviatedMonthNames` is a  
23 `System.String` array that contains "Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug",  
24 "Sep", "Oct", "Nov", "Dec" and "".

25  
26 [*Note:* The array returned by this property has 13 elements to support calendars with  
27 13 months.]  
28  
29

## 30 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.ArgumentException</b>	The value specified for a set operation is not an array with exactly 13 elements.

<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.
---	--

1

2

# 1 DateTimeFormatInfo.AMDesignator Property

```
2 [ILAsm]  
3 .property string AMDesignator { public hidebysig specialname instance  
4 string get_AMDesignator() public hidebysig specialname instance void  
5 set_AMDesignator(string value) }  
6 [C#]  
7 public string AMDesignator { get; set; }
```

## 8 Summary

9 Gets or sets the `System.String` culture-specific designator for hours that are "ante  
10 meridiem" (before noon).

## 11 Property Value

12 The `System.String` designator for hours that are before noon.

## 13 Description

14 The property value of the culture invariant  
15 `System.Globalization.DateTimeFormatInfo.AMDesignator` returns the  
16 `System.String` "AM".

## 17 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

18

19

# 1 DateTimeFormatInfo.CurrentInfo Property

```
2 [ILAsm]
3 .property class System.Globalization.DateTimeFormatInfo CurrentInfo {
4 public hidebysig static specialname class
5 System.Globalization.DateTimeFormatInfo get_CurrentInfo() }
6 [C#]
7 public static DateTimeFormatInfo CurrentInfo { get; }
```

## 8 Summary

9 Gets a read-only System.Globalization.DateTimeFormatInfo instance that formats  
10 values based on the current culture.

## 11 Property Value

12 A read-only System.Globalization.DateTimeFormatInfo instance based on the culture  
13 of the current thread.

## 14 Description

15 This property is read-only.

16

# 1 DateTimeFormatInfo.DateSeparator Property

```
2 [ILAsm]  
3 .property string DateSeparator { public hidebysig specialname instance  
4 string get_DateSeparator() public hidebysig specialname instance void  
5 set_DateSeparator(string value) }  
  
6 [C#]  
7 public string DateSeparator { get; set; }
```

## 8 Summary

9 Gets or sets the culture-specific `System.String` to use to separate the year, month, and  
10 day components of a date.

## 11 Property Value

12 The `System.String` to use to separate the year, month and day components of a date.

## 13 Description

14 The property value of the culture invariant  
15 `System.Globalization.DateTimeFormatInfo.DateSeparator` is "/".

## 16 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

17

18

# DateTimeFormatInfo.DayNames Property

```
[ILAsm]
.property string[] DayNames { public hidebysig specialname instance
string[] get_DayNames() public hidebysig specialname instance void
set_DayNames(string[] value) }

[C#]
public string[] DayNames { get; set; }
```

## Summary

Gets or sets a one-dimensional array of type `System.String` containing the culture-specific full names of the days of the week.

## Property Value

A one-dimensional array of type `System.String` containing the full names of the days of the week.

## Description

The array specified in a set operation is required to be one-dimensional and have exactly seven elements. The first element of the array is the abbreviated day name for Sunday, and the last element is the name for Saturday.

The property value of the culture invariant `System.Globalization.DateTimeFormatInfo.DayNames` is a `System.String` array that contains "Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday" and "Saturday".

## Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.ArgumentException</b>	The value specified for a set operation is not an array with exactly 7 elements.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

# 1 DateTimeFormatInfo.FullDateTimePattern 2 Property

```
3 [ILAsm]  
4 .property string FullDateTimePattern { public hidebysig specialname  
5 instance string get_FullDateTimePattern() public hidebysig specialname  
6 instance void set_FullDateTimePattern(string value) }  
  
7 [C#]  
8 public string FullDateTimePattern { get; set; }
```

## 9 Summary

10 Gets or sets the format pattern for a long date and long time value.

## 11 Property Value

12 A System.String containing the format pattern for a long date and long time value.

## 13 Description

14 The System.DateTime.ParseExact and System.DateTime.ToString methods associate  
15 the format pattern returned by this property with the 'F' format character.

16  
17 The property value of the culture invariant  
18 System.Globalization.DateTimeFormatInfo.FullDateTimePattern is "dddd, dd  
19 MMMM yyyy HH:mm:ss".

