عنوان: **استفاده از DbProviderFactory**
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۵/۱۹ ۱۹:۳۲
آدرس: www.dotnettips.info
برچسب‌ها: ADO.NET

استفاده از DbProviderFactory امکان اتصال به دیتابیس‌های مختلف با یک کد واحد را برای شما فراهم می‌سازد،بطوریکه اگر بخواهید برنامه ای بنویسید که قابلیت اتصال به Oracle و SqlServer و دیگر دیتابیس‌ها را داشته باشد، استفاده از DbProviderFactory ، کار شما را تسهیل می‌نماید.

DbProviderFactory در .Net Framework 2.0 ارائه شده است.برای درک و چگونگی استفاده از DBProviderFactory مثالی را بررسی می‌نماییم.
ابتدا کد زیر را درون یک فرم کپی نمایید:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.Common;

namespace DBFactory
{
    public partial class Form1 : Form
    {
        private string _MySQLProvider = "MySql.Data.MySqlClient";
        private string _SQLProvider="System.Data.SqlClient";
        private string _OracleProvider ="System.Data.OracleClient";
        private DbProviderFactory _DbProviderFactory;
        private DbConnection _DbConnection = null;
        private DbCommand _DbCommand = null;
        private DbDataAdapter _DbDataAdapter = null;

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            try
            {

             string _SQLconnectionstring = "Integrated Security=SSPI;Persist Security
Info=False;Initial Catalog=Test;Data Source=FARHAD-PC";
             string _Oracleconnectionstring = "Data Source=ServiceName;User
Id=Username;Password=Password";

            _DbProviderFactory = DbProviderFactories.GetFactory(_SQLProvider);
            _DbConnection = _DbProviderFactory.CreateConnection();
            _DbConnection.ConnectionString = _SQLconnectionstring;

            _DbConnection.Open();

            if (_DbConnection.State == ConnectionState.Closed)
            {

                MessageBox.Show("اتصال با دیتابیس برقرار نشده است");
            }
            else
            {
                MessageBox.Show("اتصال با دیتابیس با موفقیت بر قرار شده است");
            }
            }
            catch (System.Exception excep)
            {
                MessageBox.Show(excep.Message.ToString());
            }
```

```
        }
    }
}
```

برای استفاد از DBProviderFactory می‌بایست از فضای نامی System.Data.Common استفاده نمایید. بعد از اعلان کلاس فرم تعدادی آبجکت تعریف شده است، که سه آبجکت ابتدایی آن، بیانگر Provider دیتابیس‌های MySQL،SQLSERVER و Oracle می‌باشد:

```
private string _MySQLProvider = "MySql.Data.MySqlClient";
 private string _SQLProvider="System.Data.SqlClient";
 private string _OracleProvider ="System.Data.OracleClient";
```

Providerهای بیان شده، جهت استفاده DBFactory برای تشخیص نوع Database می‌باشد، تا بتواند آبجکت‌های مربوط به دیتابیس را ایجاد و در اختیار برنامه نویس قرار دهد. در این مثال ارتباط با دیتابیس SQLSERVER را امتحان می‌کنیم. بنابراین خواهیم داشت:

```
_DbProviderFactory = DbProviderFactories.GetFactory("System.Data.SqlClient");
```

در کد بالا، Provider، دیتابیس SQLSERVER به DbProviderFactory به عنوان ورودی داده شده است، بنابراین آبجکتهای مربوط به دیتابیس SQL Server ایجاد و در اختیار شما قرار می‌گیرد.

اگر به نام فضای نامی System.Data.Common توجه نمایید،از کلمه Common استفاده شده است و منظور این است که تمامی کلاسهایی را که این فضای نامی ارائه می‌دهد، در هر دیتابیسی قابل استفاده می‌باشد. برای تشخیص، کلاسهای مربوط به این فضای نامی نیز در ابتدای نام آنها از دو حرف DB استفاده شده است. تمامی کلاسهای زیر در فضای نامی System.Data.Common قابل ارائه و استفاده می‌باشد:

