

قصد داریم در طی چند پست متوالی، یک پروژه Paint را به صورت شی گرا پیاده سازی کنیم. خوب، پروژه ای که می‌خواهیم پیاده سازی کنیم باید دارای این امکانات باشد که مرحله به مرحله پیش میریم و پروژه کامل در نهایت در قسمت [پروژه‌ها](#) ی همین سایت قرار خواهد گرفت.

قابلیت ترسیم اشیا روی بوم گرافیکی دلخواه
قابلیت جابجایی اشیا
قابلیت تغییر رنگ اشیا
ترسیم اشیا توپر و تو خالی
تعیین پهنای خط شی ترسیم شده
تعیین رنگ پس زمینه در صورت تو پر بودن شی
قابلیت پیش نمایش رسم شکل در زمان ترسیم اشیا
توانایی انتخاب اشیا
تعیین عمق شی روی بوم گرافیکی مورد نظر
ترسیم اشیایی مانند خط، دایره، بیضی، مربع، مستطیل، لوزی، مثلث
قابلیت تغییر اندازه اشیا ترسیم شده

خوب برای شروع ابتدا یک پروژه از نوع Windows Application ایجاد می‌کنیم (البته برای این قسمت می‌توانیم یک پروژه Class Library ایجاد کنیم)

سپس یک پوشه به نام Models به پروژه اضافه نماییم.

خوب در این پروژه یک کلاس پایه به نام Shape در نظر می‌گیریم.

همه اشیا ما دارای نقطه شروع، نقطه پایان، رنگ قلم، حالت انتخاب، رنگ پس زمینه، نوع شی، می‌باشند که بعضی از خصوصیات را توسط خصوصیات دیگر محاسبه می‌کنیم. مثلا خاصیت Width و Height و X و Y توسط خصوصیات نقطه شروع و پایان می‌توانند محاسبه شوند.

ساختار کلاس‌های پروژه ما به صورت زیر است که مرحله به مرحله کلاس‌ها پیاده سازی خواهند شد.



با توجه به تصویر بالا (البته این تجزیه تحلیل شخصی من بوده و دوستان به سلیقه خود ممکن است این ساختار را تغییر دهند)

نوع شمارشی ShapeType: در این نوع شمارشی انواع شی‌های موجود در پروژه معرفی شده است

محتوای این نوع به صورت زیر تعریف شده است:

```

namespace PWS.ObjectOrientedPaint.Models
{
    /// <summary>
    /// Shape Type in Paint
    /// </summary>
    public enum ShapeType
    {
        /// <summary>
        /// هیچ
        /// </summary>
        None = 0,
        /// <summary>
        /// خط
        /// </summary>
        Line = 1,
        /// <summary>
        /// مربع
        /// </summary>
        Square = 2,
        /// <summary>
        /// مستطیل
        /// </summary>
        Rectangle = 3,
        /// <summary>
        /// بیضی
        /// </summary>
        Ellipse = 4,
        /// <summary>
        /// دایره
        /// </summary>
        Circle = 5,
        /// <summary>
        /// لوزی
        /// </summary>
        Diamond = 6,
        /// <summary>
        /// مثلث
        /// </summary>
        Triangle = 7,
    }
}

```

انشالله در پست‌های بعدی ما بقی کلاس‌ها به مرور پیاده سازی خواهند شد.

نظرات خوانندگان

نویسنده: علی صداقت
تاریخ: ۱۷:۱۷ ۱۳۹۱/۱۱/۰۷

مبحث مفید و جالبیست. منتظر ادامه این مبحث هستیم. موفق باشید.

نویسنده: Parham
تاریخ: ۱۰:۳۷ ۱۳۹۱/۱۱/۰۸

نوع شمارشی ShapeType یک فایل کد ساده است، درسته؟

نویسنده: صابر فتح الهی
تاریخ: ۳:۲ ۱۳۹۱/۱۱/۱۲

بله درسته

نویسنده: masoud
تاریخ: ۱۴:۶ ۱۳۹۱/۱۱/۱۴

با تشکر آقای فتح الهی ممنون میشم این بحث رو تا آخر پیش ببرید استفاده میکنیم.

نویسنده: مسعود بهرامی
تاریخ: ۳:۲۰ ۱۳۹۲/۰۸/۳۰

قرار دادن کلاس Square به عنوان sub type زیر کلاس Rectangle اصل LSP رو نقض می‌کنه بهتره که هر دو از Shape به ارث برسند با بهتره بگم Implement کنن

نویسنده: صابر فتح الهی
تاریخ: ۱۰:۴۰ ۱۳۹۲/۰۸/۳۰

بله کاملاً درسته از نظر شی گرای یک مربع نمیتونه مستطیل باشه، این پروژه هنوز تمام نشده و در خروجی نهایی اصلاح خواهد شد الان قابلیت ویرایش وجود نداره.
البته پست بعدی [پیاده سازی پروژه نقاشی \(Paint\) به صورت شی گرا #2](#) نگاه کنین شاید نظراون عوض شد.
موفق و موید باشید

نویسنده: خوزستان
تاریخ: ۱۱:۳۰ ۱۳۹۲/۰۸/۳۰

اصل SPR , LSP , از کجا و چه منابعی برای اینا موجود هست ؟
آیا کتابهای مهندسی نرم افزار باید خونند یا کلن مبحث جداگانه‌ای داره ؟

نویسنده: محسن خان
تاریخ: ۱۲:۷ ۱۳۹۲/۰۸/۳۰

از اینجا شروع کنید. 5 قسمت هست:

[اصول طراحی شی گرا SOLID - #بخش اول اصل SRP](#)

نویسنده: خوزستان
تاریخ: ۱۳۹۲/۰۸/۳۰ ۱۳:۱۱

تشکر .

آیا کتاب فارسی در این باره وجود دارد ؟
اساتید کتابی هست که معرفی کنن ؟

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۲/۰۸/۳۰ ۱۳:۵۹

توی گوگل SOLID در شی گرا جستجو کنین

یا توی همین سایت این [برچسب](#) دنبال کنید

نویسنده: مسعود بهرامی
تاریخ: ۱۳۹۲/۰۹/۰۱ ۳:۵۷

سلام مهندس فتح الهی عزیز من خوندم کامل ولی به نظر من بهتره از هم جدا شدن تا اصل LSP رو نقض نکنن درسته رفتار ترسیمشون مثل همه و دیگه تو کلاس مربع شما تابع رسم رو ننوشتین ولی باعث نقض LSP شدین به نظر من بهتر هردوشون از یه Abstract Class دیگه به ارث برسن و تابع Draw این دوشکل که مثل هم هستش رو بزارین اونجا با اینکار هم LSP رعایت شده و هم تکرار تابع Draw رو هم ندارین
در ضمن تابع رسم پیش نمایش اشکالات اساسی از نظر OOP داشت که من با اجازتون اونو زیر پست خودش میزارم Re factor که به نظرم بهتر میومد. دست گلتونم درد نکنه

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۲/۰۹/۰۱ ۱۲:۸

خوشحال میشم کد شمارو ببینم
منتظر روش شما هستم من که خودم چیزی به ذهنم نرسید

چند روز پیش یک افزونه در nuget نظر من رو به خودش جلب کرد . بعد از دانلود و نصب اون و مقداری کار کردن باهاش جای خودش رو تو دلم باز کرد ولی متاسفانه این افزونه تا 21 روز رایگان بود. توی نت برای پیدا کردن سریال و یا کرکش زیاد گشتم ولی هیچ چیز یافت نشد . شاید به خاطر اینکه از زمان تولیدش زیاد نمیگذره ... در هر حال گذشتن از خیرش برام سخت بود بنابر این به یاد قدیم تصمیم گرفتم خودم دست به کار بشم و release کنمش ...

بعد از deobfuscate کردنش سیستم امنیتی نکات خیلی جالبی داشت که یکیش ایجاد شناسه ی منحصر به فرد برای هر سیستم بود . البته شاید باورش سخت باشه ولی برای بخش امنیتی 23 تا کلاس داشت که هر کدوم کلی تابع و ... داشتن که همه هم به هم مرتبط بود . تا به حال ندیده بودم توی هیچ افزونه ای اینقدر روی امنیتی کار بشه و خب همینم باعث شد 4-5 ساعت وقتمو بگیره ...

کد زیر رو از یکی از کلاس هاش استخراج کردم که توسط اون میتونید یک شناسه منحصر به فرد بر اساس مشخصات پردازنده - برد اصلی و ... تولید کنید . فقط در هنگام استفاده توجه داشته باشید به ارجاع‌های برنامه و اینکه باید System.Management رو به reference های برنامه اضافه بکنید حتما ...

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using Microsoft.Win32;
using System.Management;

namespace ConsoleApplication1
{
    class Program
    {
        private static string GetSystemCode_int1(byte[] byte_0)
        {
            if (byte_0 != null)
            {
                return Convert.ToBase64String((new MD5CryptoServiceProvider()).ComputeHash(byte_0));
            }
            else
            {
                return string.Empty;
            }
        }

        private static string GetSystemCode_int0(string string_2)
        {
            if (!string.IsNullOrEmpty(string_2))
            {
                return GetSystemCode_int1(Encoding.UTF8.GetBytes(string_2));
            }
            else
            {
                return string.Empty;
            }
        }

        public static string GetSystemCode()
        {
            string key = null;
            if (key == null)
            {
                string empty = string.Empty;
                try
                {
                    ManagementClass managementClass = new ManagementClass("win32_processor");
                    ManagementObjectCollection instances = managementClass.GetInstances();
                    foreach (ManagementBaseObject instance in instances)
                {

```

```

        {
            try
            {
                empty = string.Concat(empty,
instance.Properties["processorID"].Value.ToString());
                break;
            }
            catch
            {
            }
        }
    }
    catch
    {
        try
        {
            ManagementObjectSearcher managementObjectSearcher = new
ManagementObjectSearcher("Select * From Win32_BaseBoard");
            foreach (ManagementBaseObject managementBaseObject in
managementObjectSearcher.Get())
            {
                empty = string.Concat(empty,
managementBaseObject["SerialNumber"].ToString().Trim());
            }
        }
        catch
        {
        }
    }
    try
    {
        ManagementObject managementObject = new
ManagementObject("win32_logicaldisk.deviceid=\"C:\\"");
        managementObject.Get();
        empty = string.Concat(empty, managementObject["VolumeSerialNumber"].ToString());
    }
    catch
    {
    }
    if (string.IsNullOrEmpty(empty))
    {
        empty = Environment.MachineName;
    }
    key = GetSystemCode_int0(empty);
}
return key;
}

static void Main(string[] args)
{
    Console.WriteLine(GetSystemCode());
    Console.ReadKey();
}
}

```

نظرات خوانندگان

نویسنده: ابوالفضل حسن الدین
تاریخ: ۱۱:۲۶ ۱۳۹۱/۱۱/۱۳

سلام
ممنون از پست... شاید اگر اجزای بیشتری از سیستم امنیتی یاد شده بیان می‌شد، بازخورد بهتری داشت. می‌شه اسم افزونه رو معرفی کنین؟

نویسنده: سید مهران موسوی
تاریخ: ۱۲:۲۵ ۱۳۹۱/۱۱/۱۳

افزونه‌ی مذکور رو میتونید از [سایت رسمی خودش](#) دریافت کنید .

محتوای این پست صرفا مرتبط است با عنوان پست .

تعدادی از اجزای مهم سیستم امنیتی این افزونه شامل :

- 1) ارتباط با وب سرویس سایت سازنده و دریافت کد فعال سازی موقت مختص به سیستم شما و ثبت اطلاعات سیستم شما در بانک اطلاعاتی سایت است برای جلوگیری استفاده مجدد از نسخه‌ی آزمایشی بعد از 21 روز . (البته بعد از 21 روز فقط تعدادی از قابلیت‌ها غیر فعال میشوند و بقیه رایگان هستند از جمله دیباگینگ php در vs !) .
- 2) ایجاد کد منحصر به فرد هر سیستم .
- 3) ترکیب کد منحصر به فرد با شماره package ای که بعد از نصب افزونه vs به ان اختصاص میدهد و ...

در کل نکات جالبی رو رعایت کرده بود و تعداد زیادی از کلاس‌ها رو پیاده سازی کرده بود که ارتباطاتی با هم داشتند فقط و فقط جهت سخت کردن debugging که از کارش خوشم اومد ...

برای بررسی صحت کد ملی باید کمی با ساختار این کد آشنا شویم. کد ملی ۰۰۷-۷۶۱۱۶۱-۶ را در نظر بگیرید. همانطور که مشاهده می‌کنید این کد به ۳ قسمت تقسیم شده است. قسمت اول که یک عدد ۳ رقمی است نشان دهنده محل تولد فرد است. ۶ رقم وسط شماره شناسایی فرد و رقم آخر، رقم کنترل است. در حقیقت ساختار ۹ رقم اول طبق الگوریتمی قابل بررسی است که خروجی آن الگوریتم همان رقم آخر است. اگر خروجی الگوریتم با رقم کنترل برابر باشد می‌توان گفت کد ملی وارد شده معتبر است و در غیر اینصورت کد ملی معتبر نخواهد بود.

برای این منظور دو متد الحاقی زیر به [پروژه DNT.Extensions](#) اضافه شده است که بدنه آنها به شرح زیر است:

```
/// <summary>
/// Validate IR National Code
/// </summary>
/// <param name="nationalcode">National Code</param>
/// <returns></returns>
public static bool IsValidNationalCode(this string nationalcode)
{
    int last;
    return nationalcode.IsValidNationalCode(out last);
}

/// <summary>
/// Validate IR National Code
/// </summary>
/// <param name="nationalcode">National Code</param>
/// <param name="lastNumber">Last Number Of National Code</param>
/// <returns></returns>
public static bool IsValidNationalCode(this string nationalcode, out int lastNumber)
{
    lastNumber = -1;
    if (!nationalcode.IsItNumber()) return false;
    var array = nationalcode.ToCharArray();
    if (array.Length != 10) return false;
    var j = 10;
    var sum = 0;
    for (var i = 0; i < array.Length - 1; i++)
    {
        sum += Int32.Parse(array[i].ToString(CultureInfo.InvariantCulture)) * j;
        j--;
    }
    var div = sum / 11;
    var r = div * 11;
    var diff = Math.Abs(sum - r);

    if (diff <= 2)
    {
        lastNumber = diff;
        return diff == Int32.Parse(array[9].ToString(CultureInfo.InvariantCulture));
    }
    var temp = Math.Abs(diff - 11);
    lastNumber = temp;
    return temp == Int32.Parse(array[9].ToString(CultureInfo.InvariantCulture));
}
```

طریقه استفاده از این متدها نیز به شرح ذیل است:

```
bool b = "0077611616".IsValidNationalCode();
```

پی نوشت:

لطفاً بخش نظرات خوانندگان را نیز دنبال کنید.

نظرات خوانندگان

نویسنده: علیرضا
تاریخ: ۱۷:۳۵ ۱۳۹۲/۰۱/۰۶

مشخصا برای اینکه متد قابل اطمینان باشد نباید از متد Parse استفاده کرد. به جای آن از TryParse استفاده کنید تا از وقوع exception جلوگیری شود

نویسنده: وحید نصیری
تاریخ: ۱۷:۴۰ ۱۳۹۲/۰۱/۰۶

اون وقت اگر در این بین به مشکل برخورد چطور؟ استثناء اصلا چیز بدی نیست؛ کرش بسیار پدیده مطلوبی است! چون نشان وجود مشکل در سیستم است. بجای اینکار بهتر است در همان بدو امر بررسی شود که رشته دریافتی عدد است یا خیر. چون طول رشته زیاد است می‌شود از [مثلا Regex](#) استفاده کرد:

```
public static bool IsItNumber(this string inputvalue)
{
    var isnumber = new Regex("[^0-9]");
    return !isnumber.IsMatch(inputvalue);
}
```

نویسنده: علیرضا اسم‌رام
تاریخ: ۱۷:۵۰ ۱۳۹۲/۰۱/۰۶

با تشکر از توجه شما،
یک متد الحاقی با عنوان IsItNumber به پروژه اضافه شد. همچنین متد IsValidNationalCode اصلاح شد.

