

**عنوان:** از متد `DateTime.ToString` بدون پارامتر استفاده نکنید!

**نویسنده:** وحید نصیری

**تاریخ:** ۲۱:۵۱۳۹۱/۰۸/۲۶

**آدرس:** [www.dotnettips.info](http://www.dotnettips.info)

**برچسب‌ها:** C#, DateTime

در حین تهیه کتابخانه Silverlight DatePicker فارسی، گاهی استفاده کنندگان گزارش می‌دادند که برنامه روی سیستم‌های مختلف کرش می‌کند یا تبدیل تاریخ درست انجام نمی‌شود. مشکل هم پس از بررسی طولانی به این ترتیب مشخص شد که استفاده از `DateTime.ToString` بدون ذکر پارامترهایی که در ادامه توضیح داده خواهند شد، اشتباه است.

### متد `ToString` بر اساس تنظیمات محلی عمل می‌کند

خروجی فراخوانی ذیل

```
DateTime.Now.ToString()
```

در یک سیستم می‌تواند

```
01/11/2012 09:49:08 ق.ظ
```

و در سیستمی دیگر

```
11/1/2012 9:49:08 AM
```

باشد.

این مساله خصوصا برای ذخیره سازی و پردازش اطلاعات به صورت رشته بسیار مهم و مساله ساز است. فرض کنید در یک شبکه با تنظیمات محلی متفاوت، کاربران اطلاعات تاریخ متفاوتی را به بانک اطلاعاتی ارسال کنند. پردازش صحیح این تاریخ‌ها تقریبا غیرممکن است. حالت اول روز 11 ماه یک را نمایش می‌دهد و حالت دوم روز 1 ماه 11. در حالیکه هر دو تاریخ در یک روز ثبت شده‌اند اما تنظیمات محلی کاربران متناظر یکسان نبوده است. برای رفع این مشکل نیاز است `ToString` را مستقل از تنظیمات محلی کاربران کرد:

```
DateTime.Now.ToString(CultureInfo.InvariantCulture)
```

این مورد نکته‌ای است که اگر از `FxCop` برای آنالیز اسمبلی‌های برنامه خود استفاده کنید، حتما گوشزد خواهد شد. همچنین `ReSharper` نیز رعایت آن را در نگارش‌های اخیر خود گنجانده است.

مشکل دیگر مشابه در حین کار با Silverlight و WPF، استفاده و پردازش `e.NewValue` تغییرات خواص است. این مقدار به صورت `object` ارسال می‌شود و برای پردازش آن نباید `e.NewValue.ToString` فراخوانی شود. روش صحیح دریافت تاریخ از آن باید به صورت زیر باشد:

```
public static DateTime? DateTimeTryParse(object data)
{
    if (data == null)
        return null;

    if (data.GetType().Equals(typeof(DateTime)))
        return (DateTime)data;

    DateTime result;
    if (DateTime.TryParse((string)data, CultureInfo.InvariantCulture, DateTimeStyles.None, out result))
        return result;

    return null;
}
```

دو نکته در اینجا قابل توجه است:

- در متد `DateTime.TryParse` بجای `data.ToString`، به `string` تبدیل شده است. متد `DateTime.TryParse` نیز حالت ویژه‌ای پیدا کرده و `CultureInfo.InvariantCulture` در آن قید شده است.
- همچنین چون نوع `e.NewValue` تغییرات دریافتی از نوع `object` می‌باشد، بهتر است ابتدا بررسی شود که آیا `DateTime` است یا خیر. سپس سایر بررسی‌ها صورت گیرد.

## نظرات خوانندگان

نویسنده: رضا نباتی  
تاریخ: ۲۱:۲۳ ۱۳۹۱/۰۸/۲۶

به نظر بنده بهترین راه برای گرفتن خروجی دلخواه ارسال فرمت استرینگ DateTime برای نمایش آن در خروجی می باشد.

```
DateTime thisDate1 = new DateTime(2011, 6, 10);  
Console.WriteLine("Today is " + thisDate1.ToString("MMMM dd, yyyy") + ".");
```

نویسنده: بهروز راد  
تاریخ: ۸:۲۷ ۱۳۹۱/۰۸/۲۷

البته ابزار Code Analysis موجود در Visual Studio Ultimate هم از FxCop در پشت صحنه استفاده می کنه و این موارد رو گوشزد می کنه.

نویسنده: میثم جوادی  
تاریخ: ۱۲:۷ ۱۳۹۱/۰۸/۲۷

من هم یه بار به همچین مشکلی ، شبیه به این برخوردی بودم

[توضیحات](#)

نویسنده: صابر فتح الهی  
تاریخ: ۱۴:۱۴ ۱۳۹۱/۰۸/۲۷

همونطوری که مهندس توی پست شون ذکر کردن نسخه جدید reSharper هم این عمل گوشزد می کنه