

در حین بروز استثنای Entity framework، می‌توان توسط ابزارهای Logging متنوعی مانند [ELMAH](#)، جزئیات متداول آن‌ها را برای بررسی‌های آتی ذخیره کرد. اما این جزئیات فاقد SQL نهایی تولیدی و همچنین پارامترهای ورودی توسط کاربر یا تنظیم شده توسط برنامه هستند. برای اینکه بتوان این جزئیات را نیز ثبت کرد، می‌توان یک IDbCommandInterceptor جدید را طراحی کرد.

کلاس ExceptionsInterceptor

در اینجا نمونه‌ای از یک پیاده‌سازی اینترفیس IDbCommandInterceptor را مشاهده می‌کنید. همچنین طراحی یک متد عمومی که می‌تواند به جزئیات SQL نهایی و پارامترهای آن دسترسی داشته باشد، در اینترفیس IExceptionsLogger ذکر شده است.

```
public interface IExceptionsLogger
{
    void LogException<TResult>(DbCommand command,
        DbCommandInterceptionContext<TResult> interceptionContext);
}

using System.Data.Common;
using System.Data.Entity.Infrastructure.Interception;

namespace ElmahEFLogger
{
    public class ExceptionsInterceptor : IDbCommandInterceptor
    {
        private readonly IExceptionsLogger _exceptionsLogger;

        public ExceptionsInterceptor(IExceptionsLogger exceptionsLogger)
        {
            _exceptionsLogger = exceptionsLogger;
        }

        public void NonQueryExecuted(DbCommand command, DbCommandInterceptionContext<int> interceptionContext)
        {
            _exceptionsLogger.LogException(command, interceptionContext);
        }

        public void NonQueryExecuting(DbCommand command, DbCommandInterceptionContext<int> interceptionContext)
        {
            _exceptionsLogger.LogException(command, interceptionContext);
        }

        public void ReaderExecuted(DbCommand command, DbCommandInterceptionContext<DbDataReader> interceptionContext)
        {
            _exceptionsLogger.LogException(command, interceptionContext);
        }

        public void ReaderExecuting(DbCommand command, DbCommandInterceptionContext<DbDataReader> interceptionContext)
        {
            _exceptionsLogger.LogException(command, interceptionContext);
        }

        public void ScalarExecuted(DbCommand command, DbCommandInterceptionContext<object> interceptionContext)
        {
            _exceptionsLogger.LogException(command, interceptionContext);
        }

        public void ScalarExecuting(DbCommand command, DbCommandInterceptionContext<object> interceptionContext)
        {
            _exceptionsLogger.LogException(command, interceptionContext);
        }
    }
}
```

تهیه یک پیاده سازی سفارشی از IEFExceptionsLogger توسط ELMAH

اکنون که ساختار کلی IDbCommandInterceptor سفارشی برنامه مشخص شد، می توان پیاده سازی خاصی از آن را جهت استفاده از [ELMAH](#) به نحو ذیل ارائه داد:

```
using System;
using System.Data.Common;
using System.Data.Entity.Infrastructure.Interception;
using Elmah;

namespace ElmahEFLogger.CustomElmahLogger
{
    public class ElmahEfExceptionsLogger : IEFExceptionsLogger
    {
        /// <summary>
        /// Manually log errors using ELMAH
        /// </summary>
        public void LogException<TResult>(DbCommand command,
            DbCommandInterceptionContext<TResult> interceptionContext)
        {
            var ex = interceptionContext.OriginalException;
            if (ex == null)
                return;

            var sqlData = CommandDumper.LogSqlAndParameters(command, interceptionContext);
            var contextualMessage = string.Format("{0}{1}OriginalException:{1}{2} {1}", sqlData,
                Environment.NewLine, ex);

            if (!string.IsNullOrEmpty(contextualMessage))
            {
                ex = new Exception(contextualMessage, new ElmahEfInterceptorException(ex.Message));
            }

            try
            {
                ErrorSignal.FromCurrentContext().Raise(ex);
            }
            catch
            {
                ErrorLog.GetDefault(null).Log(new Error(ex));
            }
        }
    }
}
```

در اینجا شیء Command به همراه SQL نهایی تولید و پارامترهای مرتبط است. همچنین interceptionContext.OriginalException جزئیات عمومی استثنای رخ داده را به همراه دارد. می توان این اطلاعات را پس از اندکی نظم بخشیدن، به [متد Raise](#) مربوط به ELMAH ارسال کرد تا جزئیات بیشتری از استثنای رخ داده شده، در لاگ های آن ظاهر شوند.

