

قبلا در همین وبسایت در مورد یکسان‌سازی حروف "ی" و "ک" [مطلبی](#) بیان شده است. تمرکز آن مطلب بر روی اعمال تغییرات، قبل از ذخیره در دیتابیس با استفاده از EF است. به عبارتی، متن‌هایی که توسط مدیر سایت یا هر کاربر دیگری برای ذخیره شدن در دیتابیس وارد شده است. اما در صورتی که کاربری در جستجوی خود از حروف "ی" و "ک" عربی استفاده کند چه می‌شود؟ در اینجا می‌خواهیم روشی را پیاده‌سازی کنیم که عمومی بوده و بتواند هر دوی این حالات را پوشش دهد.

مسئله این است که بایستی تمام ورودی‌های کاربران سایت که از نوع رشته هستند چک و در صورت نیاز اصلاح شوند. بهترین روشی که به ذهن من می‌رسد این است که در فرایند Model Binding این عمل انجام بگیرد. با تعریف یک Model Binder برای نوع رشته می‌توانیم عمل یکدست‌سازی را به صورت عمومی در تمام وبسایت اعمال کنیم.

در MVC یک Model Binder پیش‌فرض داریم به نام [DefaultModelBinder](#). ما هم از همین کلاس استفاده می‌کنیم تا تمام کارها را برای ما انجام دهد. تنها بایستی در متد [BindModel](#) آن کد موردنظر خود را اضافه کنیم و سپس اجازه دهیم بقیه فرایند به شکل عادی ادامه پیدا کند.

کلاسی به نام StringModelBinder اضافه کرده و کلاس DefaultModelBinder را به عنوان کلاس پایه آن تعریف می‌کنیم. سپس متد BindModel آنرا override کرده، کد مربوط به یکدست‌سازی را اضافه می‌کنیم:

```
public class StringModelBinder : DefaultModelBinder
{
    public override object BindModel(ControllerContext controllerContext, ModelBindingContext bindingContext)
    {
        object value = base.BindModel(controllerContext, bindingContext);
        if (value == null)
        {
            return value;
        }
        return value.ToString().Replace((char)1610, (char)1740).Replace((char)1603, (char)1705);
    }
}
```

ابتدا مقدار به دست آمده توسط متد پیش‌فرض را می‌گیریم. سپس در صورت نیاز یکسان‌سازی را انجام می‌دهیم.

برای فعال کردن این Model Binder بایستی آنرا در رویداد Application_start برنامه، برای نوع رشته به ModelBinder های برنامه اضافه کنیم:

```
ModelBinders.Binders.Add(typeof(string), new StringModelBinder());
```

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۳۹۴/۰۱/۰۶ ۲۱:۴۱

در مطلب تکمیلی « [یک دست سازی ی و ک در برنامه‌های Entity framework 6](#) » روش دیگری برای اینکار معرفی شده‌است. در این حالت تمام کوئری‌هایی که توسط EF صادر می‌شوند و تمام پارامترهای آن‌ها پیش از ارسال به بانک اطلاعاتی، تحت کنترل قرار می‌گیرند (هر دو حالت کوئری‌های select و یا insert/update/delete توسط interceptorها در اختیار هستند و نه فقط حالت insert/update/delete [مطلب قبلی](#)).