## 20 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

21

22

# 1 DateTimeFormatInfo.InvariantInfo Property

```
2 [ILAsm]  
3 .property class System.Globalization.DateTimeFormatInfo InvariantInfo {  
4 public hidebysig static specialname class  
5 System.Globalization.DateTimeFormatInfo get_InvariantInfo() }  
6 [C#]  
7 public static DateTimeFormatInfo InvariantInfo { get; }
```

## 8 Summary

9 Gets a culture invariant instance of System.Globalization.DateTimeFormatInfo that  
10 is read-only.

## 11 Property Value

12 A read-only System.Globalization.DateTimeFormatInfo instance.

## 13 Description

14 This property is read-only.

15

# 1 DateTimeFormatInfo.IsReadOnly Property

```
2 [ILAsm]  
3 .property bool IsReadOnly { public hidebysig specialname instance bool  
4 get_IsReadOnly() }  
5 [C#]  
6 public bool IsReadOnly { get; }
```

## 7 Summary

8 Gets a System.Boolean value indicating whether the current  
9 System.Globalization.DateTimeFormatInfo instance is read-only.

## 10 Property Value

11 true if the System.Globalization.DateTimeFormatInfo is read-only; otherwise,  
12 false.

## 13 Description

14 This property is read-only.

15

# 1 DateTimeFormatInfo.LongDatePattern

## 2 Property

```
3 [ILAsm]  
4 .property string LongDatePattern { public hidebysig specialname instance  
5 string get_LongDatePattern() public hidebysig specialname instance void  
6 set_LongDatePattern(string value) }  
  
7 [C#]  
8 public string LongDatePattern { get; set; }
```

### 9 Summary

10 Gets or sets the format pattern for a long date value.

### 11 Property Value

12 A System.String containing the format pattern for a long date value.

### 13 Description

14 The System.DateTime.ParseExact and System.DateTime.ToString methods associate  
15 the format pattern returned by this property with the 'D' format character.

16  
17 The property value of the culture invariant  
18 System.Globalization.DateTimeFormatInfo.LongDatePattern is "dddd, dd MMMM  
19 yyyy".

### 20 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

21

22

# 1 DateTimeFormatInfo.LongTimePattern 2 Property

```
3 [ILAsm]  
4 .property string LongTimePattern { public hidebysig specialname instance  
5 string get_LongTimePattern() public hidebysig specialname instance void  
6 set_LongTimePattern(string value) }  
  
7 [C#]  
8 public string LongTimePattern { get; set; }
```

## 9 Summary

10 Gets or sets the format pattern for a long time value.

## 11 Property Value

12 A System.String containing the format pattern for a long time value.

## 13 Description

14 The System.DateTime.ParseExact and System.DateTime.ToString methods associate  
15 the format pattern returned by this property with the 'T' format character.

16  
17 The property value of the culture invariant  
18 System.Globalization.DateTimeFormatInfo.LongTimePattern is "HH:mm:ss".

## 19 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

20

21

# 1 DateTimeFormatInfo.MonthDayPattern 2 Property

```
3 [ILAsm]  
4 .property string MonthDayPattern { public hidebysig specialname instance  
5 string get_MonthDayPattern() public hidebysig specialname instance void  
6 set_MonthDayPattern(string value) }  
  
7 [C#]  
8 public string MonthDayPattern { get; set; }
```

## 9 Summary

10 Gets or sets the format pattern for a month and day value.

## 11 Property Value

12 A System.String containing the format pattern for a month and day value.

## 13 Description

14 The System.DateTime.ParseExact and System.DateTime.ToString methods associate  
15 the format pattern returned by this property with the 'm' and 'M' format characters.

16  
17 The property value of the culture invariant  
18 System.Globalization.DateTimeFormatInfo.MonthDayPattern is "MMMM dd".

## 19 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

20

21

# DateTimeFormatInfo.MonthNames Property

```
[ILAsm]
.property string[] MonthNames { public hidebysig specialname instance
string[] get_MonthNames() public hidebysig specialname instance void
set_MonthNames(string[] value) }

[C#]
public string[] MonthNames { get; set; }
```

## Summary

Gets or sets a one-dimensional array of type `System.String` containing the culture-specific full names of the months.

## Property Value

A one-dimensional array of type `System.String` containing the full names of the months. For cultures with 12-month calendars the 13th element of the array is an empty string.

## Description

The array specified in a set operation is required be one-dimensional and have exactly 13 elements.

The property value of the culture invariant `System.Globalization.DateTimeFormatInfo.MonthNames` is a `System.String` array that contains "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December" and "".