استفاده از ElmahEfExceptionsLogger جهت طراحی یک Interceptor عمومی

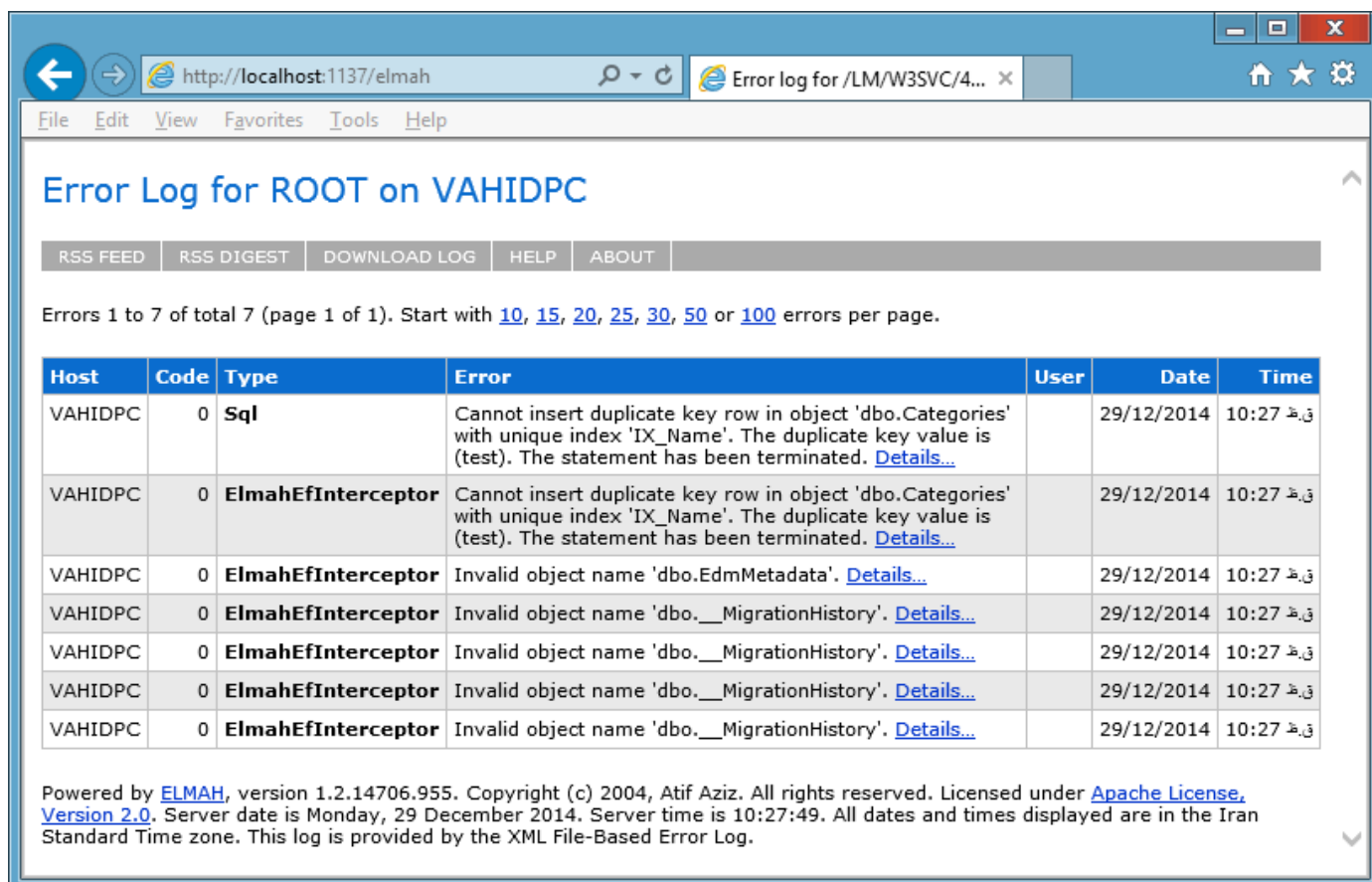
```
public class ElmahEfInterceptor : EFExceptionsInterceptor
{
    public ElmahEfInterceptor()
        : base(new ElmahEfExceptionsLogger())
    { }
}
```