```
public static bool IsValidNationalCode(this string nationalcode, out int lastNumber)
{
    lastNumber = -1;
    if (!nationalcode.IsItNumber()) return false;
    var array = nationalcode.ToCharArray();
    if (array.Length != 10) return false;
    var j = 10;
    var sum = 0;
    for (var i = 0; i < array.Length - 1; i++)
    {
        sum += Int32.Parse(array[i].ToString(CultureInfo.InvariantCulture)) * j;
        j--;
    }
    var div = sum / 11;
    var r = div * 11;
    var diff = Math.Abs(sum - r);

    if (diff <= 2)
    {
        lastNumber = diff;
        return diff == Int32.Parse(array[9].ToString(CultureInfo.InvariantCulture));
    }
    var temp = Math.Abs(diff - 11);
    lastNumber = temp;
    return temp == Int32.Parse(array[9].ToString(CultureInfo.InvariantCulture));
}
```

نویسنده: صابر فتح الهی
تاریخ: ۱۹:۱۲ ۱۳۹۲/۰۱/۰۶

سلام برای بررسی کد ملی من از یک الگوریتم دیگه استفاده کردم توی این [پست](#)

نویسنده: _mehdi
تاریخ: ۵:۲۷ ۱۳۹۲/۰۱/۰۷

سلام

من متوجه دلیل استفاده از متدهای الحاقی برای بررسی صحت کد ملی (جایی دیگه هم بررسی صحت ایمیل را دیده بودم) نمیشم . چون به نظر من البته اگه درست باشه با فلسفه وجودی متدهای الحاقی جور در نیامد . صرف اینکه کد ملی را از نوع string میگیریم بررسی صحت اون قابلیت نیست که بخواهیم به دلیل بسته بودن کلاس string به اون اضافه کنیم . چرا که ارزیابی صحت چیزی که منطق حاکم بر اون هیچ ربطی به کلاس string نداره را نمیتوان بهش افزود . بهتر نیست این نوع بررسیها به یه کلاس یوتیلیتی منتقل بشه .

نویسنده: محسن
تاریخ: ۹:۴۷ ۱۳۹۲/۰۱/۰۷

این نوع کلاسها هم تماما کمکی هستند. به همان روش سابق نام کلاس نام متد هم قابل دسترسی اند. به علاوه، به این ترتیب، علاوه بر روش قدیمی فراخوانی، حالت فراخوانی fluent هم میسر می شود که از دیدگاه زبان انگلیسی خواندنش ساده تر است و روان تر.

نویسنده: مجید
تاریخ: ۱۲:۵۵ ۱۳۹۲/۰۱/۰۷

سلام

جالبه! فقط یه نکته میشه به جای

```
var div = sum / 11;  
var r = div * 11;  
var diff = Math.Abs(sum - r);
```

نوشت :

```
var diff = sum % 11;
```

نویسنده: علیرضا اسمرام
تاریخ: ۱۳:۲۹ ۱۳۹۲/۰۱/۰۷

سلام. بله دو دستور فوق معادل یکدیگرند. من قبل از انتشار نسخه نهایی با مقادیر r و div کارهای دیگری انجام داده بودم که با توجه به حذف آن موارد می توان باقیمانده تقسیم sum بر 11 را تنها نگه داشت.

نویسنده: RezaNBT
تاریخ: ۱:۵۰ ۱۳۹۲/۰۱/۱۳

با سلام الگوریتم طراحی شده با reminder کمتر یا مساوی 2 مقدار غلط بر میگردد. بعنوان مثال مقدار محاسبه شده برای کد 0010350829 که معتبر می باشد برابر با 2 می باشد که باید از دوره تناوب کنترل کننده ی نقلی یعنی عدد 11 کم شود. یعنی :

```
if (diff <= 2)  
{  
    lastNumber = diff;  
    return diff ==(11- Int32.Parse(array[9].ToString(CultureInfo.InvariantCulture)));  
}
```

به نظر بنده نیازی به چک کردن و شاخه ای کردن کد نمی باشد و این بلوک شرطی می بایستی حذف گردد و در هر حالتی 11 از رقم نقلی کم شود و با مقدار حساب شده مقایسه شود.

از طرفی چون این فرمول نقلی به ازای هر عدد ده رقمی دارای تکرار فقط یک رقم خروجی صحیح دارد می بایستی این مورد هم کنترل گردد. یعنی در ابتدای تابع الحاقی باید داشته باشیم:

```
var allDigitEqual = new[] { "0000000000", "1111111111", "2222222222", "3333333333", "4444444444",  
"5555555555", "6666666666", "7777777777", "8888888888", "9999999999" };  
if (allDigitEqual.Contains(nationalcode)) return false;
```

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۲/۰۱/۱۳ ۳:۵۲

البته به جای این کد می توان [از این هم](#) استفاده کرد که به نظر خواناتر هست.

نویسنده: مجید
تاریخ: ۱۳۹۲/۰۱/۱۳ ۱۲:۲۲

شما اطمینان دارید که کد ملی 0010350829 معتبر است؟

نویسنده: علیرضا اسمرام
تاریخ: ۱۳۹۲/۰۱/۱۳ ۱۲:۳۳

با تشکر از شما متد فوق بصورت زیر اصلاح شد و کلاس مربوطه بروزرسانی شده است:

```
/// <summary>  
/// Validate IR National Code  
/// </summary>  
/// <param name="nationalcode">National Code</param>  
/// <param name="lastNumber">Last Number Of National Code</param>  
/// <returns></returns>  
public static bool IsValidNationalCode(this string nationalcode, out int lastNumber)  
{  
    lastNumber = -1;  
    if (!nationalcode.IsItNumber()) return false;  
    var invalid = new[]  
    {  
        "0000000000", "1111111111", "2222222222", "3333333333",  
        "4444444444", "5555555555",  
        "6666666666", "7777777777", "8888888888", "9999999999"  
    };  
    if (invalid.Contains(nationalcode)) return false;  
    var array = nationalcode.ToCharArray();  
    if (array.Length != 10) return false;  
    var j = 10;  
    var sum = 0;  
    for (var i = 0; i < array.Length - 1; i++)  
    {  
        sum += Int32.Parse(array[i].ToString(CultureInfo.InvariantCulture)) * j;  
        j--;  
    }  
    var diff = sum % 11;  
    if (diff < 2)  
    {  
        lastNumber = diff;  
        return diff == Int32.Parse(array[9].ToString(CultureInfo.InvariantCulture));  
    }  
    var temp = Math.Abs(diff - 11);  
    lastNumber = temp;  
    return temp == Int32.Parse(array[9].ToString(CultureInfo.InvariantCulture));  
}
```

}

البته من فرض کردم که کد ملی 0010350829 معتبر و کد 0010350822 نامعتبر است. آیا شما مطمئن هستید که کد 0010350829 معتبر است؟

نویسنده: صابر فتح الهی
تاریخ: ۸:۴۸ ۱۳۹۲/۰۱/۱۴

سلام

مهندس [این تکه کد](#) توسط آقای [ابراهیم بیاگوی](#) بهبود داده شده و به نظر شخصی من بهتره.

نویسنده: RezaNBT
تاریخ: ۲۰:۴۶ ۱۳۹۲/۰۱/۱۴

بله از صحت کد ملی 0010350829 اطمینان کامل دارم. شما هم می‌توانید محض اطمینان با الگوریتم‌های دیگر تست کنید.

نویسنده: sh-a
تاریخ: ۲۱:۲۷ ۱۳۹۲/۰۱/۱۴

با سلام

من از الگوریتم زیر استفاده میکنم گفتم اینجا بنویسم شاید به کار بیاد

```
Private Function ValidateNcode(ByVal NC As Int64) As Boolean
    If IsNumeric(NC) = False Or NC.ToString.Length < 10 Then
        Return False
    End If
    Dim vnc As String = NC.ToString
    Dim pos As Integer = 10
    Dim sum As Integer = 0

    For i = 0 To 8
        sum += Integer.Parse(vnc.ToCharArray()(i).ToString()) * pos
        pos = pos - 1
    Next i

    Dim remind As Integer = sum Mod 11
    Dim controlNum As Integer = Integer.Parse(vnc.ToCharArray()(9))

    If remind <= 2 Then
        If controlNum = remind Then
            Return True
        Else
            Return False
        End If
    Else
        If (11 - remind) = controlNum Then
            Return True
        Else
            Return False
        End If
    End If
End Function
```

```
private bool ValidateNcode(Int64 NC)
{
    if (IsNumeric(NC) == false | NC.ToString.Length < 10) {
        return false;
    }
    string vnc = NC.ToString;
    int pos = 10;
    int sum = 0;

    for (i = 0; i <= 8; i++) {
        sum += int.Parse(vnc.ToCharArray()(i).ToString()) * pos;
        pos = pos - 1;
    }
}
```

```
}  
  
int remind = sum % 11;  
int controlNum = int.Parse(vnc.ToCharArray()(9));  
  
if (remind <= 2) {  
    if (controlNum == remind) {  
        return true;  
    } else {  
        return false;  
    }  
} else {  
    if ((11 - remind) == controlNum) {  
        return true;  
    } else {  
        return false;  
    }  
}  
}
```

نویسنده: f.beigirad
تاریخ: ۱۴۰۱/۰۶/۱۷

اتباع خارجی هم کد شناسایی دارن.
آیا میشه متدیم برای تشخیص صحت این کد پیشنهاد بدین.

نویسنده: افشین
تاریخ: ۱۳۹۲/۰۸/۲۷ ۷:۵۹

یک حالت جدید پیدا کردم که مطمئنا خود ثبت احوال هم نمی‌دونه! هر عدد ده رقمی که به فرم قریبه وارد بشه، صحیح فرض میشه!
مثلا a00000000 یا 1234554321 یا 1346886431 همگی صحیح فرض میشن حال آنکه اشتباه هستن!

نویسنده: سعید
تاریخ: ۱۳۹۲/۰۸/۲۷ ۱۷:۵۱

من براتون یه کد ملی میارم که کل این الگوریتم رو نقض کنه. کد ملی 1111111111 متعلق به آقای در شهر رضا هست! من چون به تعداد بالایی کد ملی دسترسی دارم، بررسی کردم و الان مطمئن هستم ثبت احوال در سالهای ابتدایی تخصیص کد ملی، خطا داشته.

نویسنده: وحید
تاریخ: ۱۳۹۳/۰۴/۲۲ ۱:۳۹

سلام به دوستان عزیز، این الگوریتم برای حدود 10 درصد از کد ملی‌ها درست عمل نمی‌کند به عنوان نمونه کدهای زیر در الگوریتم فوق فاقد اعتبار هستند (از صحت کد ملی اطمینان دارم)

1381647899
0922351873
4122587241
4312281949
4220196526
6219937997
0077910345
4569973963
2133288158
2003987765
2758531405
4251235692

2156345454

1222365215

5412568954

از این دست می‌توانم تعداد بسیاری را مثال بزنم

افزودن یک DataType جدید برای نگهداری تاریخ خورشیدی - 1

حامد قنادی

۲۳:۳۰ ۱۳۹۲/۰۲/۰۹

www.dotnettips.info

SQL Server, SQL Server 2012, C#.NET, Persian, CLR

عنوان:

نویسنده:

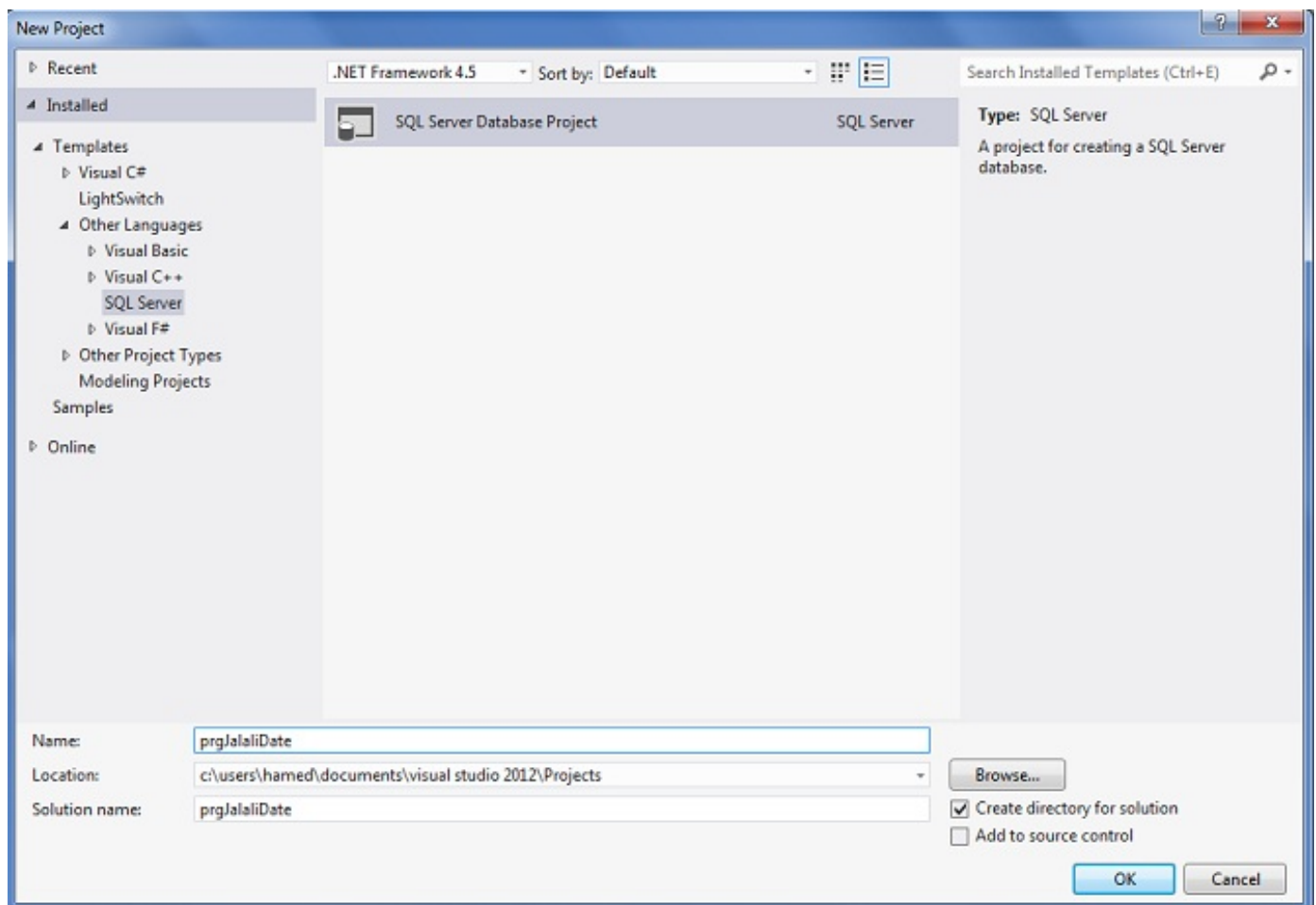
تاریخ:

آدرس:

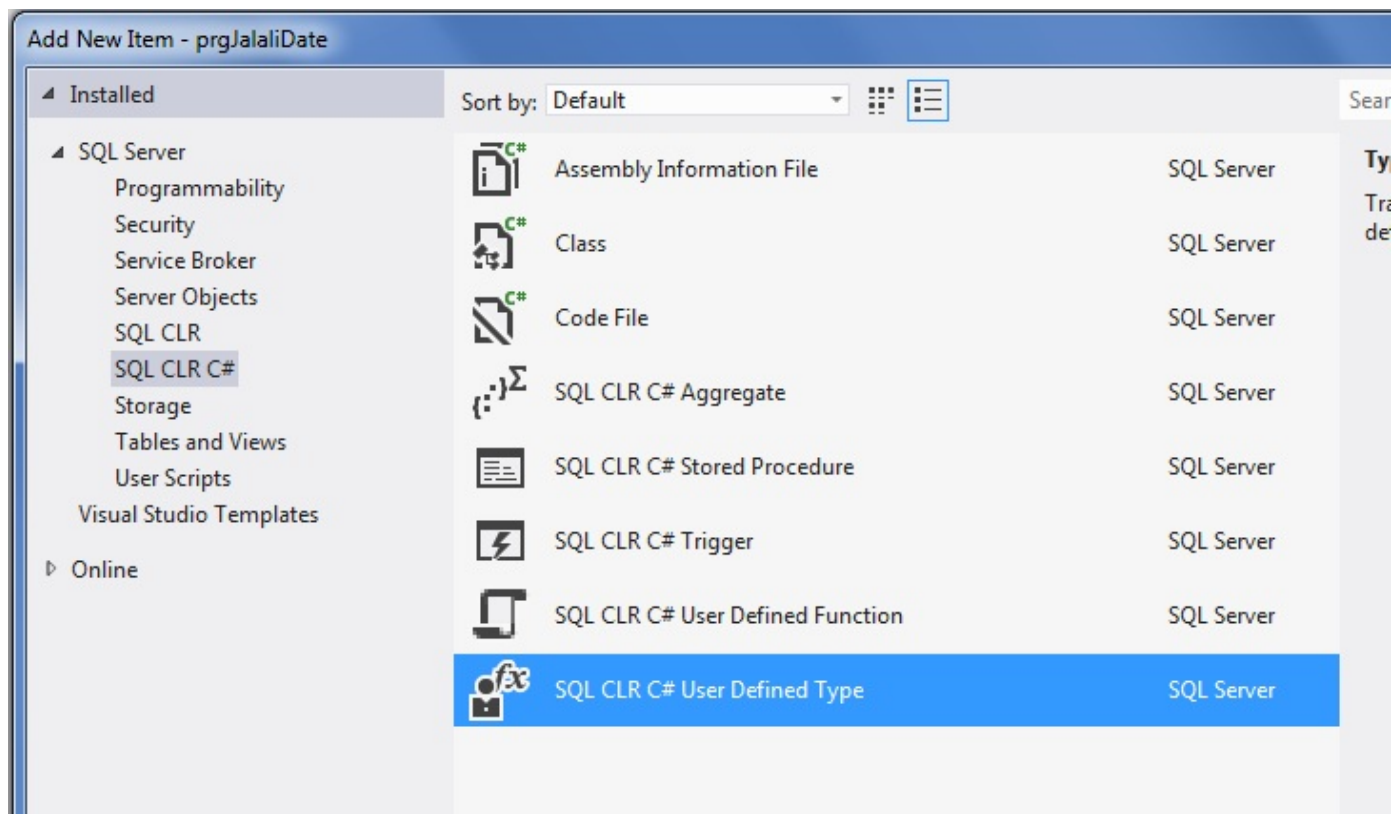
گروه‌ها:

ثبت و نگهداری تاریخ خورشیدی در SQL Server از دیرباز یکی از نگرانی‌های برنامه‌نویسان و طراحان پایگاه داده‌ها بوده است. در این نوشتار، راه‌کار تعریف یک DataType در SQL Server 2012 به روش CLR آموزش داده خواهد شد.

در ویتوال استودیو یک پروژه‌ی جدید از نوع SQL Server Database Project به شکل زیر ایجاد کنید:



نام پروژه را به یاد تقویم خیام، prgJalaliDate می‌گذارم. در Solution Explorer روی نام پروژه راست‌کلیک کرده، سپس روی Add New Item کلیک کنید. در پنجره‌ی باز شده مطابق شکل SQL CLR C# User Defined Type را برگزینید؛ سپس نام JalaliDateType را برای آن انتخاب کنید.



متن موجود در صفحه‌ی بازشده را کاملاً حذف کرده و با کد زیر جای‌گزین کنید.
 (در کد زیر تمامی توابع لازم برای مقداردهی به سال، ماه، روز، ساعت، دقیقه و ثانیه و البته گرفتن مقدار از آن‌ها، تبدیل تاریخ خورشیدی به میلادی، گرفتن تاریخ به تنهایی، گرفتن زمان به تنهایی، افزایش یا کاهش زمان برپایه‌ی یکی از متغیرهای زمان و بررسی و اعتبارسنجی انواع بخش‌های زمان گنجانده شده است. در صورت پرسش یا پیشنهاد روی هر کدام در قسمت نظرات، پیام خود را بنویسید.)

