

Parsing exact dates in C# shouldn't force you to create an IFormatProvider

Asked 16 years, 3 months ago Modified 9 years, 11 months ago Viewed 4k times



Someone please correct me if I'm wrong, but parsing a yyyy/MM/dd (or other specific formats) dates in C# **should** be as easy as

5



```
DateTime.ParseExact(theDate, "yyyy/MM/dd");
```

but no, C# forces you to create an IFormatProvider.



Is there an app.config friendly way of setting this so I don't need to do this each time?

```
DateTime.ParseExact(theDate, "yyyy/MM/dd", new CultureInfo("en-CA", true));
```

c#

.net

datetime

parsing

globalization

Share

Improve this question

Follow

edited Dec 23, 2014 at 15:47



Bohemian ♦

424k ● 100 ● 598 ● 743

asked Sep 15, 2008 at 16:52



AlanR

1,172 ● 4 ● 14 ● 26

9 Answers

Sorted by: Highest score (default)



The IFormatProvider argument can be null.

14

Share Improve this answer Follow

answered Sep 15, 2008 at 16:54



s n



From the MSDN docs: "If provider is a null reference (Nothing in Visual Basic), the CultureInfo object that corresponds to the current culture is used." – John Sheehan Sep 15, 2008 at 16:57

I realize I should not have implied that app.config needed to be used. You got at what I wanted -- 'just parse the date and don't bother me with stuff I don't care about'. Thanks.

– [AlanR](#) Sep 15, 2008 at 17:30



ParseExact needs a culture : consider "yyyy MMM dd". MMM will be a localized month name that uses the current culture.

6

Share Improve this answer Follow

answered Sep 15, 2008 at 17:04



Joe



Use the current application culture:

5

```
DateTime.ParseExact("2008/12/05", "yyyy/MM/dd",  
    System.Globalization.CultureInfo.CurrentCulture);
```

You can set the application culture in the app.config using the Globalization tag. I think.

Share Improve this answer Follow

answered Sep 15, 2008 at 16:59



[David J. Sokol](#)

3,556 ● 3 ● 32 ● 25



Create an extension method:

3

```
public static DateTime ParseExactDateTime(this string dateString, string  
    formatString) {  
    return DateTime.ParseExact(dateString, formatString, new CultureInfo("en-  
    CA", true));  
}
```

Share

Improve this answer

Follow

edited Sep 15, 2008 at 17:01



user1228

answered Sep 15, 2008 at 16:56



[John Sheehan](#)

78.1k ● 30 ● 161 ● 194



Thanks Will. I didn't really think that one through very well. – [John Sheehan](#) Sep 15, 2008 at 17:05



3

It requires the format provider in order to determine the particular date and time symbols and strings (such as names of the days of the week in a particular language). You can use a null, in which case the CultureInfo object that corresponds to the current culture is used.



If you don't want to have to specify it each time, create an extension method which either passes null or CultureInfo("en-CA", true) as the format provider.

[Share](#) [Improve this answer](#) [Follow](#)

answered Sep 15, 2008 at 17:06



[Scott Dorman](#)

42.5k ● 12 ● 81 ● 112



1

You could also simply create the IFormatProvider once and store it for later use.

[Share](#) [Improve this answer](#) [Follow](#)

answered Sep 15, 2008 at 16:59



[Jeff Hubbard](#)

9,882 ● 3 ● 32 ● 29



1

You could also use the Convert class

```
Convert.ToDateTime("2008/11/25");
```



[Share](#) [Improve this answer](#) [Follow](#)

answered Sep 15, 2008 at 17:02



[Xian](#)

76.6k ● 12 ● 45 ● 49



0

//Convert date to MySql compatible format

```
DateTime DateValue = Convert.ToDateTime(datetimepicker.text);
```

```
string datevalue = DateValue.ToString("yyyy-MM-dd");
```



[Share](#) [Improve this answer](#) [Follow](#)

answered Jan 7, 2009 at 20:25



[Nkemjika](#)



What's wrong with using `Globalization.CultureInfo.InvariantCulture` ?

0

[Share](#) [Improve this answer](#) [Follow](#)

answered Mar 2, 2010 at 17:15



[Nicholas Head](#)

3,726 ● 5 ● 29 ● 39



Nothing, just dont' want to litter my code w/ it since Globalization isn't a concern in our shop.

– [AlanR](#) Mar 8, 2010 at 14:51