برای استفاده از ElmahEfExceptionsLogger و تهیه یک Interceptor عمومی، می توان با ارث بری از کلاس Interceptor ابتدای بحث شروع کرد و وهله ای از ElmahEfExceptionsLogger را به سازنده ی آن تزریق نمود (یکی از چندین روش ممکن). سپس برای

استفاده از آن کافی است به ابتدای متد Application_Start فایل Global.asax.cs مراجعه و در ادامه سطر ذیل را اضافه نمود:

```
DbInterception.Add(new ElmahEfInterceptor());
```

پس از آن جزئیات کلیه استثنای EF در لاگهای نهایی ELMAH به نحو ذیل ظاهر خواهند شد:



http://localhost:1137/elmah

Error log for /LM/W3SVC/4...

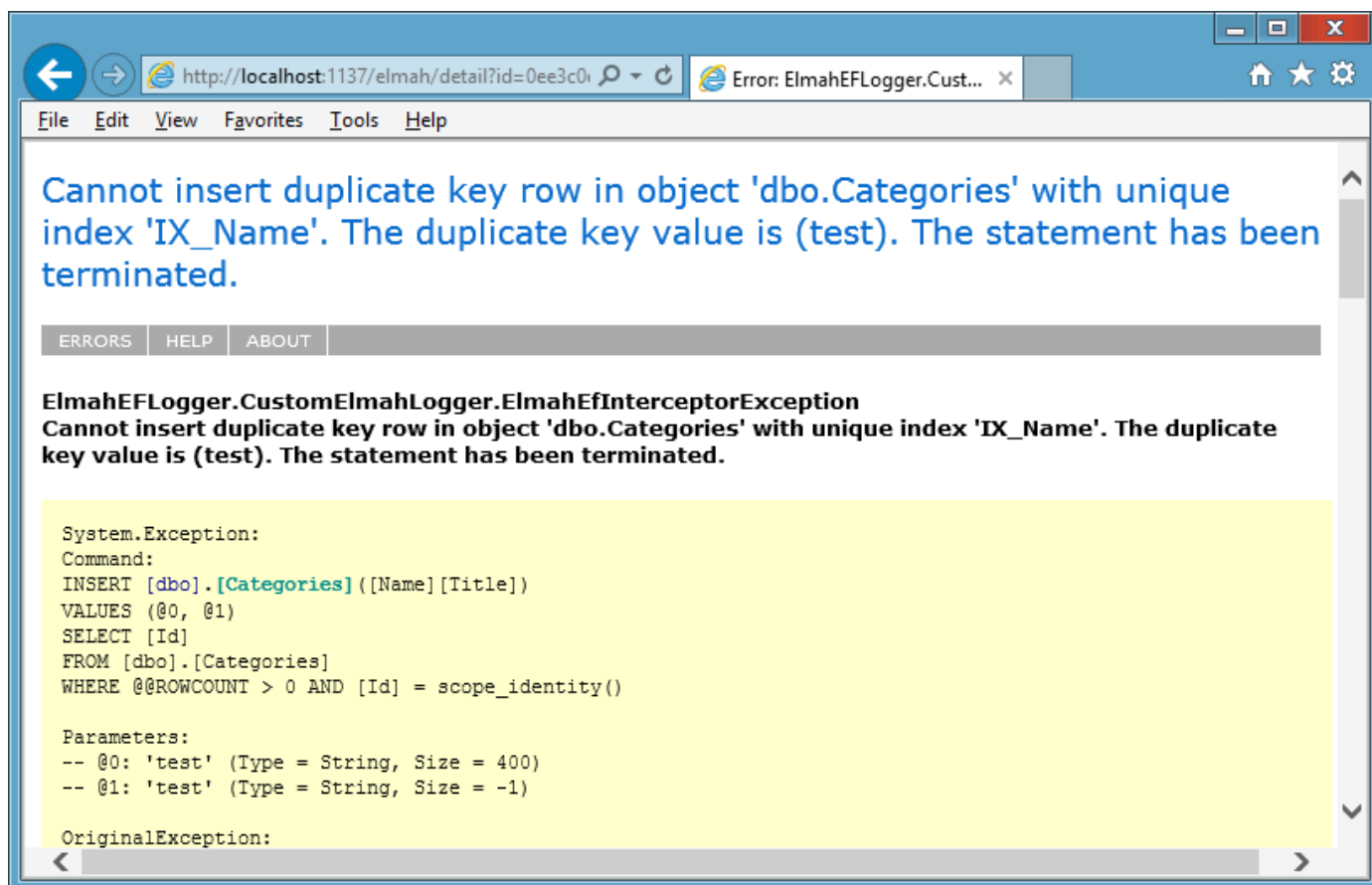
Error Log for ROOT on VAHIDPC

RSS FEED RSS DIGEST DOWNLOAD LOG HELP ABOUT

Errors 1 to 7 of total 7 (page 1 of 1). Start with [10](#), [15](#), [20](#), [25](#), [30](#), [50](#) or [100](#) errors per page.

Host	Code	Type	Error	User	Date	Time
VAHIDPC	0	Sql	Cannot insert duplicate key row in object 'dbo.Categories' with unique index 'IX_Name'. The duplicate key value is (test). The statement has been terminated. Details...		29/12/2014	10:27 ق.ظ
VAHIDPC	0	ElmahEfInterceptor	Cannot insert duplicate key row in object 'dbo.Categories' with unique index 'IX_Name'. The duplicate key value is (test). The statement has been terminated. Details...		29/12/2014	10:27 ق.ظ
VAHIDPC	0	ElmahEfInterceptor	Invalid object name 'dbo.EdmMetadata'. Details...		29/12/2014	10:27 ق.ظ