```
using System;
using System.Data.SqlTypes;
using Microsoft.SqlServer.Server;

[Serializable()]
[SqlUserDefinedType(Format.Native)]
public struct JalaliDate : INullable
{
    private Int16 m_Year;
    private byte m_Month;
    private byte m_Day;
    private byte m_Hour;
    private byte m_Minute;
    private byte m_Second;
    private bool is_Null;

    public Int16 Year
    {
        get
        {
            return (this.m_Year);
        }
        set
        {
            m_Year = value;
        }
    }

    public byte Month
    {
        get
        {
            return (this.m_Month);
        }
    }
}
```

```

    }
    set
    {
        m_Month = value;
    }
}

public byte Day
{
    get
    {
        return (this.m_Day);
    }
    set
    {
        m_Day = value;
    }
}

public byte Hour
{
    get
    {
        return (this.m_Hour);
    }
    set
    {
        m_Hour = value;
    }
}

public byte Minute
{
    get
    {
        return (this.m_Minute);
    }
    set
    {
        m_Minute = value;
    }
}

public byte Second
{
    get
    {
        return (this.m_Second);
    }
    set
    {
        m_Second = value;
    }
}

public bool IsNull
{
    get
    {
        return is_Null;
    }
}

public static JalaliDate Null
{
    get
    {
        JalaliDate jl = new JalaliDate();
        jl.is_Null = true;
        return (jl);
    }
}

public override string ToString()
{
    if (this.IsNull)
    {
        return "NULL";
    }
    else

```

```

    {
        return this.m_Year.ToString("D4") + "/" + this.m_Month.ToString("D2") + "/" +
this.m_Day.ToString("D2") + " " + this.Hour.ToString("D2") + ":" + this.Minute.ToString("D2") + ":" +
this.Second.ToString("D2");
    }
}

public static JalaliDate Parse(SqlString s)
{
    if (s.IsNull)
    {
        return Null;
    }

    System.Globalization.PersianCalendar pers = new System.Globalization.PersianCalendar();
    string str = Convert.ToString(s);
    string[] JDate = str.Split(' ')[0].Split('/');

    JalaliDate jl = new JalaliDate();

    jl.Year = Convert.ToInt16(JDate[0]);
    byte MonthsInYear = (byte)pers.GetMonthsInYear(jl.Year);
    jl.Month = (byte.Parse(JDate[1]) <= MonthsInYear ? (byte.Parse(JDate[1]) > 0 ?
byte.Parse(JDate[1]) : (byte)1) : MonthsInYear);
    byte DaysInMonth = (byte)pers.GetDaysInMonth(jl.Year, jl.Month); ;
    jl.Day = (byte.Parse(JDate[2]) <= DaysInMonth ? (byte.Parse(JDate[2]) > 0 ?
byte.Parse(JDate[2]) : (byte)1) : DaysInMonth);
    if (str.Split(' ').Length > 1)
    {
        string[] JTime = str.Split(' ')[1].Split(':');
        jl.Hour = (JTime.Length >= 1 ? (byte.Parse(JTime[0]) < 23 && byte.Parse(JTime[0]) >=
(byte)0 ? byte.Parse(JTime[0]) : (byte)0) : (byte)0);
        jl.Minute = (JTime.Length >= 2 ? (byte.Parse(JTime[1]) < 59 && byte.Parse(JTime[1]) >=
(byte)0 ? byte.Parse(JTime[1]) : (byte)0) : (byte)0);
        jl.Second = (JTime.Length >= 3 ? (byte.Parse(JTime[2]) < 59 && byte.Parse(JTime[2]) >=
(byte)0 ? byte.Parse(JTime[2]) : (byte)0) : (byte)0);
    }
    else { jl.Hour = 0; jl.Minute = 0; jl.Second = 0; }

    return (jl);
}

public SqlString GetDate()
{
    return this.m_Year.ToString("D4") + "/" + this.m_Month.ToString("D2") + "/" +
this.m_Day.ToString("D2");
}

public SqlString GetTime()
{
    return this.Hour.ToString("D2") + ":" + this.Minute.ToString("D2") + ":" +
this.Second.ToString("D2");
}

public SqlDateTime ToGregorianTime()
{
    System.Globalization.PersianCalendar pers = new System.Globalization.PersianCalendar();
    return SqlDateTime.Parse(pers.ToDateTime(this.Year, this.Month, this.Day, this.Hour,
this.Minute, this.Second, 0).ToString());
}

public SqlString JalaliDateAdd(SqlString interval, int increment)
{
    System.Globalization.PersianCalendar pers = new System.Globalization.PersianCalendar();
    DateTime dt = pers.ToDateTime(this.Year, this.Month, this.Day, this.Hour, this.Minute,
this.Second, 0);
    string CInterval = interval.ToString();
    bool isConvert = true;
    switch (CInterval)
    {
        case "Year":
            dt = pers.AddYears(dt, increment);
            break;
        case "Month":
            dt = pers.AddMonths(dt, increment);
            break;
        case "Day":
            dt = pers.AddDays(dt, increment);
            break;
        case "Hour":

```

```

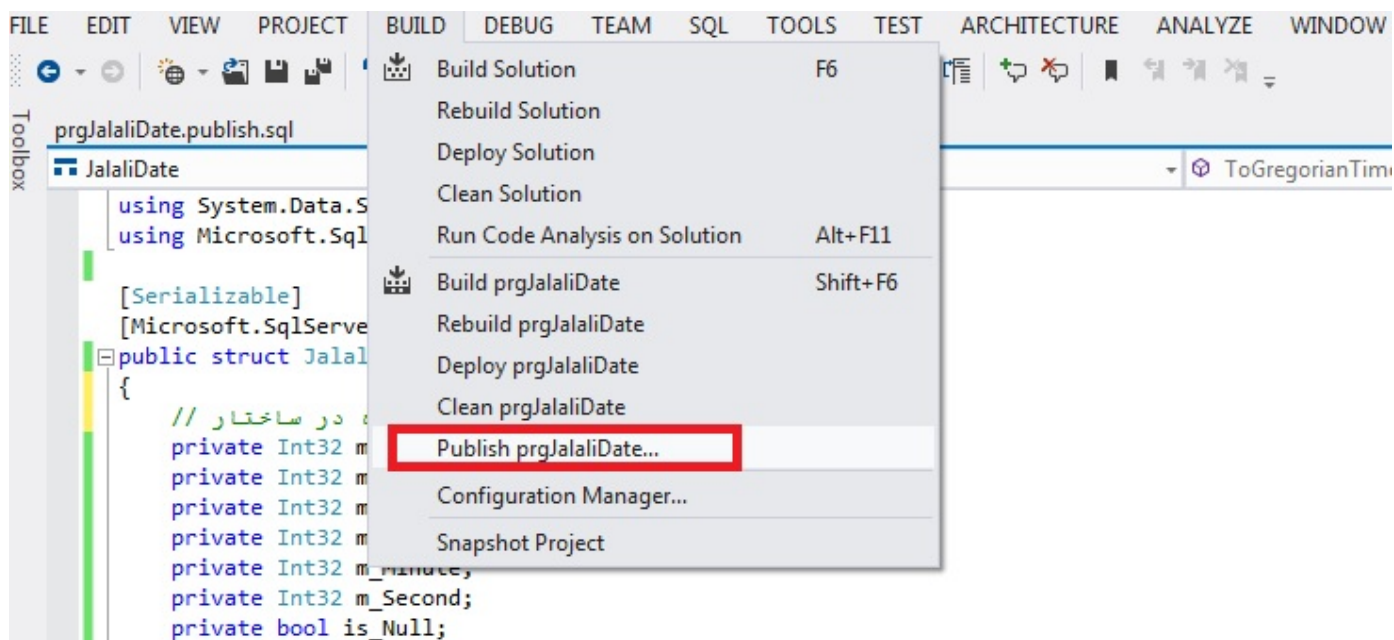
        dt = pers.AddHours(dt, increment);
        break;
    case "Minute":
        dt = pers.AddMinutes(dt, increment);
        break;
    case "Second":
        dt = pers.AddSeconds(dt, increment);
        break;
    default:
        isConvert = false;
        break;
}

if (isConvert == true)
{
    this.Year = (Int16)pers.GetYear(dt);
    this.Month = (byte)pers.GetMonth(dt);
    this.Day = (byte)pers.GetDayOfMonth(dt);
    this.Hour = (byte)pers.GetHour(dt);
    this.Minute = (byte)pers.GetMinute(dt);
    this.Second = (byte)pers.GetSecond(dt);
}

