

خوب در دو قسمت قبلی ([۱](#) و [۲](#)) با T4 و قواعد کد نویسی Text Template آشنا شدید. در این قسمت برخی مفاهیم را با یک مثال کاربردی‌تر بررسی می‌کنیم.

در ادامه قواعد زیر را در نظر بگیرید :

- 1 - برای استفاده از یک کتابخانه خارجی (dll) در داخل کد Text Template از بلوک `<# assembly @#>` استفاده می‌شود .
مثلا برای استفاده از System.xml کد `<# "assembly name="System.xml" @#>` رو قرار بدید .
- 2- برای import کردن یک فضای نام نیز می‌توانید از بلوک `<# @#>` استفاده نمایید .
به عنوان مثال : `<# "import namespace="System.Data" @#>`
- 3- encoding خروجی قابل تنظیم می‌باشد. مثال: `<# output extension = ".html" encoding = "utf-8" @#>`
- 4- برای استفاده از یک فایل tt درون فایل دیگر میتوان از include استفاده کرد. مثال : `<# include file=" testpath " @#>`

`<# \basetest.tt`

حالا به مثال زیر توجه کنید:

در بسیاری از پروژه‌ها، معادل تمامی جداول موجود در یک دیتابیس Class هایی به نام DTO یا Data transfer object ساخته میشود که عموماً "کلاسهای سبکی هستند که فقط شامل خصوصیت‌های معادل فیلدهای جداول می‌باشند و از آن‌ها جهت مدل کردن داده‌ها و ... استفاده می‌گردد. تولید این کلاسهای ساده می‌تواند بصورت اتوماتیک صورت گیرد و از این جهت در زمان تولید پروژه صرفه جویی شود. همچنین با تغییر ساختار دیتابیس می‌توان همواره کلاسها را بروزرسانی کرد. نمونه ای از کد T4 برای تولید تمامی کلاسهای DTO به شکل زیر میباشد . فقط برای تست کد زیر دقت کنید که ConnectionString را در داخل کد متناسب با دیتابیس خود تغییر دهید:

```
<#@ template language="C#" debug="True" hostspecific="True" #>
<#@ output extension=".cs" #>
<#@ assembly name="System.Data" #>

<#@ assembly name="System.xml" #>
<#@ import namespace="System.Collections.Generic" #>
<#@ import namespace="System.Data.SqlClient" #>
<#@ import namespace="System.Data" #>

using System;
namespace MyProject.Entities
{
    <#
        string connectionString = "Password=22125110;Persist Security Info=True;User ID=sa;Initial
Catalog=DnnDB;Data Source=ABBASPOOR299";
        SqlConnection conn = new SqlConnection(connectionString);
        conn.Open();
        System.Data.DataTable schema = conn.GetSchema("TABLES");
        string selectQuery = "select * from @tableName";
        SqlCommand command = new SqlCommand(selectQuery,conn);
        SqlDataAdapter ad = new SqlDataAdapter(command);
        System.Data.DataSet ds = new DataSet();

        foreach(System.Data.DataRow row in schema.Rows)
        {
            <#
            public class <#= row["TABLE_NAME"].ToString().Trim('s') #>
            {
                <#
                command.CommandText =
selectQuery.Replace("@tableName",row["TABLE_NAME"].ToString());
                ad.FillSchema(ds, SchemaType.Mapped, row["TABLE_NAME"].ToString());

                foreach (DataColumn dc in ds.Tables[row["TABLE_NAME"].ToString()].Columns)
                {
                    <#
                }
            }
        }
    }
}
```

```
private <# dc.DataType.Name #> _<# dc.ColumnName.Replace(dc.ColumnName[0].ToString(), dc.ColumnName[0].ToString().ToLower()) #>;
public <# dc.DataType.Name #> <# dc.ColumnName #>
{
    get { return _<# dc.ColumnName.Replace(dc.ColumnName[0].ToString(),
dc.ColumnName[0].ToString().ToLower()) #>; }
    set { _<# dc.ColumnName.Replace(dc.ColumnName[0].ToString(),
dc.ColumnName[0].ToString().ToLower()) #> = value; }
}

<# } #>

}

<#
} #>

}
```

خوب! خودتون می‌توانید تست کنید ... این یکی از رایج‌ترین استفاده‌های T4 Text Template هست که برنامه نویسان از آن استفاده می‌کنند.

نظرات خوانندگان

نویسنده: افشین عباسپور
تاریخ: ۲۳:۲۵ ۱۳۹۲/۱۱/۳۰

متن رنگی کد این قسمت رو میتونید از [اینجا](#) دانلود کنید

نویسنده: علی صداقت
تاریخ: ۱۲:۱۴ ۱۳۹۲/۱۲/۰۳

با تشکر از مطلب مفید شما. ظاهرا پس از اتمام درج اطلاعات ستونهای یک جدول باید متد `ds.Tables.Clear` را فراخوانی کرد تا جداول موجود در دیتاست پاک شوند. در حال حاضر فیلدهای جدول اول به جای فیلدهای تمامی جداول قرار می گیرند.

نویسنده: افشین عباسپور
تاریخ: ۱۵:۲ ۱۳۹۲/۱۲/۰۳

ممنون دوست عزیز و پوزش از بی دقتی من .
مشکل `Clear` شدن دیتاست نیست . حلقه دو باید تغییر کنه `ds.Tables[0].Columns ...` باید تغییر کنه و تبدیل به `ds.Tables[row["TABLE_NAME"].ToString()].Columns` بشه ! باز هم ممنون از دقت شما

نویسنده: افشین عباسپور
تاریخ: ۱۵:۳۲ ۱۳۹۲/۱۲/۰۳

کد اصلاح شد .

نویسنده: ابوالفضل علیاری
تاریخ: ۲۳:۴۳ ۱۳۹۲/۱۲/۱۶

سلام
من کامل متوجه نشدم ، وقتی برنامه اجرا میشه این کدها به صورت اتوماتیک اجرا میشه؟
ممنون

نویسنده: افشین عباسپور
تاریخ: ۱۴:۱ ۱۳۹۲/۱۲/۱۷

نه دوست عزیز .. این به اجرای برنامه ربطی نداره ! وقتی فایل رو (فایل `tt`) در ویژوال استدیو تغییر میدید و `Save` میکنید بطور اتوماتیک اجرا میشه و خروجی رو در فایل تعیین شده در `output` (بالای کد) ایجاد میکنه ... قسمت چهارم رو مطالعه کنید بیشتر متوجه میشید
ضمن اینکه مثال ساده قسمت دوم رو اجرا کنید تا متوجه عملکرد فایل [Text template](#) بشی ...

نویسنده: افشین عباسپور
تاریخ: ۱۴:۳ ۱۳۹۲/۱۲/۱۷

در مورد اجرای مثال قسمت سوم هم دقت کنید که `connectionString` رو باید متناسب دیتابیس مورد نظر خودتون در کامپیوتر تغییر بدید بعد نتیجه جادویی رو ببینید !