

بسم الله الرحمن الرحيم



مجموعه نکات برنامه نویسی در سی شارپ دات نت

این مجموعه حاصل جمع آوری تمام نکات ریز و درشت برنامه نویسی در سی شارپ دات نت در تالار برنامه نویس است که تا تاریخ ۱۳۸۱/۶/۳۰ از بخش ۱۰۱ نکته در سی شارپ

و سایر قسمت های تالار برنامه نویس استخراج شده است. **تقدیم به تمام برنامه نویسان ایرانی**

جمع آوری مطالب و تهیه مقاله: EhsanAvr

وب سایت: www.EhsanAvr.Com

ایمیل: Info@EhsanAvr.Com

منبع: www.Barnamenevis.Org

استفاده از مطالب این مقاله در هر شکلی مجاز است.

فهرست:

توجه: نکته‌ها از صفحه‌ی ۱۱ شروع می‌شوند و فهرست تا صفحه‌ی ۱۰ ادامه دارد.

اختصاص دادن رویداد به کنترل‌های ساخته شده در runtime

اجرا کردن یک فایل اجرایی با کدنویسی

اتصال به بانک‌های اطلاعاتی اوراکل در سی شارپ

اتصال به SqlServer با استفاده از IP Address

Serialize کردن باینری آبجکت در فایل

Serialize کردن object در XML

Serialize کردن object در SOAP XML

Shutdown, Restart و Logoff کردن یک کامپیوتر بوسیله کد

خواندن از Registry ویندوز

restart کردن (بستن و مجدداً اجرا کردن) برنامه

Resize کردن یک تصویر با کدنویسی

picture box

Map کردن درایوهای شبکه به کمک اجرای دستورات خط فرمان

Initilize کردن کالکشن‌ها در دات نت ۳.۰ به بالا

readonly کردن combobox

Ping کردن یک کامپیوتر تحت شبکه

Dock و Anchor چپ هستند ؟

DeSerialize کردن باینری آبجکت از فایل

Standby و Hibernate کردن کامپیوتر

فایل swf

API SndPlaySound در C# Mamnoon

حاشیه دار کردن کنترل TextBox با یک رنگ دلخواه

DeSerialize کردن object از SOAP XML

Crop کردن یک تصویر (یک برش مستطیلی از تصویر)

رسم یک چندضلعی توپر روی یک فرم

محو شدن تدریجی یک فرم با تغییر دادن خاصیت Opacity

انتقال آیتمهای یک آرایه از اعداد به یک لیست باکس

پیاده سازی حالت ساده ی الگوی سینگلتون در یک کلاس

انجام عملیات متداول روی فایلها و دایرکتوریه‌ها

کشیدن نقاشی با توابع ساده گرافیکی و ذخیره خروجی در یک فایل

ویرایش فقط یک سطر از DataGridView

انتقال آیتمهای Enum به یک ListBox

انجام عملیات متداول در رجیستری ویندوز

انجام عملیات متداول بر رشته ها

افزودن کنترل در زمان اجرا

اعمال محدودیت برای ذخیره فایل توسط کاربر

افزودن یک زیرمنو و دستور دلخواه به context menu ویندوز اکسپلورر

افزودن و حذف کردن یک برنامه به Startup ویندوز از طریق کلیدهای رجیستری

اطلاعاتی در مورد موس

اضافه و جدا کردن نام فایل از مسیر کامل

استفاده کارآمد از دستور Using برای آزاد سازی بهتر منابع

اعتبار سنجی آدرسهای ایمیل به کمک عبارات منظم

ارسال متغیر به صفحه بعد با استفاده از url

ایجاد فرمهایی با اشکال غیرمستطیلی

ایجاد طیف رنگ (Gradient) بعنوان پشت زمینه ی کنترلها

ایجاد تصویر Bitmap در زمان اجرا (روش دوم)

ایجاد طیف رنگ با بیش از دو رنگ توسط ColorBlend

انجام محاسبات بر روی یک فیلد از DataTable

انجام کارهای متداول روی درایوها (Partitions)

باز و بسته کردن درایو CD-ROM

ایجاد تصاویر Bitmap در زمان اجرا (روش اول)

ایجاد اشیای گرافیکی

بدست آوردن نام کامپیوتر، نام کاربر جاری و ... (بوسیله کلاس Environment)

بدست آوردن نام شرکت سازنده پراسسور (CPU Manufacturer)