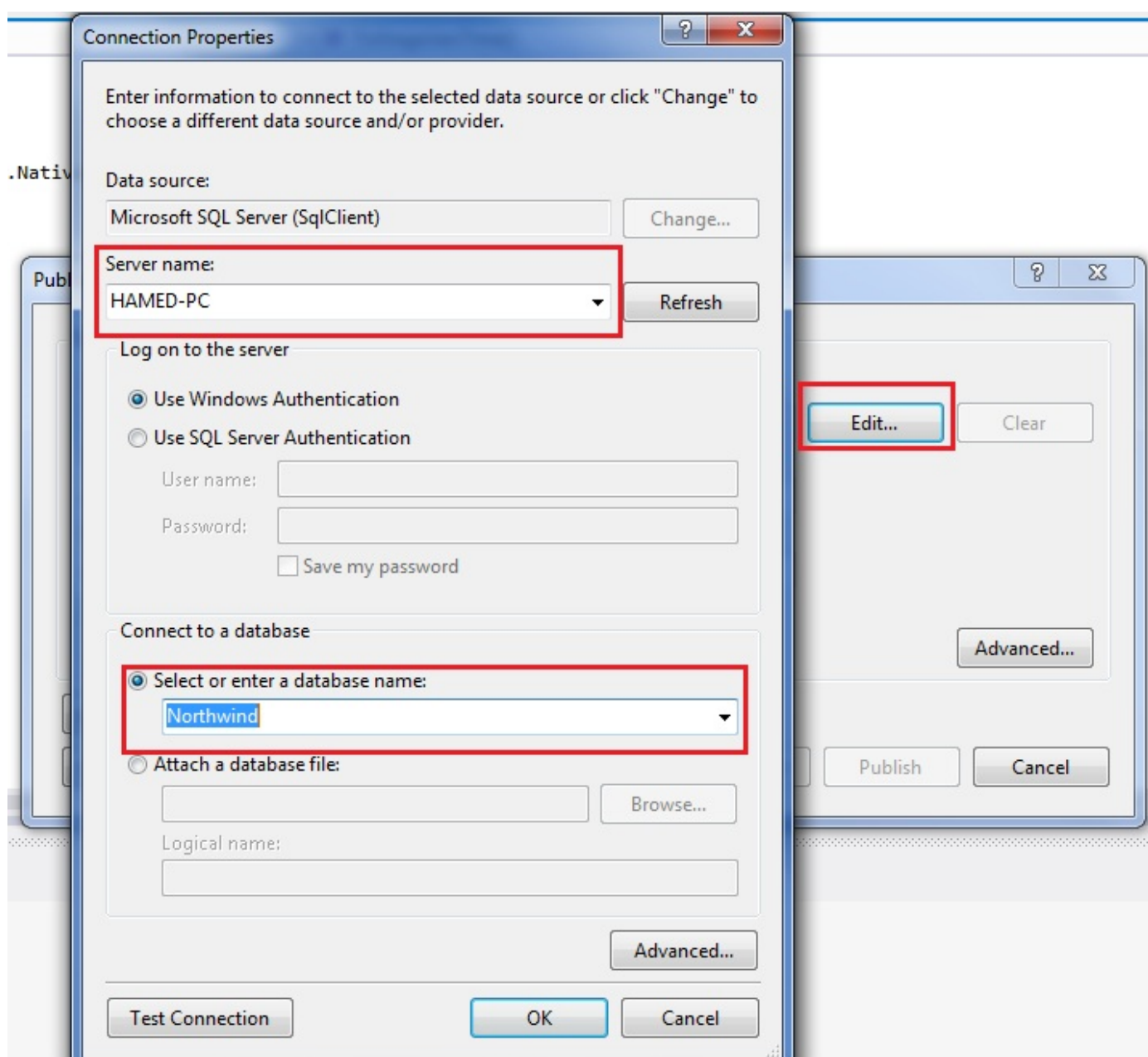
return this.m_Year.ToString("D4") + "/" + this.m_Month.ToString("D2") + "/" +
this.m_Day.ToString("D2") + " " + this.Hour.ToString("D2") + ":" + this.Minute.ToString("D2") + ":" +
this.Second.ToString("D2");
}
}

```

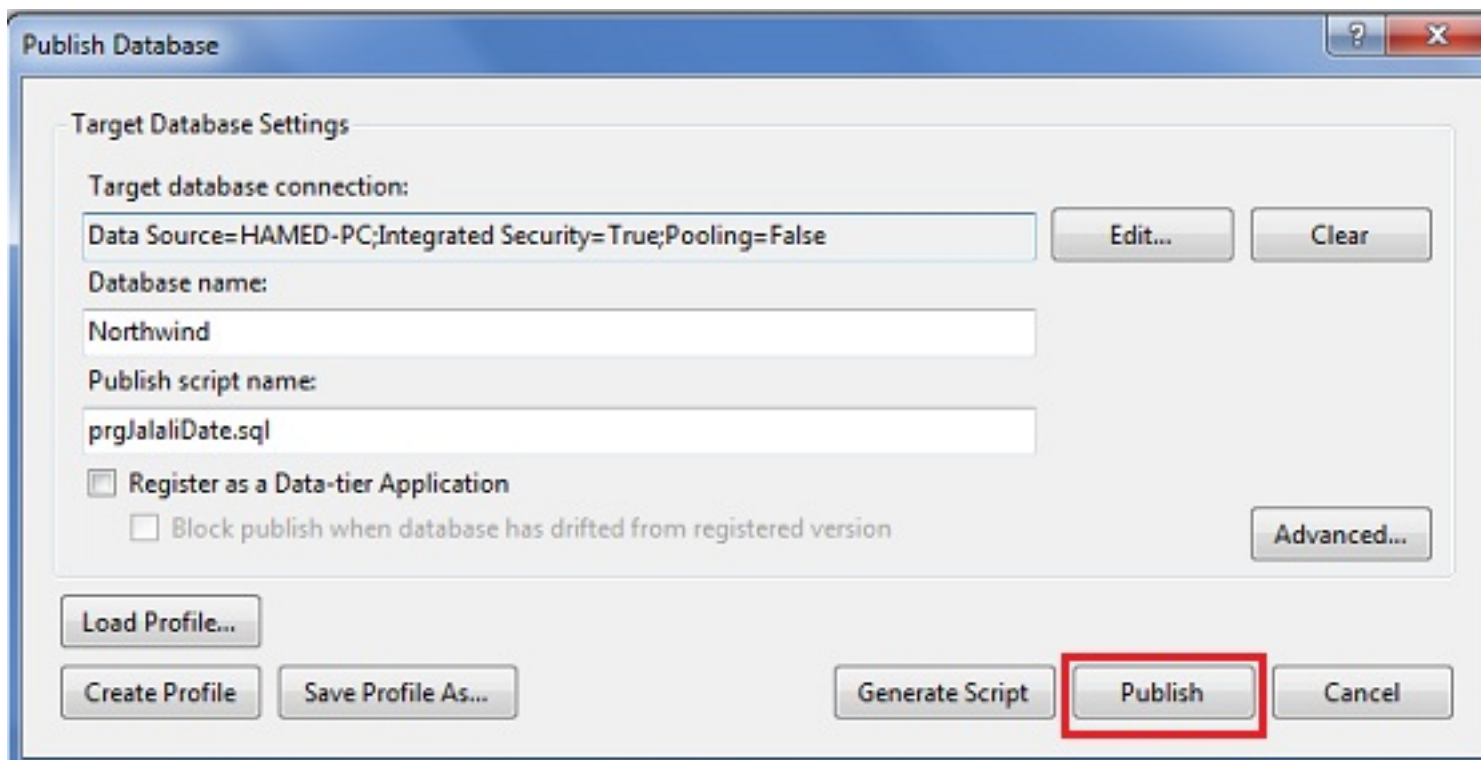
از منوهای بالا روی منوی Build و سپس گزینه‌ی Publish prgJalaliDate کلیک کنید:



در پنجره‌ی بازشده روی دکمه‌ی Edit کلیک کنید سپس تنظیمات مربوط به اتصال به پایگاه داده را انجام دهید.



روی دکمه‌ی OK کلیک کنید و سپس در پنجره‌ی اولیه، روی دکمه‌ی Publish کلیک کنید:



به همین سادگی، DataType مربوطه در SQL Server 2012 ساخته می‌شود. خبر خوش این که شما می‌توانید با راست‌کلیک روی نام پروژه و انتخاب گزینه‌ی Properties در قسمت Project Setting تنظیمات مربوط به نگارش SQL Server را انجام دهید. (از نگارش 2005 به بعد در VS 2012 پشتیبانی می‌شود).

اکنون زمان آن رسیده است که DataType ایجادشده را در SQL Server 2012 بیازماییم. SQL Server را باز کنید و دستور زیر را در آن اجرا کنید.

```
USE Northwind
GO
CREATE TABLE dbo.TestTable
(
  Id int NOT NULL IDENTITY (1, 1),
  TestDate dbo.JalaliDate NULL
) ON [PRIMARY]
GO
```

همین‌طور که مشاهده می‌کنید؛ امکان به‌کارگیری DataType تعریف‌شده وجود دارد. اکنون چند رکورد درون این جدول درج می‌کنیم:

```
Insert into TestTable (TestDate) Values ('1392/02/09'),('1392/02/09 22:40'),('1392/12/30 22:40')
```

پس از اجرای این دستور خطای زیر در پایین صفحه‌ی SQL Server نمایان می‌شود:

Messages
Msg 6263, Level 16, State 1, Line 1
Execution of user code in the .NET Framework is disabled. Enable "clr enabled" configuration option.

این خطا به این خاطر است که CLR را در SQL Server فعال نکرده ایم. جهت فعال کردن CLR دستور زیر را اجرا کنید:

```
sp_configure 'clr enabled', 1
Reconfigure
```

بار دیگر دستور درج را اجرا می‌کنیم:

```
Insert into TestTable (TestDate) Values ('1392/02/09'),('1392/02/09 22:40'),('1392/12/30 22:40')
```

	Id	TestDate
▶	1	1392/02/09 00:00:00
	2	1392/02/09 22:40:00
	3	1392/12/29 22:40:00
*	NULL	NULL

ملاحظه می‌کنید که داده‌ها در جدول مربوطه ذخیره شده است. در رکورد نخست چون ساعت، دقیقه و ثانیه تعریف نشده است؛ به طور هوشمند صفر درج شده است. در رکورد دوم، ساعت و دقیقه مقدار دارد ولی ثانیه صفر ثبت شده است. و در رکورد سوم چون سال 1392 کیبسه نیست؛ به صورت هوشمند آخرین روز ماه به جای روز ثبت شده است. هرچند می‌توان با دست‌کاری در توابع سی‌شارپ، این قوانین را عوض کرد.

اکنون زمان آن رسیده است که توسط یک پرس‌وجو، تمامی توابعی که در سی‌شارپ برای این نوع داده نوشتیم، بیازماییم. پرس‌وجوی زیر را اجرا کنید:

```
Select TestDate.ToString() as JalaliDateTime,
       TestDate.GetDate() as JalaliDate, TestDate.GetTime() as JalaliTime,
       TestDate.ToGregorianTime() as GregorianTime,
       TestDate.JalaliDateAdd('Day',1) JalaliTomorrow,
       TestDate.Month as JalaliMonth from TestTable
```

خروجی این پرس‌وجو به شکل زیر خواهد بود:

	JalaliDateTime	JalaliDate	JalaliTime	GregorianTime	JalaliTomorrow	JalaliMonth
1	1392/02/09 00:00:00	1392/02/09	00:00:00	2013-04-29 00:00:00.000	1392/02/10 00:00:00	2
2	1392/02/09 22:40:00	1392/02/09	22:40:00	2013-04-29 22:40:00.000	1392/02/10 22:40:00	2
3	1392/12/29 22:40:00	1392/12/29	22:40:00	2014-03-20 22:40:00.000	1393/01/01 22:40:00	12

البته درباره‌ی ستون پنجم و ششم شما می‌توانید روی تمامی اجزای تاریخ افزایش و کاهش داشته باشید و همچنین می‌توانید با تابع مربوطه هر کدام از اجزای زمان را جداگانه به دست بیاورید که در این مثال عدد ماه نشان داده شده است.

نیازی به گفتن نیست که می‌توانید به سادگی از توابع مربوط به DateTime در SQL Server بهره ببرید. برای مثال برای به دست

آوردن فاصله‌ی میان دو روز از پرس‌وجوی زیر استفاده کنید:

```
Declare @a JalaliDate = '1392/02/07 00:00:00'  
Declare @b JalaliDate = '1392/02/05 00:00:00'  
  
SELECT DATEDIFF("DAY",@b.ToGregorianTime(),@a.ToGregorianTime()) AS DiffDate
```

شاد و پیروز باشید.

نظرات خوانندگان

نویسنده: ali

تاریخ: ۱۳۹۲/۰۲/۱۰ ۰:۵

Nice article

Thanks

نویسنده: سام ناصری

تاریخ: ۱۳۹۲/۰۲/۱۰ ۴:۲۲

بسیار خوب.

فقط من تو مقاله شما دلیلی برای اینکه چرا زمان و تاریخ را میخواهی به این صورت ذخیره کنی متوجه نشدم؟ چرا به همان شکل استانداردش ذخیره نکنیم؟
دیگر اینکه نوع داده جدید به چه شکل در دیتابیس ذخیره میشود. Sql Server از کجا میداند که باید چگونه لیترالها را پارس کند(چگونه متود Parse برگزیده میشود)؟ عملگرهای مقایسه چگونه کار خواهند کرد؟

نویسنده: حامد قنادی

تاریخ: ۱۳۹۲/۰۲/۱۰ ۷:۸

با درود

به پرسش‌های شما در بخش دوم پاسخ خواهم داد.

نویسنده: قاسم

تاریخ: ۱۳۹۲/۰۲/۱۰ ۹:۴۲

سلام، میشه مراحل انجام کار توی VS2010 هم بنویسید، فقط مراحل کار. خیلی ممنون میشم

نویسنده: فرشید علی اکبری

تاریخ: ۱۳۹۲/۰۲/۱۰ ۹:۵۵

سلام

کدهای شمارو درست همونطوریکه گفتین کپی کردم ولی موقعی که Publish رو میزنم پیغام زیر رو میده :

Publish cannot begin until your project(s) build successfully

و پنجره publish ظاهر نمی‌شه و هیچ جایی هم برای تعریف کانکشن وجود نداره... مشکل از کجاست؟

دوم اینکه clr رو هربار که خواستیم روی یک دیتابیس جدید این نوع رو تعریف کنیم باید فعال کنیم... مثلاً توی ایجاد سال مالی جدید که سیستم یک دیتابیس خام ایجاد میکنه تا مانده حسابها رو بهش انتقال بدیم قبلش باید دستور فعال سازی clr رو هم مجدداً بدیم یا فقط توی زمان طراحی کفایت میکنه ؟

سوم اینکه : در مورد استفاده اون توی EF Code First هم اگه نکته‌ی خاصی وجود داره محبت کنین ممنون میشم.
شاد و پیروز باشید.

نویسنده: حامد قنادی

تاریخ: ۱۳۹۲/۰۲/۱۰ ۱۱:۴

با درود

New Project -> Database -> SQL Server -> Visual C# SQL CLR Database Project

- تنظیمات اتصال به پایگاه داده ها

- انتخاب دکمه Yes

Add New Item -> User Defined Type -

- کپی کدها

- استفاده از Deploy در منوی Build یا استفاده از روشی که در بخش دوم نوشتار آموزش داده خواهد شد.

نویسنده: حامد قنادی

تاریخ: ۱۱:۲۶ ۱۳۹۲/۰۲/۱۰

با درود

- 1- لطفاً پروژه را پیش از Publish یک بار Rebuild کنید. احتمالاً به خاطر یک ارور خاص Publish نمی‌شود. اگر به راحتی Rebuild شده ولی باز هم Publish کار نمی‌کند؛ می‌توانیم با روش دیگری که در بخش دوم آموزش می‌دهم DLL را به SQL معرفی کنیم.
- 2- کافی است یک بار پس ساخت Database این پرس‌وجو را اجرا کنید.
- 3- تست نکردم ولی به نظر می‌رسد این نوع داده از سمت EF شبیه به Hierarchy باشد.

نویسنده: ش.د

تاریخ: ۱۴:۳۵ ۱۳۹۲/۰۲/۱۰

آیا فقط در sql2012 قابل اجرا می‌باشد؟

نویسنده: محسن خان

تاریخ: ۱۴:۵۴ ۱۳۹۲/۰۲/۱۰

در متن نوشتن از نگارش 2005 به بعد اس کیوال سرور این قابلیت استفاده از افزونه‌های CLR اضافه شدن.

نویسنده: سید امیر سجادی

تاریخ: ۱۸:۳۸ ۱۳۹۲/۰۲/۱۰

با تشکر از مطلب مفیدتون.

چند تا سوال برام پیش اومده.

اول اینکه آیا به صورت یک DLL به بانک اضافه میشه؟

دوم اینکه اگه از بانک بک آپ بگیریم و جایی دیگه خواستیم اون رو ریستور کنیم چی میشه؟

آپلود بانک روی هاست (بک آپ یا اتچ) ؟

نویسنده: امیر بختیاری

تاریخ: ۹:۴۳ ۱۳۹۲/۰۲/۱۱

به صورت اسمبلی به بانک اضافه میشه

وقتی بک آپ بگیرید و ریستور کنید همراه بک آپ این اسمبلی هم انتقال داده می‌شود

در اتچ هم به همین شکل

فقط زمانی که از این اسمبلی در توابع و پروسیجرها استفاده می‌کنید نمی‌تونید حذفش کنید و در صورت تغییرات باید اسمبلی را به روز کنید.

یک مورد دیگه که من زیاد تو هاست‌های شیر شده بهش برخوردم اینه که این امکان توشون فعال نیست و ادمین سرورها هم به سختی این امکان را فعال می‌کنند پس اگر خواستید از این امکان استفاده کنید ابتدا از فعال بودن آن مطمئن شوید

نویسنده: rahim

تاریخ: ۱۲:۱۴ ۱۳۹۳/۰۵/۱۱

با تشکر از مطلب مفیدتون

من زمانی که از این روش استفاده میکنم در هنگام درج رکورد جدید با پیغام خطای زیر مواجه میشم.

:"A .NET Framework error occurred during execution of user-defined routine or aggregate "JalaliDate

.System.OverflowException: Value was either too large or too small for an unsigned byte

تعامل MATLAB (متلب) با دات نت - قسمت اول

عنوان:

نویسنده: مسعود مشهدی

تاریخ: ۱۵:۰ ۱۳۹۲/۰۳/۱۴

آدرس: www.dotnettips.info

برچسب‌ها: NET, C#.NET, MATLAB.

متلب (MATLAB) یکی از پرکاربردترین نرم افزارهای محاسباتی در حوزه مهندسی بویژه برق، ریاضیات، مکانیک و ... می‌باشد. بدون شک تعامل نرم افزارهای مختلف با هم در جهت کاربردی‌تر کردن یک پروژه کمک بسزایی به کاربران نهایی می‌کند. قطعاً استفاده از علوم روز همچون شبکه‌های عصبی، منطق فازی و الگوریتم‌های تکاملی همچون ژنتیک بدون استفاده از متلب بسیار سخت و پیچیده خواهد بود. دستورات و تابع‌های (functions) آماده و ساده در متلب در جهت استفاده از این علوم تقریباً هر پژوهشگر و کاربری را ترغیب به استفاده از متلب می‌کند. طبقاً استفاده از کتابخانه‌های دانت در متلب کمک بسیاری به توسعه دهندگان این حوزه می‌کند.

در این سری از مطالب سعی بر بررسی این تعامل شده است.

بطور کلی دو نوع تعامل در این زمینه وجود دارد :

1- استفاده از اسمبلی‌های دات نت در متلب تحت عنوان MATLAB .NET Interface

2- استفاده از پکیج تابع‌های متلب در پروژه‌های مبتنی بر دات نت تحت عنوان MATLAB Builder NE

در مورد اول از دات نت فقط در پلت فورم ویندوز استفاده می‌شود. کلیه امکانات دات نت 2 را ساپورت میکند و با ورژن‌های 3 و 3.5 سازگار است اما با ورژن 4 تنها بعضی از امکانات در دسترس است و هنوز مورد تست کلی قرار نگرفته است. کلیه امکانات دات نت در C# در متلب بجز یک سری از موارد که در جدول زیر ذکر شده است در دسترس است.

Features Not Supported in MATLAB

Cannot use `ClassName.propertyname` syntax to set static properties. Use [NET.setStaticProperty](#) instead.

Unloading an assembly

Passing a structure array, sparse array, or complex number to a .NET property or method

Subclassing .NET classes from MATLAB

Accessing nonpublic class members

Displaying generic methods using `methods` or `methodsview` functions. For a workaround, see [Display .NET Generic Methods Using Reflection](#).

Creating an instance of a nested class. For a workaround, see [Working With Nested Classes](#).

Saving (serializing) .NET objects into a MAT-file

Creating .NET arrays with a specific lower bound

Concatenating multiple .NET objects into an array

Implementing interface methods

Hosting .NET controls in figure windows

Casting operations

Calling constructors with `ref` or `out` type arguments

Using `System.Console.WriteLine` to write text to the command window

Pointer type arguments, function pointers, `Dllimport` keyword

.NET remoting

Using the MATLAB: (colon) operator in a `foreach` iteration

Adding event listeners to .NET events defined in static classes

Handling .NET events with signatures that do not conform to the standard signature

Creating empty .NET objects

Creating .NET objects that do not belong to a namespace

به عنوان مثال از کلاس speech synthesizer دات نت 3 در متلب بصورت زیر استفاده می‌کنیم :