```
DbCommand
DbCommandBuilder
DbConnection
DbDataAdapter
DbDataReader
DbException
DbParameter
DbTransaction
```

جهت اطلاع: ممکن است سئوالی در ذهن شما ایجاد شود که دات نت چگونه براساس نام Provider نوع دیتابیس را تشخیص می‌دهد؟
جواب: زمانی که دیتابیس‌های مختلف روی سیستم شما نصب می‌شود، Providerهای مربوط به هر دیتابیس درون فایل Machine.config که مربوط به دات نت میباشد، درج می‌شود. و دات نت براساس اطلاعات مربوط به همین فایل آبجکت‌های دیتابیس را ایجاد می‌نماید.

امیدوارم مطلب فوق مفید واقع شود.

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۱۹ ۱۹:۵۸

من توصیه می‌کنم که ADO.NET رو به شکل خام آن فراموش کنید. این نوع روش‌ها هرچند پایه و اساس تمام ORMهای نوشته شده هستند، اما فقط ابتدای کار را به شما نشان می‌دهند. واقعیت این است که سوئیچ کردن بین بانک‌های اطلاعاتی مختلف نیاز به تولید SQL قابل فهم برای آن موتور خاص را نیز دارد. اینجا است که ORMها در وقت شما صرفه جویی می‌کنند. شما کوئری LINQ می‌نویسید اما در پشت صحنه بر اساس پروایدر مورد استفاده، این کوئری LINQ به معادل SQL قابل فهم برای بانک اطلاعاتی مورد نظر ترجمه می‌شود. خیلی از توابع هستند که در بانک‌های اطلاعاتی مختلف تفاوت می‌کنند و این SQL ایی که مورد بحث است ... در عمل آنچنان استاندارد نیست. توابع تاریخ در SQLite با SQL Server فرق می‌کند. نوع‌های داده‌ای این‌ها عموما تطابق ندارد و مسایل دیگر. ORMها می‌توانند این مسایل را به خوبی مدیریت کنند بدون اینکه شما آنچنان درگیر این جزئیات شوید.

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۵/۱۹ ۲۱:۱۲

سلام
با تشکر از توصیه شما
تا حدودی با نظر شما موافق هستم، اگر بخواهیم با امکانات جدید مایکروسافت نرم افزاری ایجاد نماییم. قطعا، روش بیان شده ضرورتی ندارد، اما برای پروژه هایی که با امکانات قدیمی‌تر نوشته شده اند و بدلایلی امکان بازنویسی آنها وجود، ندارد، و از طرفی میبایست با دیتابیس‌های مختلف نیز کار کند، روش فوق می‌تواند مفید باشد،
در مورد اینکه دیتابیس‌ها با هم متفاوت می‌باشند، نیز با شما موافقم، حتی معتقدم که Provider ی را که مایکروسافت برای Oracle ارائه داده است،در مقایسه با Provider شرکت Oracle بسیار ضعیف‌تر عمل می‌نماید، به عنون مثال در جاهایی که مدت زمان درج اطلاعات زیادی بصورت Batch بسیار اهمیت دارد،Provider، شرکت Oracle برای دیتابیس Oracle سازگارتر و کاراتر میباشد.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۱۹ ۲۱:۳۱

- اگر به هر دلیلی مجبور هستید که از دات نت 2 استفاده کنید، NHibernate می‌تونه پیشنهاد خوبی باشه و نسخه مخصوص دات نت 2 هم دارد (به آرشیو قدیمی آن سایت مراجعه کنید). (پایه زبان فعلی جاوا از خیلی از جهات شبیه به دات نت 2 است)
- مایکروسافت کلا توسعه پروایدر ADO.NET مخصوص اوراکل را رسما متوقف کرده و خود اوراکل الان داره این کار رو ادامه می‌ده . خلاصه از پروایدر مایکروسافت برای کار با اوراکل استفاده نکنید.