باز کردن یک سایت توسط internet explorer

آیا کاربر جاری Administrator هست؟

ایجاد یک نام تصادفی برای فایل

بدست آوردن مسیر فولدرهای ویژه

بدست آوردن مسیر دایرکتوری جاری برنامه

بدست آوردن مدت زمان بیکاری سیستم

بدست آوردن نام سیستم لوکال

بدست آوردن میزان فضای خالی یک درایو (پارتیشن)

بدست آوردن لیست تمامی فونتهای نصب شده در سیستم

بدست آوردن لیست تمامی فرمهای باز در یک برنامه

بدست آوردن لیست IP های یک سیستم

بدست آوردن خروجی دستورهای خط فرمان

بدست آوردن لیست تمامی ناحیه های زمانی (Time Zone)

بدست آوردن جزییات اطلاعات در مورد یک پروسه در حال اجرا

بدست آوردن حجم کلی یک درایو (پارتیشن)

بدست آوردن اطلاعاتی (مانند: سرعت، نوع و ...) از کانکشنهای شبکه ای موجود لوکال

بدست آوردن اطلاعاتی در مورد OS و CLR

بدست آوردن اطلاعاتی از نسخه ویندوز موجود بوسیله توابع API

بدست آوردن Handle و عنوان پنجره ی فعال جاری

ترسیم یک نوشته در مرکز یک کنترل

ترسیم و نوشتن مستقیم بر دسکتاپ ویندوز

بدست آوردن Handle تمامی پنجره های موجود در سیستم

ترسیم انواع چک باکس ها به کمک GDI+

ترسیم برخی دکمه های ویژه به کمک GDI+

تبدیل یک IP به HostName و بالعکس

ترسیم انواع border های ۳ بعدی به کمک GDI+

تبدیل یک تصویر رنگی به معادل GrayScale آن

تبدیل string به int

پر کردن یک listbox از رنگهای سیستمی

تبدیل انوماتیک دکمه Enter به Tab جهت انتقال فوکوس در کنترلها

پخش فایل صوتی موجود در Resource

پاک کردن یک پردازش از حافظه

پر کردن دو DataGridView به صورت master detail

پخش کردن برخی اصوات و صداهاى سیستمى تنها با یک خط!

برنامه ای مشابه ColorBalance فوتوشاپ

پاک کردن فایل

بهبود رسم شکلهای با روانسازی منحنیها و زاویه ها

به دست آوردن لیست کاربران یک گروه خاص در سیستم

بررسی اتصال به اینترنت Dial up و ADSL

یک کلاس ژنریک برای سریالایز و دیسریالایز کردن به شکل XML

برقراری ارتباط تلفنی (Dial up) توسط TAPI32

تغییر دادن میزان تفکیک پذیری (Resulation) صفحه نمایش

تغییر خواص یک فایل

تغییر Render کنترل های دات نت ۲۰۰۵

تشخیص روشن بودن Caps Lock و Num Lock

تشخیص خالی بودن pictureBox

Method overloading چیست؟

تعریف یک متغیر بدوت تعیین کردن نوع آن در دات نت فریمورک ۲.۰ به بعد

abstract class یا کلاس مجرد چیست؟

تضمین آنکه چطور همیشه پس از اتمام کار با یک آبجکت حتما از حافظه خارج (dispose) شود

خواندن و نوشتن فایل به صورت unicode با استفاده از richTextBox

بافر :: نشان دادن مقدار کپی شدن به فایل ::

خوندن خط به خط یک رشته چند سطری

جابجا کردن فرم با کلیک بر روی هر قسمت از آن - با استفاده از توابع API

تغییر نام دادن (Rename) یک فایل

جابجا کردن فرم با کلیک بر روی هر قسمت از آن - بدون استفاده از توابع API

تغییر زبان پیشفرض ویندوز بصورت داینامیک در برنامه

چطوری همیشه از توابع API استفاده کرد

چگونه فقط یک نسخه از برنامه اجرا بشه !!!!