```
function Speak(text)
    NET.addAssembly('System.Speech');
    speak = System.Speech.Synthesis.SpeechSynthesizer;
    speak.Volume = 100;
```

```
Speak(speak,text);  
end
```

سپس برای رندر کردن یک متن به صوت دستور زیر را اجرا می‌کنیم :

```
Speak('You can use .NET Libraries in MATLAB');
```

در ارتباط با استفاده از توابع متلب در یک پروژه مبتنی بر دات نت در قسمت بعد توضیح داده خواهد شد.

منبع : Help متلب

نظرات خوانندگان

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۳/۱۶ ۰۵:۵۸

در این کدی که تهیه کردید، Speak داخل متد Speak [چطور فراخوانی شده](#) بدون ارجاع به یک شیء؟ (سطر آخر متد). بعد در متلب نیازی به new نیست؟ فقط فراخوانی NET.addAssembly باعث شناسایی System.Speech.Synthesis.SpeechSynthesizer میشه؟

نویسنده: مسعود مشهدی
تاریخ: ۱۳۹۲/۰۳/۱۶ ۹:۱۸

متلب به حروف کوچک و بزرگ حساس است. دقت کرده باشید speak داخل Speak است. در حقیقت Speak نام تابع است. در متلب تابع NET.addAssembly کار بارگذاری اسمبلی‌های یاد شده را دارد. خیر در متلب احتیاجی به new نیست. در کل برنامه نویسی در متلب کمی متفاوت با زبان‌های معمول (C# و ..) است.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۳/۱۶ ۱۰:۲۰

اینجا در سطر آخر متد نباید نوشته می‌شد speak.Speak ؟ یا اینکه این Speak نوشته شده [با دو پارامتر](#) ، منظور تابعی دیگر هست؟

نویسنده: مسعود مشهدی
تاریخ: ۱۳۹۲/۰۳/۱۶ ۱۲:۱۸

نه دوست عزیز همونطور که گفتم ساختار متلب متفاوت هست.
منظور از Speak همون تابع است.

کافیست تابع فوق را در یک M فایل (کدها در متلب با پسوند ام فایل ذخیره می‌شوند) ذخیره کنید. سپس در داخل متلب مسیر اجرای کد را به مسیر M فایل تغییر دهید و سپس دستور رندر کردن رو که بالا گفته شد در Command Window اجرا کنید. متن فوق به تابع Speak ارجاع داده می‌شود.

نویسنده: علی
تاریخ: ۱۳۹۲/۰۳/۲۳ ۲۲:۳۷

استفاده از Matlab در دات نت نیاز به نصب کلی خود Matlab داره؟

نویسنده: مسعود مشهدی
تاریخ: ۱۳۹۲/۰۳/۲۴ ۲۱:۳۰

در این مورد در قسمت بعد توضیح خواهم داد. بله احتیاج است. اگر تصمیم به نوشتن کدها و کامپایل کردن اونها دارید. خیر، اگر نویسنده کدها یک پکیج از پروژه متلب در اختیار شما قرار بدهد.

عنوان: دریافت مقدار از یک AnonymousType

نویسنده: مجتبی صحرایی

تاریخ: ۲۲:۳۵ ۱۳۹۲/۰۳/۱۸

آدرس: www.dotnettips.info

برچسب‌ها: Anonymous Type, C#.NET

AnonymousType ها به شما این امکان را میدهند که بدون ساخت Type جدیدی چندین پراپرتی رو در یک object قرار بدید و از اون استفاده کنید.

به مثال زیر توجه کنید:

```
var v = new { Amount = 108, Message = "Hello" };
Console.WriteLine(v.Amount + v.Message);
```

دقت داشته باشید که در این مثال Type ی بنام v از قبل ساخته نشده بود و فیلدهای Amount, Message هم وجود ندارند اما با استفاده از AnonymousType ها این امکان فراهم شده تا بتونید بدون ساخت Type، رفتار Type ها رو شبیه سازی کنید.

بطور مثال فرض کنیم می‌خواهیم نام و شناسه پرسنل رو در یک ComboBox نمایش دهیم. برای این منظور نیازی نیست که تمام فیلدهای مربوط به پرسنل را بازایی نموده و تنها نام و شناسه پرسنل را نمایش دهیم و همانطور که بسیاری از شما می‌دانید تنها نام و شناسه پرسنل را بازایی می‌کنیم که این مورد را با استفاده از AnonymousType انجام می‌دهیم <=

```
var business = new Customers();
var modelsCollection = business.GetAll(w => w.MCode);
cmbCustomerName.DataSource = modelsCollection.Select(w => new { w.CustomerName, w.Code }).ToList();
```

چنانچه از modelsCollection فیلد و یا فیلدی را انتخاب نمی‌کردیم و از تمام فیلدها استفاده می‌کردیم برای دریافت آیتم انتخاب شده مشکلی نداشتیم و با یک Cast ساده به Model Type مورد نظر، می‌توانستیم مقدار شناسه پرسنل انتخابی را بدست آوریم اما با این قطعه کد چنانچه بخواهیم عمل Casting را برای Model Type مورد نظر انجام دهیم، با خطا مواجه خواهیم شد چون Type جدید از نوع AnonymousType بوده و شامل دو فیلد CustomerName, Code می‌باشد.

سوالاتی که پیش خواهد آمد:

1- مقدار Code رو چطور استخراج کنیم؟

2- AnonymousType یک Type موجود نیست که بتوان آیتم انتخابی را به آن Cast نموده و مقدار Code را دریافت نمود!

.

.

.

با استفاده از Reflection قادر خواهید بود مقدار فیلد مورد نظر خود را از یک AnonymousType استخراج کنید. برای این منظور تابع زیر رو در نظر بگیرید <=

```
public static T GetValueFromAnonymousType<T>(object dataitem, string itemkey)
{
    Type type = dataitem.GetType();
    T itemvalue = (T)type.GetProperty(itemkey).GetValue(dataitem, null);
    return itemvalue;
}
```

با استفاده از این تابع براحتی خواهید توانست مقدار Code را بدست بیاورید <=

```
int code = GetValueFromAnonymousType<int>(cmbCustomerName.SelectedItem, "Code");
```

این یک مثال از نوع‌های بی نام بوده و تحت هر شرایطی که نیاز به دریافت مقدار از یک نوع بی نام داشته باشید می‌توانید به همین ترتیب عمل کنید

نظرات خوانندگان

نویسنده: محسن خان
تاریخ: ۲۲:۴۵ ۱۳۹۲/۰۳/۱۸

یک سؤال: combo box عموماً selectedText و selectedValue هم دارند. در این حالت عموماً مقدار این‌ها رشته‌ای است. در این صورت نیازی به این cast نیست.

نویسنده: مجتبی صحرائی
تاریخ: ۲۲:۵۱ ۱۳۹۲/۰۳/۱۸

بله درست. هدف از بیان این مثال درک مفهوم AnonymousType و استفاده از تابع GetValueFromAnonymousType بود
میتونید این تابع رو در Library خودتون جای بدید و به موقع از اون استفاده کنید

نویسنده: پدram جباری
تاریخ: ۱۲:۵ ۱۳۹۲/۰۳/۱۹

سلام؛ روش شما درست هست ولی از dynamic هم میشه استفاده کرد که در زمان compile ایرادی نگیره.

1. فرستادن ایمیل‌ها با وقفه زمانی
 2. نفرستادن پشت سر ایمیل‌ها به یک host خاص
 3. استفاده نکردن از کلمه‌هایی که احتمال اسپم شناخته شدن ایمیل را افزایش می‌دهند در قسمت Subject Email
- در لینک‌های زیر لیست بعضی از این کلمات را می‌توانید مشاهده کنید:

<http://blog.hubspot.com/blog/tabid/6307/bid/30684/The-Ultimate-List-of-Email-SPAM-Trigger-Words.aspx>

<http://www.inmotionhosting.com/support/edu/everything-email/spam-prevention-techniques/common-spam-words>

4. فعال نکردن high priority

با فعال شدن این گزینه ایمیل شما مورد بررسی‌های بیشتری قرار می‌گیرد و شانس اسپم شناخته شدن آنرا افزایش می‌دهد.

```
mailMessage.Priority = MailPriority.High ; //not good
```

5. Set کردن Encoding صحیح

```
mailMessage.BodyEncoding = System.Text.Encoding.GetEncoding("utf-8");
```

6. استفاده از حداکثر سه image در متن ایمیل

7. اضافه کردن هم htmlview و هم plainview به view ایمیل ارسالی

با مثال نحوه این کار را نشان می‌دهم:

```
System.Net.Mail.AlternateView plainView = System.Net.Mail.AlternateView.CreateAlternateViewFromString(
System.Text.RegularExpressions.Regex.Replace(BodyText, @"<(.|\n)*?>", string.Empty), null,
"text/plain");
System.Net.Mail.AlternateView htmlView =
System.Net.Mail.AlternateView.CreateAlternateViewFromString(BodyText, null, "text/html");
mailMsg.AlternateViews.Add(plainView);
mailMsg.AlternateViews.Add(htmlView);
```

منابع:

<http://stackoverflow.com/questions/5042309/email-messages-going-to-spam-folder>

<http://www.andreas-kraus.net/blog/tips-for-avoiding-spam-filters-with-systemnetmail>

نظرات خوانندگان

نویسنده:

حمید حسین وند

تاریخ:

۱۱:۴۷ ۱۳۹۳/۰۱/۱۴

سلام

لطفا کدهای فوق رو به صورت زیر اصلاح کنید. هنگام استفاده مشکل دارند.

```
AlternateView plainView = AlternateView.CreateAlternateViewFromString(
    Regex.Replace(EmailBody, @"<(.|\n)*?>", string.Empty), null, "text/plain");
AlternateView htmlView = AlternateView.CreateAlternateViewFromString(EmailBody, null, "text/html");
MyMsg.AlternateViews.Add(plainView);
MyMsg.AlternateViews.Add(htmlView);
```

البته من namespace هارو در بالای کلاس تعریف کردم و از اینجا حذف کردم.

```
using System.Net.Mail;
using System.Text.RegularExpressions;
```

با تشکر

هنگامیکه می‌خواهید در متدهای خود مقداری (از هر نوع datatype دلخواه) را return نمایید، در حالت عادی قادر خواهید بود که فقط از یک return در بدنه متد خود استفاده نمایید:

```
public int Sum(int a, int b)
{
    return a + b;
}
```

اما چنانچه از متدهای تکرار شونده استفاده نمایید، چطور؟

متدهای تکرار شونده یا Iterator method ها، در داخل یک collection به صورت دلخواه iterate کرده یا به اصلاح پیمایش می‌کنند. این متدها از کلمه کلیدی Yield در هنگام return کردن مقادیر استفاده می‌کنند. (در C# از Yield return و در VB از Yield استفاده می‌شود) به عبارت دیگر یک متد با خروجی از نوع قابل پیمایش (مانند IEnumerable)، با استفاده از چند yield return، دارای قابلیت پیمایش و بازگرداندن چندین مقدار به جای یک مقدار واحد می‌گردد.

برای درک بهتر مسئله از مثالی برای ادامه توضیحات استفاده می‌کنم. متد پیمایش شونده (Iterate method) زیر را در نظر بگیرید که خروجی IEnumerable دارد:

```
public static IEnumerable SomeNumbers()
{
    yield return 3;
    yield return 5;
    yield return 8;
}
```

برای استفاده از مقادیر بازگشتی متد بالا از حلقه foreach زیر استفاده می‌نمایم:

```
static void Main()
{
    foreach (int number in SomeNumbers())
    {
        Console.Write(number.ToString() + " ");
    }
    // Output: 3 5 8
    Console.ReadKey();
}
```

حلقه foreach فوق، در پایان اولین پیمایش، عدد 3 را باز گردانده و مکان این return را حفظ می‌کند. در چرخه بعدی عدد 5 را باز می‌گرداند و این نقطه را نیز نگه می‌دارد و در چرخه پایانی عدد 8 را برگردانده و سپس حلقه با رسیدن به نقطه پایانی متد، خاتمه می‌یابد.

برای خاتمه پیمایش در Iterator method ها، می‌توانید از foreach استفاده کنید و یا اینکه عبارت yield break را بعد از تمامی yield return ها به کار گیرید:

```
public static IEnumerable SomeNumbers()
{
    yield return 3;
    yield return 5;
    yield return 8;
    yeild break;
}
```

- در هنگام ایجاد Iterator method ها، نوع مقادیر خروجی متد ، باید یکی از انواع IEnumerable, IEnumerable<T>, IEnumarator<T> یا IEnumarator باشد.
- در هنگام declare کردن ، نمی‌توانید از پارامترهای ref و out استفاده نمایید.
- در Anonymous method ها (متدهای بی نام) و Unsafe block ها نمی‌توانید از yield return (yield در VB) استفاده نمایید.
- نمی‌توانید از Yield return در بلوکهای try-catch استفاده کنید. اما می‌توانید در قسمت try بلوک try-finally استفاده نمایید.
- از yield break می‌توانید در بلوک try و یا بلوک catch استفاده نمایید ، اما در بلوک finally خیر.
- هنگام بروز خطا در foreach هایی که خارج از iterator method استفاده می‌شوند، بلوک finally داخل این متدها اجرا می‌گردد.

مثالی دیگر با استفاده Iterator method ها و yield return جهت بازگرداندن روزهای هفته:

```
static void Main()
{
    DaysOfTheWeek days = new DaysOfTheWeek();
    foreach (string day in days)
    {
        Console.Write(day + " ");
    }
    // Output: Sun Mon Tue Wed Thu Fri Sat
    Console.ReadKey();
}

public class DaysOfTheWeek : IEnumerable
{
    private string[] days = { "Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat" };
    public IEnumerator GetEnumerator()
    {
        for (int index = 0; index < days.Length; index++)
        {
            // Yield each day of the week.
            yield return days[index];
        }
    }
}
```

منابع:

[yield](#) , [Iterators](#)

نظرات خوانندگان

نویسنده: محسن جمشیدی
تاریخ: ۱۳۹۲/۰۶/۲۵ ۸:۴۵

گمان کنم "پیمایش" ترجمه مناسب‌تری نسبت "چرخش" باشد.
به نظر ترجمه Iterator Method به "متد تکرار شونده" منظور رو بدرستی منتقل نمی‌کنه اما کمله مناسبی به ذهنم نمی‌رسه

نویسنده: شهره مرتضوی
تاریخ: ۱۳۹۲/۰۶/۲۵ ۱۰:۲۵

با تشکر از حسن نظر شما. کلمه "پیمایش" رو مد نظر قرار خواهم دادم. در مورد "متد تکرار شونده" ، اگر دوستان کلمه بهتری به ذهنشون رسید ، بفرمایند تا تغییر بدم.

در [قسمت قبل](#) در مورد استفاده دات نت در متلب توضیح داده شد. در این قسمت به نحوه استفاده توابع متلب در دات نت بصورت ساده می‌پردازیم.

فرض کنید تیم برنامه‌نویس متلب و تیم برنامه‌نویس دات نت در تعامل با یکدیگر هستند. وظیفه تیم برنامه نویسی متلب به شرح زیر می‌باشد :

- 1- نوشتن توابع در متلب و تست کردن آنها جهت توسعه و ارائه مناسب به تیم مقابل
- 2- درست کردن کامپوننت دات نت در متلب با استفاده از محیط Deployment Tool GUI (با اجرای دستور deploytool در متلب)
- 3- استفاده از یک پکیج بسته‌بندی شده از فایل‌های قابل ارائه به تیم مقابل (اختیاری)
- 4- کپی پکیج در محل از قبل تعیین شده توسط دو تیم یا ارائه آن به تیم مقابل جهت استفاده

برای مثال M فایل (اصطلاح فایل‌ها در متلب همانند کلاس در دات نت) makesquare.m را که در مسیر

```
matlabroot\toolbox\dotnetbuilder\Examples\VS8\NET\MagicSquareExample\MagicSquareComp
```

است را در نظر بگیرید :

```
function y = makesquare(x)
%MAKESQUARE Magic square of size x.
% Y = MAKESQUARE(X) returns a magic square of size x.
% This file is used as an example for the MATLAB
% Builder NE product.

% Copyright 2001-2012 The MathWorks, Inc.

y = magic(x);
```

تابع magic یک ماتریس در ابعاد x در x درست می‌کند که درایه‌های آن اعداد صحیح از 1 تا x^2 بوده و مجموع سطر و ستون‌های آن با هم برابر است. x باید بزرگتر یا مساوی 3 باشد.

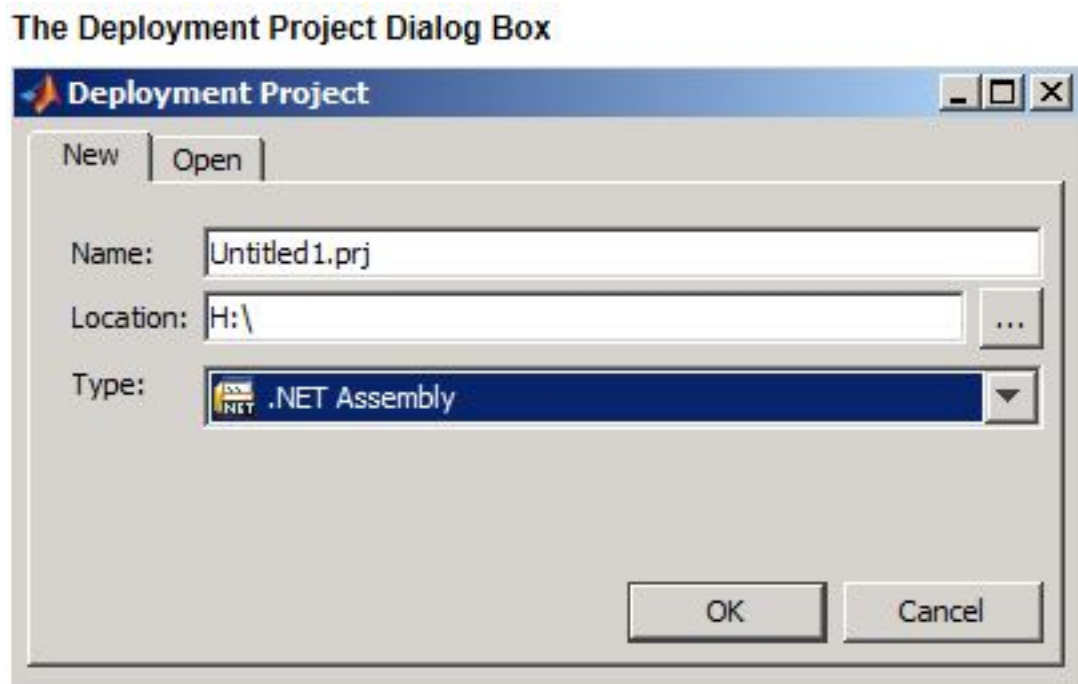
در صورتی که x برابر 5 انتخاب شود خروجی متلب بصورت زیر خواهد بود :

```
17 24 1 8 15
23 5 7 14 16
4 6 13 20 22
10 12 19 21 3
11 18 25 2 9
```

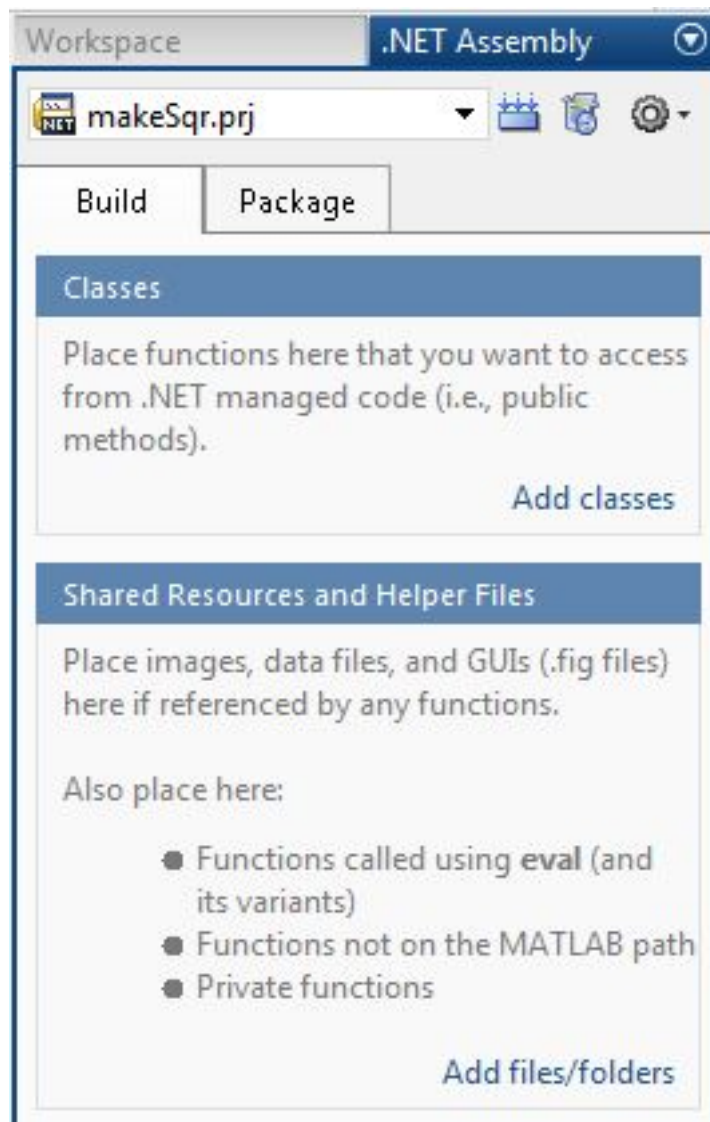
در قسمت تهیه یک کامپوننت دات نت اطلاعات زیر را در نظر بگیرید :

Project Name	makeSqr
Class Name	MLTestClass
File to compile	makesquare.m

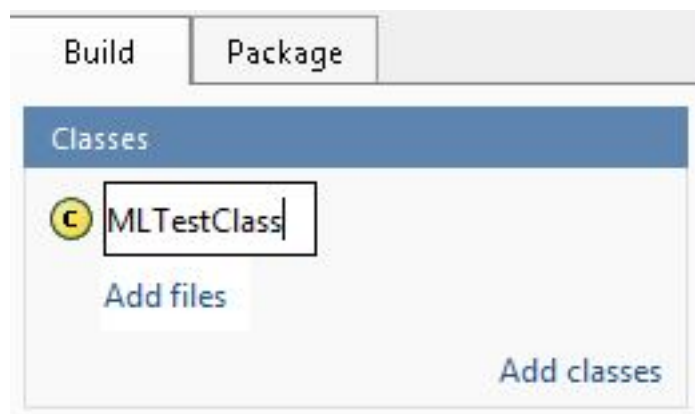
سپس برای درست کردن کامپوننت در محیط Deployment Tool GUI برنامه متلب را اجرا کرده و در پنجره command دستور deploytool را اجرا کنید تا پنجره زیر باز شود :



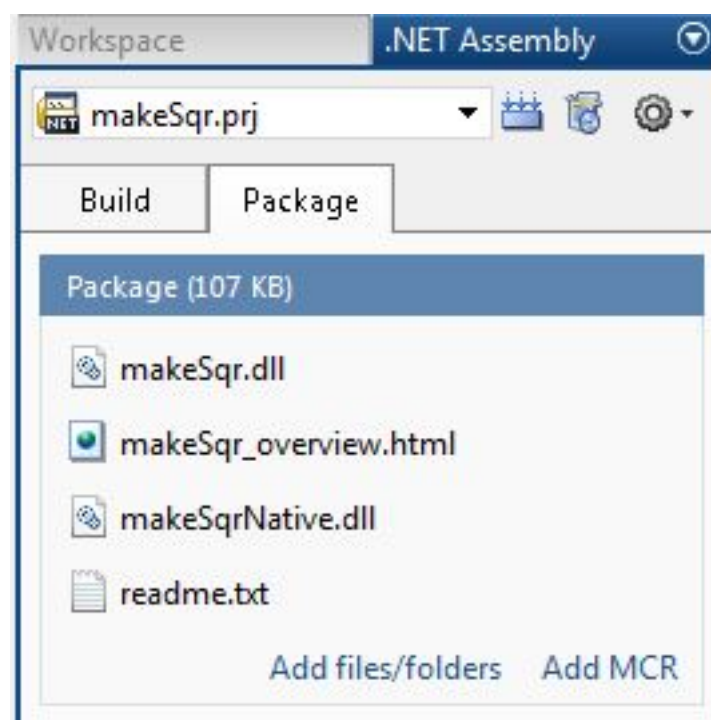
نام و مسیر پروژه را تعیین کنید سپس از منوی کشویی نوع پروژه، که دات نت اسمبلی باشد را انتخاب کنید. پنجره‌ای در به شکل زیر مشاهده خواهد شد :



در تب build اگر قصد استفاده از اپلیکیشن COM را دارید و یا فایل‌هایی جهت تکمیل پروژه قصد پیوست دارید را در قسمت پایین Add files را انتخاب کنید. و اگر قصد استفاده از اپلیکیشن دات نت را دارید قسمت بالایی Add classes را انتخاب کنید و نام کلاس را وارد کنید.

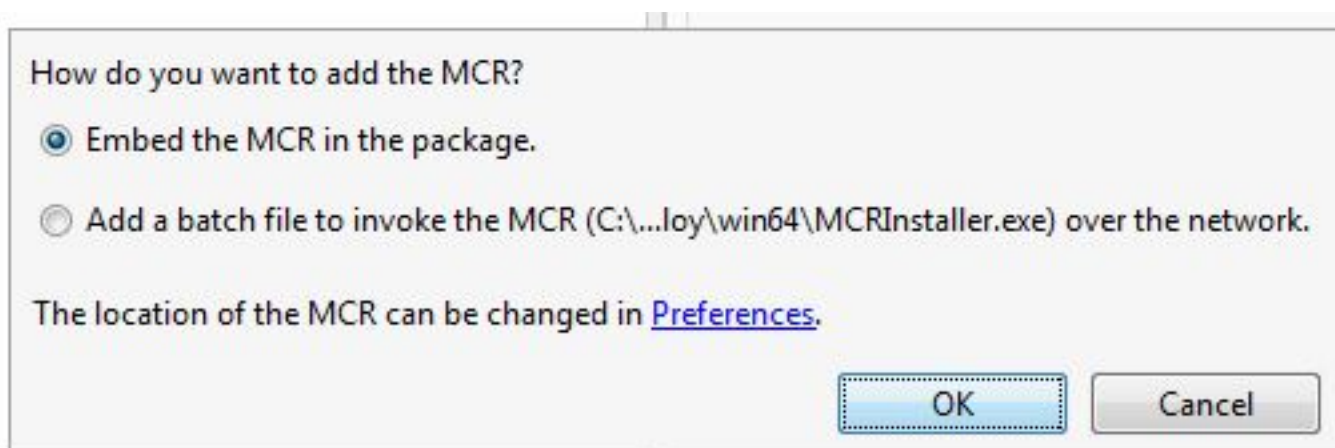


سپس برای کلاس مورد نظر فایل‌های متلبی که قصد کامپایل کردن آنها را دارید از قسمت Add files پیوست کنید. در صورتیکه قصد اضافه کردن کلاس اضافی را داشتید مجدداً مراحل را طی کنید. در انتها دکمه build را زده تا عملیات کامپایل آغاز شود. اما برای استفاده تیم برنامه‌نویسی دات نت احتیاج به کامپایلر متلب می‌باشد که این مهم در پکیجی که به این تیم ارائه خواهد شد مد نظر قرار خواهد گرفت. در قسمت تب Package گزینه Add MCR را انتخاب نمایید :



بعد از انتخاب، دو گزینه برای انتخاب وجود دارد که بطور خلاصه گزینه اول فایل‌های کامپایلر متلب در داخل پروژه جهت ارائه قرار می‌گیرد. همچنین این گزینه جهت استفاده در مواقع درون شبکه‌ای، مواردی که فضای دیسک و عملکرد و چندان اهمیت ندارد مورد استفاده قرار می‌گیرد. اما گزینه دوم عکس قضیه بالا عمل می‌کند و برای تعداد یوزر بالا و شبکه‌ای و ... مورد استفاده می‌باشد.

در اینجا گزینه اول را انتخاب می‌کنیم. در صورتیکه فایل‌های دیگری جهت ضمیمه به پکیج احتیاج است به آن اضافه می‌کنیم.



سپس کلید پکیج را زده تا پکیج مورد نظر آماده شود. دقت داشته باشید که بعد از انتخاب کامپایلر متلب، حجم پکیج نزدیک به 400 مگابایت خواهد شد. پکیج مورد نظر بصورت یک فایل exe فشرده خواهد شد. معمولاً پکیج شامل فایل‌های زیر باید باشد :

componentName.xml	Documentation files
componentName.pdb (if Debug option is selected)	Program Database File, which contains debugging information
componentName.dll	Component assembly file
MCR Installer	MCR Installer (if not already installed on the target machine).

بعد از طی مراحل فوق نوبت به تیم برنامه‌نویسی دات نت می‌رسد. بعد از دریافت پکیج از تیم برنامه‌نویسی متلب در صورتیکه بر روی سیستم هدف کامپایلر متلب و یا خود متلب نصب نیست باید از داخل پکیج این کامپایلر نصب شود. دقت داشته باشید که ورژن کامپایلر بر روی سیستم باید با ورژن پکیج دریافتی یکی باشد. در VS یک پروژه کنسول ایجاد کنید و از فولدر پکیج پروژه دریافتی در زیرفولدر distrib فایل makeSqr.dll را به رفرنس برنامه VS اضافه کنید. در ادامه از مسیر نصب کامپایلر فایل MArray.dll را هم به رفرنس پروژه اضافه کنید. این فایل جهت تبادل داده اپلیکیشن با کامپایلر متلب مورد استفاده قرار می‌گیرد.

installation_folder\toolbox\dotnetbuilder\bin\architecture\framework_version

اسمبلی‌های زیر را به کلاس Program برنامه اضافه کنید :