VAHIDPC	0	ElmahEfInterceptor	Invalid object name 'dbo.__MigrationHistory'. Details...		29/12/2014	10:27 ق.ظ
VAHIDPC	0	ElmahEfInterceptor	Invalid object name 'dbo.__MigrationHistory'. Details...		29/12/2014	10:27 ق.ظ
VAHIDPC	0	ElmahEfInterceptor	Invalid object name 'dbo.__MigrationHistory'. Details...		29/12/2014	10:27 ق.ظ
VAHIDPC	0	ElmahEfInterceptor	Invalid object name 'dbo.__MigrationHistory'. Details...		29/12/2014	10:27 ق.ظ

Powered by [ELMAH](#), version 1.2.14706.955. Copyright (c) 2004, Atif Aziz. All rights reserved. Licensed under [Apache License, Version 2.0](#). Server date is Monday, 29 December 2014. Server time is 10:27:49. All dates and times displayed are in the Iran Standard Time zone. This log is provided by the XML File-Based Error Log.



کدهای کامل این پروژه را از اینجا می‌توانید دریافت کنید:

[ElmahEFLogger](#)

نظرات خوانندگان

نویسنده: صابر فتح الهی
تاریخ: ۲۰:۴۷ ۱۳۹۳/۱۰/۰۸

با سلام و تشکر از مطلب شما

در صورت استفاده از پکیج بالا باید فیلتر کلی که [در پست آموزش MVC قسمت 16](#) توضیح دادین برای ثبت استثنای elmah حذف بشه؟

نویسنده: وحید نصیری
تاریخ: ۲۱:۲۵ ۱۳۹۳/۱۰/۰۸

خیر. کار این Interceptor صرفا لاگ کردن سراسری و نامرئی ریز جزئیات استثنای EF در کل برنامه است. استثنای رخ داده، در ادامه از فیلتر اصلی تنظیم شده رد خواهند شد و یکبار هم در آنجا لاگ می‌شوند. این مورد چون منحصر به EF نیست، بنابراین ضرورتی به حذف آن هم نیست.