[*Note:* The array returned by this property has 13 elements to support calendars with 13 months.]

## Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.ArgumentException</b>	The value specified for a set operation is not an array with exactly 13 elements.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

1

2

# 1 DateTimeFormatInfo.PMDesignator Property

```
2 [ILAsm]  
3 .property string PMDesignator { public hidebysig specialname instance  
4 string get_PMDesignator() public hidebysig specialname instance void  
5 set_PMDesignator(string value) }  
6 [C#]  
7 public string PMDesignator { get; set; }
```

## 8 Summary

9 Gets or sets the culture-specific System.String designator for hours that are "post  
10 meridiem" (after noon).

## 11 Property Value

12 The System.String designator for hours that are after noon.

## 13 Description

14 The property value of the culture invariant  
15 System.Globalization.DateTimeFormatInfo.PMDesignator is "PM".

## 16 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

17

18

# 1 DateTimeFormatInfo.ShortDatePattern 2 Property

```
3 [ILAsm]  
4 .property string ShortDatePattern { public hidebysig specialname instance  
5 string get_ShortDatePattern() public hidebysig specialname instance void  
6 set_ShortDatePattern(string value) }  
  
7 [C#]  
8 public string ShortDatePattern { get; set; }
```

## 9 Summary

10 Gets or sets the format pattern for a short date value.

## 11 Property Value

12 A System.String containing the format pattern for a short date value.

## 13 Description

14 The System.DateTime.ParseExact and System.DateTime.ToString methods associate  
15 the format pattern returned by this property with the 'd' format character.

16  
17 The property value of the culture invariant  
18 System.Globalization.DateTimeFormatInfo.ShortDatePattern is "MM/dd/yyyy".

## 19 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

20

21

# 1 DateTimeFormatInfo.ShortTimePattern 2 Property

```
3 [ILAsm]  
4 .property string ShortTimePattern { public hidebysig specialname instance  
5 string get_ShortTimePattern() public hidebysig specialname instance void  
6 set_ShortTimePattern(string value) }  
  
7 [C#]  
8 public string ShortTimePattern { get; set; }
```

## 9 Summary

10 Gets or sets the format pattern for a short time value.

## 11 Property Value

12 A System.String containing the format pattern for a short time value.

## 13 Description

14 The System.DateTime.ParseExact and System.DateTime.ToString methods associate  
15 the format pattern returned by this property with the 't' format character.

16  
17 The property value of the culture invariant  
18 System.Globalization.DateTimeFormatInfo.ShortTimePattern is "HH:mm".

## 19 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

20

21

# 1 DateTimeFormatInfo.TimeSeparator Property

```
2 [ILAsm]  
3 .property string TimeSeparator { public hidebysig specialname instance  
4 string get_TimeSeparator() public hidebysig specialname instance void  
5 set_TimeSeparator(string value) }  
6 [C#]  
7 public string TimeSeparator { get; set; }
```

## 8 Summary

9 Gets or sets the culture-specific `System.String` to use to separate the components of  
10 time values (hour, minutes, seconds).

## 11 Property Value

12 The `System.String` to use to separate the components of time; that is, the hour, the  
13 minutes and the seconds.

## 14 Description

15 The property value of the culture invariant  
16 `System.Globalization.DateTimeFormatInfo.TimeSeparator` is ":".

## 17 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

18

19

# 1 DateTimeFormatInfo.YearMonthPattern 2 Property

```
3 [ILAsm]  
4 .property string YearMonthPattern { public hidebysig specialname instance  
5 string get_YearMonthPattern() public hidebysig specialname instance void  
6 set_YearMonthPattern(string value) }  
  
7 [C#]  
8 public string YearMonthPattern { get; set; }
```

## 9 Summary

10 Gets or sets the format pattern for a year and month value.

## 11 Property Value

12 The format pattern for a year and month value.

## 13 Description

14 The `System.DateTime.ParseExact` and `System.DateTime.ToString` methods associate  
15 the format pattern returned by this property with the 'y' and 'Y' format character.

16  
17 The property value of the culture invariant  
18 `System.Globalization.DateTimeFormatInfo.YearMonthPattern` is "yyyy MMMM".

## 19 Exceptions

Exception	Condition
<b>System.ArgumentNullException</b>	The value specified for a set operation is a null reference.
<b>System.InvalidOperationException</b>	The current instance is read-only and a set operation was attempted.

20  
21