```
using System;
using MathWorks.MATLAB.NET.Arrays;
using MyComponentName;
```

سپس کدهای زیر را به کلاس فوق اضافه نمایید :

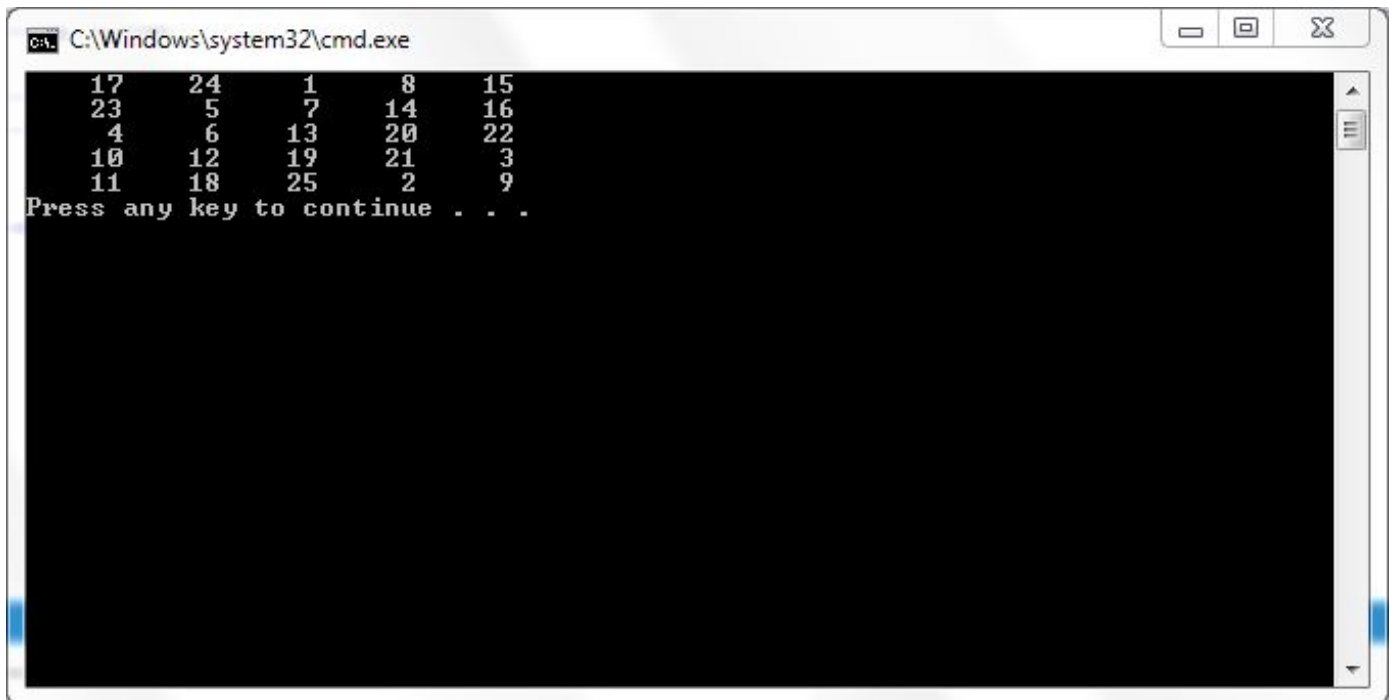
```
static void Main(string[] args)
{
    MLTestClass obj = new MLTestClass();
    MArray[] result = obj.makesquare(1, 5);

    MWNumericArray output = (MWNumericArray)result[0];
    Console.WriteLine(output);
}
```

توضیحات کدهای فوق :

- 1- MWNumericArray یک اینترفیس جهت تعیین و نمایش نوع آرایه‌های عددی در متلب است.
- 2- MArray یک کلاس abstract جهت دسترسی، فرمت‌دهی و مدیریت آرایه‌های متلب می‌باشد.
- 3- عدد 1 مشخص کننده تعداد خروجی تابع متلب و عدد 5 ورودی تابع می‌باشد.

خروجی برنامه همانند خروجی متلب بصورت زیر خواهد بود :



```
C:\Windows\system32\cmd.exe

17    24    1    8    15
23    5    7    14   16
 4    6   13   20   22
10   12   19   21    3
11   18   25    2    9
Press any key to continue . . .
```

نکته:

ورژن فریمورک دات نت در هنگام کامپایل با ورژن Mwarrray.dll باید یکی باشد.

نظرات خوانندگان

نویسنده: سید امیر سجادی
تاریخ: ۱۱:۲۶ ۱۳۹۲/۰۸/۲۸

بسیار عالی بود. ممنون

نویسنده: بهار
تاریخ: ۱۳:۸ ۱۳۹۳/۰۵/۱۲

آموزشتون عالی بود. ممنون

ولی من نمی‌تونم برنامه را اجرا کنم. با پیغام زیر مواجه می‌شم

Could not load file or assembly 'MWMArray, Version=2.11.0.0, Culture=neutral, PublicKeyToken=e1d84a0da19db86f' or one of its dependencies

برنامه را با VS 2010 و متلب R2011a در ویندوز 7 ، 64 بیتی نوشتم. از اینکه راهنمایی می‌کنید سپاسگزارم.

سوال دیگه ای که دارم، اینه که، چه جوری می‌تونم به جای برنامه ماتریس جادویی از شبکه عصبی استفاده کنم؟ آیا تغییر این دو برنامه فقط در تب Build است و نیازی به تغییر در تب package ندارد؟

نویسنده: محسن خان
تاریخ: ۱۵:۲۰ ۱۳۹۳/۰۵/۱۲

برنامه‌ی نوشته شده اگر کد خالص دات نت هست، [مشکلی با 64 بیت و 32 بیت نداره](#) . اما اگر داخلش ناخالصی native وجود داره، مثلاً از یک DLL بومی ویندوز استفاده می‌کنه که دات نت نیست و همچنین این DLL از نوع 32 بیتی هست و برنامه روی Any CPU تنظیم شده، حتماً کرش می‌کنه با خطایی که گفتید. راه حلش اینه که در خواص پروژه، any CPU را به X86 تغییر بدید.

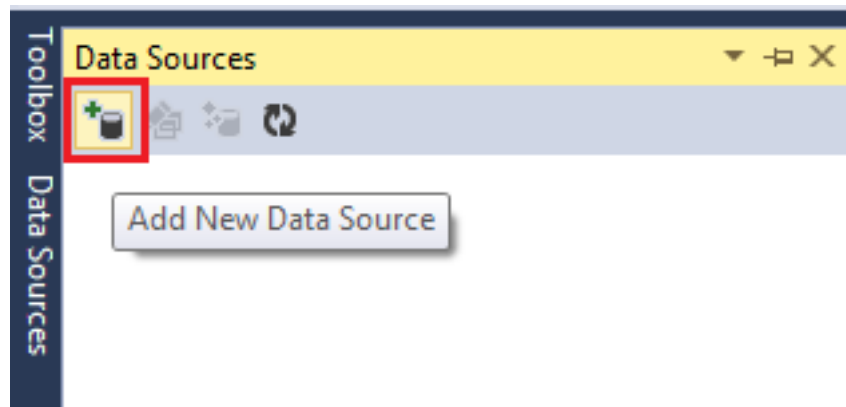
نویسنده: مسعود مشهدی
تاریخ: ۱۷:۲۳ ۱۳۹۳/۰۵/۱۲

یک نکته دیگه رو هم دقت کنید که ورژن فریمورک دات نت در هنگام کامپایل با ورژن Mwmarray.dll باید یکی باشد. در مورد ماتریس یا شبکه‌های عصبی کلیت تفاوتی نمی‌کند همانطور که این مثال یک تابع ماتریس magic هست شبکه‌های عصبی هم همان تابع هست با این تفاوت که از توابع تو در تو تشکیل شده است.

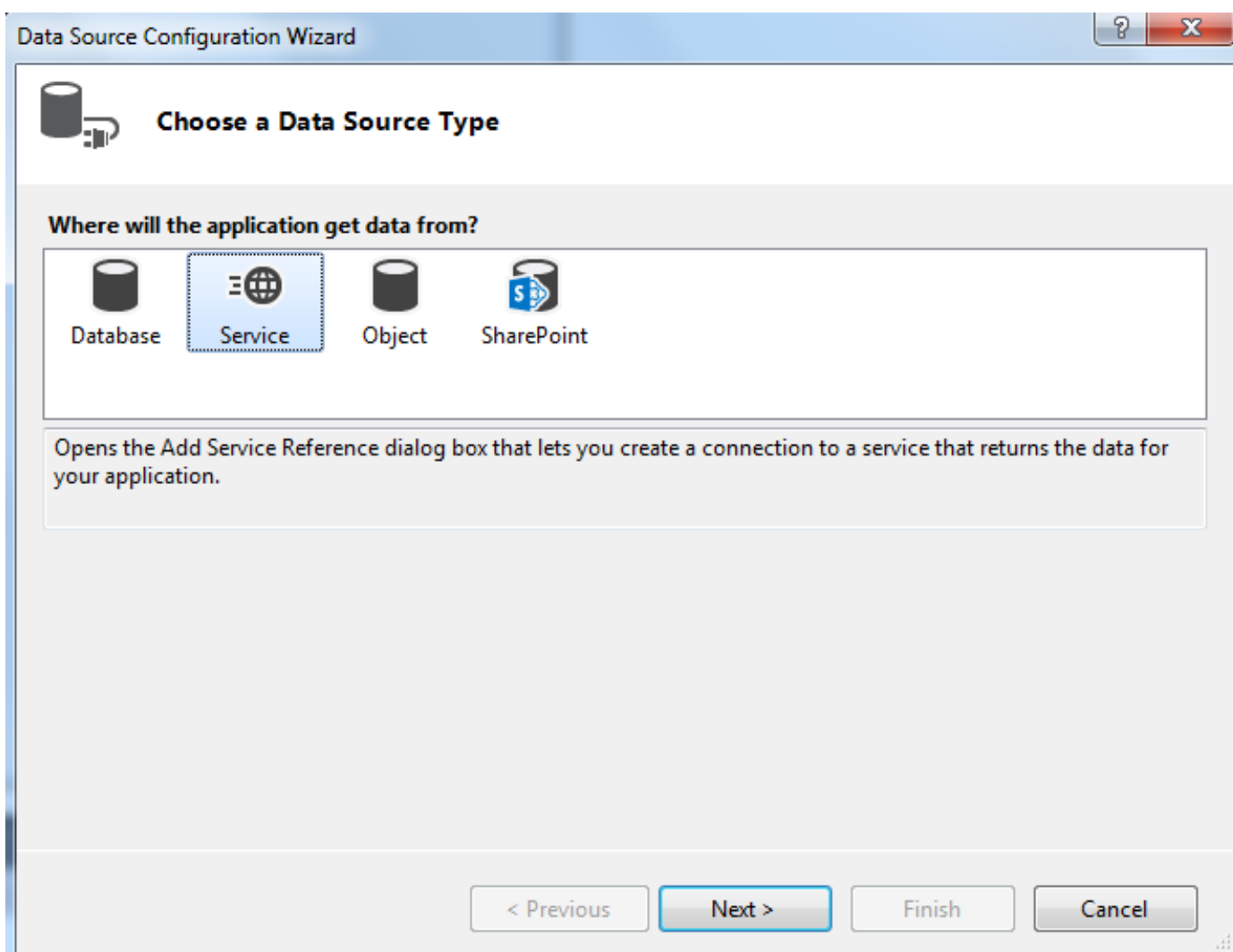
نویسنده: بهار
تاریخ: ۲۲:۱۷ ۱۳۹۳/۰۵/۱۲

ممنون از اینکه پاسخ دادید. مشکل حل شد. به جای X86 از Any CPU استفاده کردم و برنامه اجرا شد

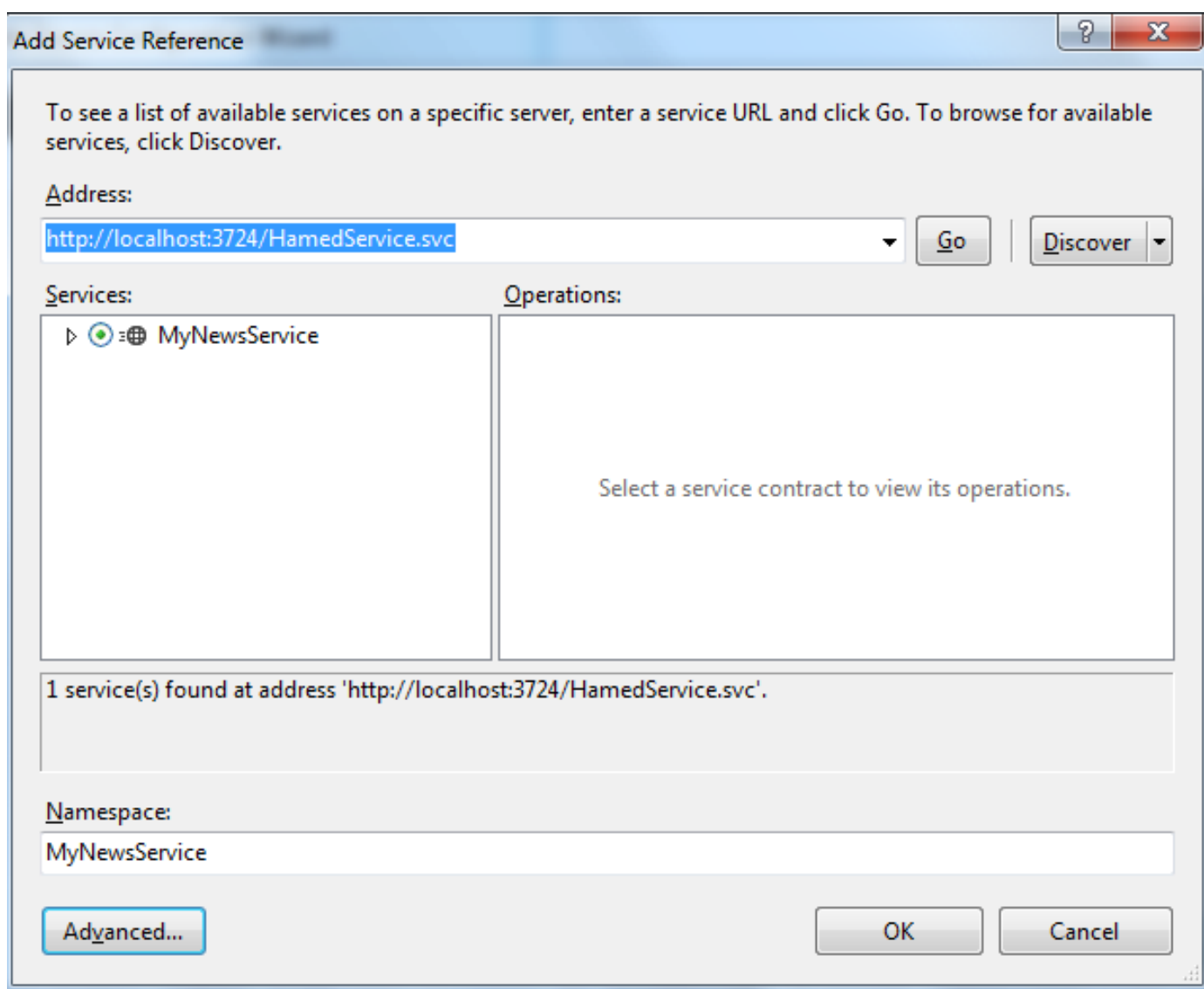
در Solution Explorer روی نام راه حل - MyNews - راست کلیک کنید و Add-> New Project را انتخاب کنید. سپس یک پروژه از نوع Windows Forms Application انتخاب کنید و نام آن را MyNewsWinApp بگذارید. یا کلیدهای ترکیبی Shift + Alt + D پنجره‌ی Data Sources را نمایان کنید. برابر با شکل روی ابزار Add New Data Source کلیک کنید:



از پنجره‌ی باز شده روی گزینه‌ی Service کلیک کنید:



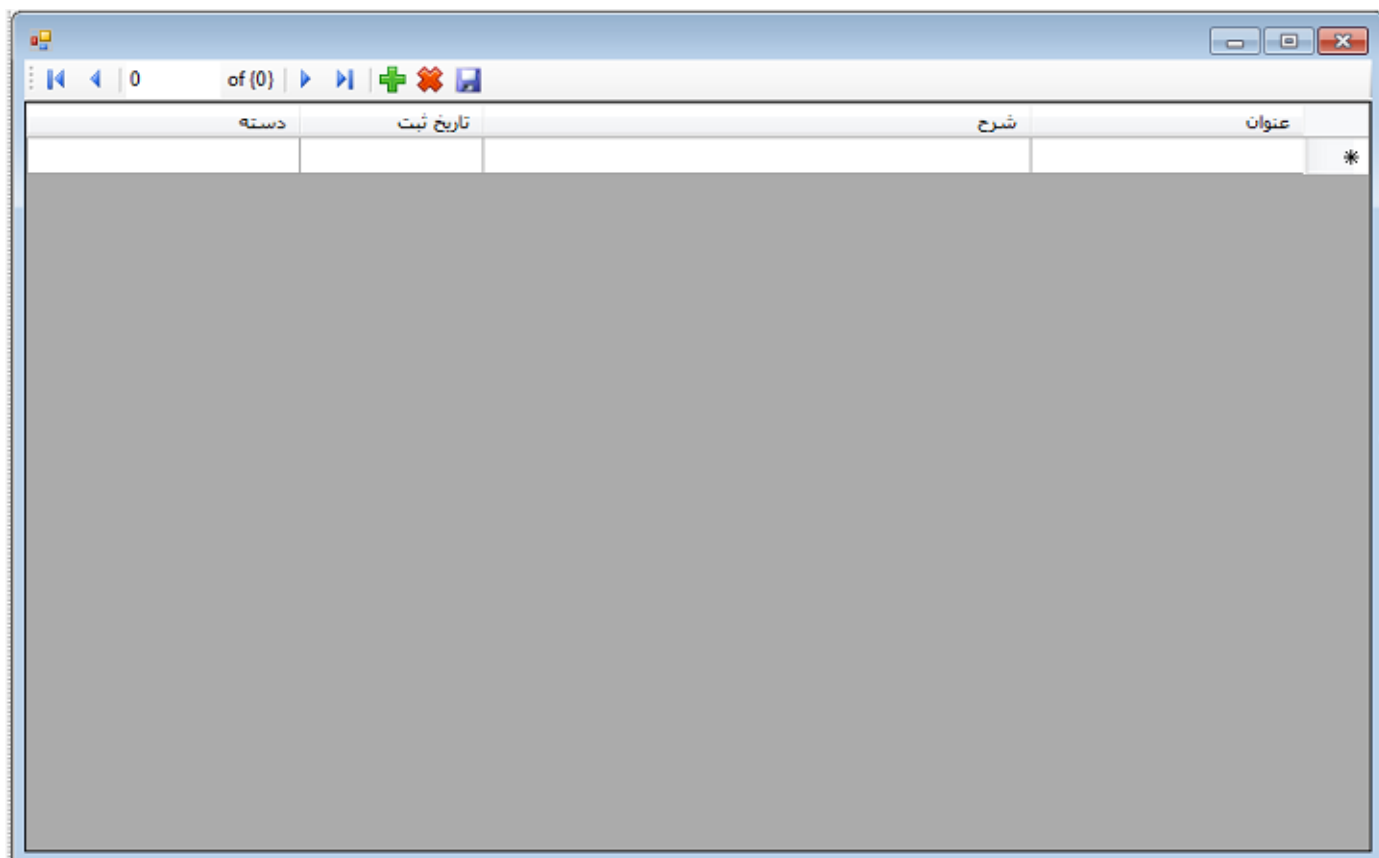
روی گزینه‌ی Next کلیک کنید و در پنجره‌ای که باز می‌شود در قسمت Address نشانی وب‌سایتی که در بخش پیشین تولید کردیم و ممکن است شما در IIS افزوده باشید؛ قرار دهید و روی دکمه‌ی GO بفشارید تا سرویس در کادر پایین افزوده شود. سپس در قسمت Namespace نامی مناسب برای فراخوانی سرویس وارد کنید آن‌گاه دکمه‌ی OK را بفشارید.



از پنجره‌ی بازشده روی دکمه‌ی Finish کلیک کنید. پس از مکثی کوتاه سرویس به همراه دو موجودیت آن درون Data Sources دیده خواهد شد. از آن‌طرف در Solution Explorer نیز در پوشه‌ی Service References سرویس تعریف‌شده ارجاع داده خواهد گرفت.

از Data Sources روی tblNews کلیک کنید سپس آن‌را کشیده و به روی فرم رها کنید. خواهید دید که یک DataGridView شامل همه‌ی ویژگی‌های موجودیت tblNews و یک Binding Navigator که با موجودیت tblNews در پیوند است و یک منبع داده به نام tblNewsBindingSource به صورت خودکار در فرم افزوده خواهد شد.

چیدمان فرم، رنگ‌ها، اندازه‌ها و فونت را آن‌گونه که می‌پسندید تنظیم کنید. سپس ستون‌هایی که به آن‌ها نیازی ندارید حذف یا پنهان کرده و عنوان ستون‌های مانده را ویرایش کنید. کلیدهای افزودن، حذف و ذخیره را روی Navigator ایجاد کنید و بقیه‌ی کلیدها را اگر به آن نیازی ندارید حذف کنید. البته می‌توانید بنا به سلیقه‌ی کاری‌تان یک Panel برای این‌کار اختصاص دهید. در این‌جا یک فرم ساده در نظر گرفته شده است:



اکنون نوبت به کدنویسی است. سورت فرم را باز کنید و نخست سرویس را به این صورت در جای مناسب تعریف کنید:

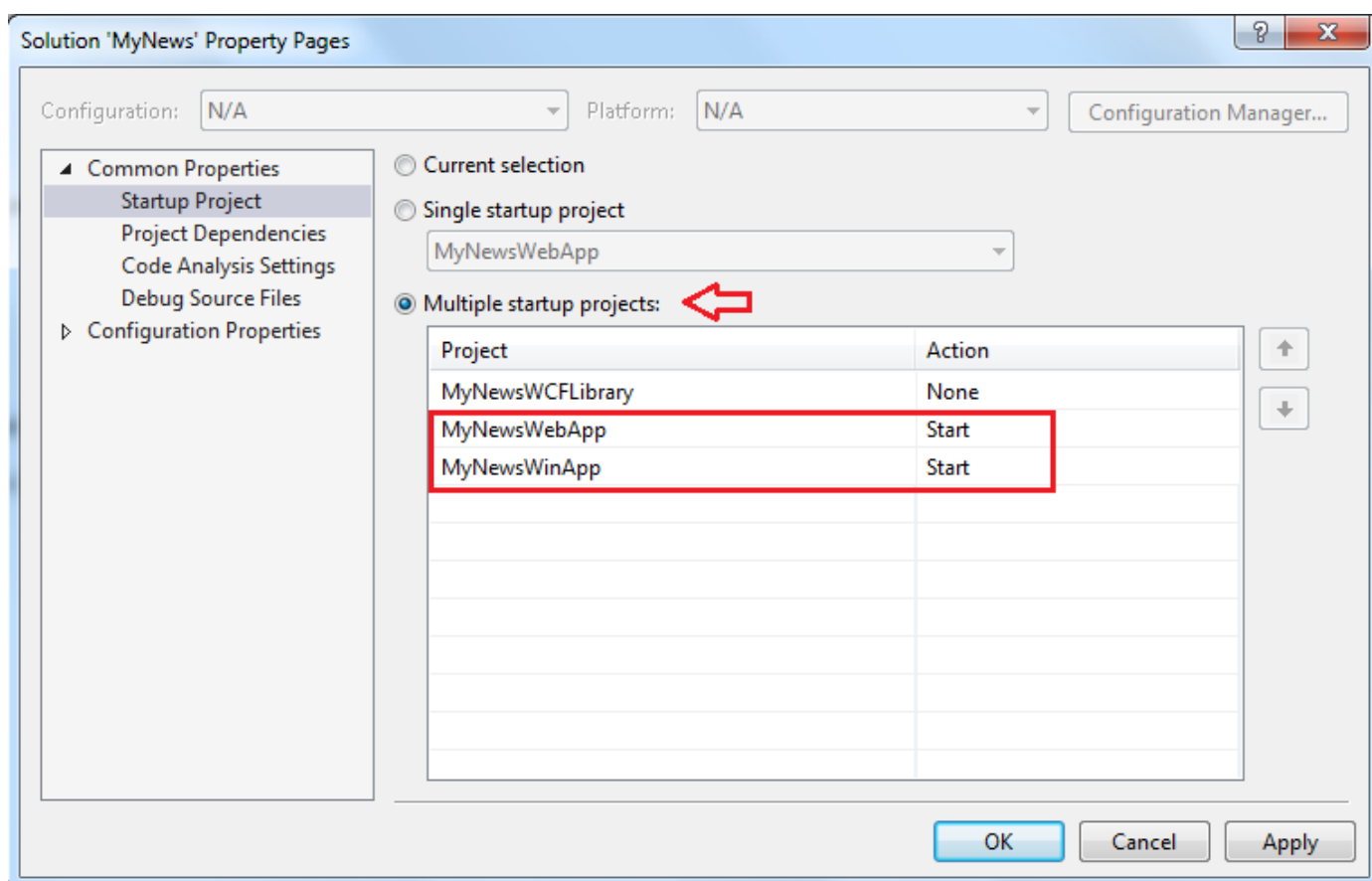
```
MyNewsService.MyNewsServiceClient MyNews = new MyNewsService.MyNewsServiceClient();
```

یک تابع کوچک برای تبدیل تاریخ میلادی به شمسی بنویسید سپس رویداد Load فرم را به این صورت بنویسید:

```
string MiladiToShamsi(DateTime MyDate)
{
    System.Globalization.PersianCalendar pers = new System.Globalization.PersianCalendar();
    return string.Format("{0}/{1}/{2}", pers.GetYear(MyDate),
        pers.GetMonth(MyDate).ToString("D2"), pers.GetDayOfMonth(MyDate).ToString("D2"));
}

private void Form1_Load(object sender, EventArgs e)
{
    tblNewsBindingSource.DataSource = MyNews.GetAllNews().Select(p => new {p.tblNewsId,
        p.tblCategory.CatName, p.Title, p.Description, RegDate= MiladiToShamsi( p.RegDate) });
}
```

پیش از اجرای پروژه از Solution Explorer روی نام راه حل راست کلیک کنید و گزینه‌ی Properties را انتخاب کنید. در پنجره‌ی باز شده تنظیمات زیر را انجام دهید:



این کار باعث می‌شود که به طور هم‌زمان پروژه‌ی وب‌سایت و ویندوز اجرا شود. اکنون پروژه را اجرا کنید. اگر با پیغام خطا روبه‌رو شدید؛ تگ Connection String را از App.Config پروژه WCF Library به Web.Config پروژه وب‌سایت کپی کنید. در این صورت پروژه به راحتی اجرا خواهد شد.

عنوان	شرح	تاریخ ثبت	نام دسته
رئیس کمیته ملی المپیک انتخاب شد	کیومرث هاشمی رئیس کمیته ملی المپیک شد. به گزارش ایسنا، چهل‌ویکمین دوره انتخابات کمیته ملی المپیک پس از یک سال و نیم تاخیر از ساعت ۱۵ امروز (دوشنبه) آغاز شد که پس از رای‌گیری برای پست ریاست، کیومرث هاشمی به عنوان رئیس کمیته ملی المپیک انتخاب شد.	۱۳۹۲/۱۰/۲۷	ورزشی
صعود آسان استقلال و تراکتورسازی	تیم های فوتبال استقلال و تراکتورسازی، با غلبه بر حریفان خود به مرحله یک چهارم نهایی جام حذفی ایران صعود کردند. مس کرمان نیز نفت امیدیه را ۲ بر ۱ مغلوب کرد تا جمع تیم های صعود کننده به مرحله یک چهارم نهایی جام حذفی تکمیل شود.	۱۳۹۲/۱۰/۲۸	ورزشی

در بخش پسین پیرامون افزودن، ویرایش و حذف و برخی توضیحات برای توسعه‌ی کار خواهیم نوشت.

شاید برای شما هم پیش آمده باشد که بخواهید در هر بار واکنشی لیستی از اطلاعات، مثلاً از دیتابیس، آیتمهای آن را بصورت تصادفی مرتب کنید.

من در پروژه اخیرم برای نمایش یک سری سوال مجبور بودم که در هر بار نمایش سوالات، لیست را به صورت رندوم مرتب کنم و به کاربر نمایش بدم. برای حصول این مهم، یک extension method به شکل زیر نوشتم:

```
public static class RandomExtentions
{
    public static void Shuffle<T>(this IList<T> list)
    {
        Random rng = new Random();
        Thread.Sleep(100);
        int n = list.Count;
        while (n > 1)
        {
            n--;
            int k = rng.Next(n + 1);
            T value = list[k];
            list[k] = list[n];
            list[n] = value;
        }
    }
}
```

در این تابع که اسمش را Shuffle گذاشتم، با دریافت یک لیست از نوع T، آیتمهای درون لیست را به صورت تصادفی مرتب می‌کند.

مثال :

```
var x = new List<int>();
x.Add(1);
x.Add(2);
x.Add(3);
x.Add(4);
x.Add(5);
x.Shuffle();
```

در این مثال لیست x که از نوع int میباشد پس از فراخوانی Shuffle به یک لیست نامرتب تبدیل میشود که نحوه چیدمان در هر بار فراخوانی، تصادفی خواهد بود.

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۴:۱۶ ۱۳۹۳/۰۲/۰۷

اگر از EF استفاده می‌کنید، برای اینکار یک ستون Guid پویا را اضافه می‌کند. سپس بر اساس این ستون، مرتب سازی را انجام می‌دهد. [اطلاعات بیشتر](#)

نویسنده: بهزاد دات نت
تاریخ: ۱۴:۳۰ ۱۳۹۳/۰۲/۰۷

با سپاس از شما. در صورت استفاده از EF روشی که شما فرمودین بهتر و کارآمدتر هستش.