چرخاندن یک تصویر با زاویه دلخواه

چطور میتونم IP خودمو به دست بیارم ؟

جستجوی فایل در هارد

دستکاری خواص (Attribute) یک فایل

باز و بسته کردن CD Drive

خواندن خواص (Attribute) یک فایل

جدا کردن یک تکه از عکس

ایندهکسرها (Indexers)

گرفتن یک فایل از طریق دیالوگ OpenFileDialog و دستکاری نام فایل

جستجو در دیتاگرید

چگونه یک کلاس را مهر و موم کنیم؟

ذخیره DataGrid در اکسل Excel

دمای CPU

خواندن و نوشتن رنگ یک پیکسل به کمک کلاس Bitmap

خواندن و نوشتن رنگ یک پیکسل به کمک توابع API

دریافت فایل متنی از web در Windows Application

دسترس‌ی به پیکره بندی سیستم در CLR

شفاف کردن پشت زمینه ی کنترلها

شروع آشنایی با LINQ بصورت ساده

خالی کردن سطل بازیافت ویندوز (RecycleBin)

سیاه و سفید کردن عکس با کد نویسی

شبیه سازی عمل کلیک ماوس روی یک کنترل با کدنویسی

سایه دار کردن یک فرم

سریالیزشین چیست ؟

ساختن لیستی از کنترلها در زمان اجرا

روش ارسال ایمیل به چندین گیرنده

ساختن یک سند word و باز کردن آن در MSWord

ساده ترین راه برای جلوگیری از Not Respond شدن برنامه در حلقه های طولانی

کار کردن با کلاس EventLog

قفل کردن کامپیوتر از طریق کدنویسی

کار کردن با کلاس TimeZoneInfo

فقط یک نمونه از برنامه بتواند اجرا شود (با استفاده از WMI)

قطع کردن اتصال اینترنت دابل آپ از طریق دستورات خط فرمان

قرار دادن یک کنترل داخل منو

قرار دادن شورتکات یک برنامه بر روی Desktop و Start Menu

فقط یک نمونه از برنامه بتواند اجرا شود (با استفاده از Mutex)

فرستادن پارامتر به برنامه email ویندوز

فقط یک نمونه از برنامه بتواند اجرا شود (با استفاده از Process)

غیرفعال کردن دکمه Close یک فرم

عوض کردن نشانگر ماوس (cursor) روی یک کنترل

عوض کردن تصویر پشت زمینه desktop با کدنویسی

فراخوانی یک رویداد از رویداد دیگر

فارسی کردن کیبورد

عدم نیاز به نوشتن انواع توابع سازنده (ctor) در دات نت فریمورک ۳.۰ به بعد

طریقه تغییر دادن خاصیت FontStyle یک کنترل در زمان اجرا

نوشتن یک Event برای تعدادی کنترل و انجام کار مشابه بر روی آنها

نوار تبلیغاتی

ضبط (رکورد) صدا از طریق میکروفون

وادر کردن لیست باکس به اسکرول تا یک آیتم مشخص

نمایش تصاویر در Windows picture and fax viewer از طریق کدنویسی

معرفی کلاسی با سه متود برای جستجو در EventLog

محو شدن تدریجی یک فرم با تغییر دادن خاصیت Opacity

محدود کردن تکست باکس به پذیرش اعداد ۹-۰

معرفی روشی بهینه جهت بدست آوردن درخت کامل یک دایرکتوری

محاسبه زمان اجرای قسمتی برنامه

گرفتن و تسخیر کردن (Capture) تصویر صفحه نمایش

محاسبه ی حجم کلی یک دایرکتوری

کشیدن نقاشی با توابع ساده گرافیکی و ذخیره خروجی در یک فایل

کشیدن لایه شفاف با رنگ دلخواه بر روی عکس

کار کردن با Start-Up ویندوز در C#.net ???!

کلاس hatchBrush

کپی کامل یک دایرکتوری و تمامی زیرشاخه های آن بصورت بازگشتی

یافتن مقادیر ماکزیمم و مینیمم Primitive Type های عددی

کپی کردن فایل های درون یک دایرکتوری (فقط فایلها)

گرفتن کد اسکی و یونیکد کاراکتر ها

گرد کردن و رُند کردن یک مقدار اعشاری

اعداد فارسی در TextBox

غیر فعال کردن یک رویداد در زمان اجرا

هرس کردن یک رشته متنی

نحوه استفاده کاراکترهای خاص (مثلا علامت کپی رایت)

استفاده از Place Holder ها در کار با رشته ها

روش بررسی خالی بودن یک رشته

خوندن خط به خط یک رشته چند سطری

شمارش تعداد آیتم های ایست باکس

نحوه برداشتن تیک از آیتم های کنترل checkedListBox

حالت های مختلف نمایش تاریخ و زمان با کمک String Formatting

جلوگیری از اجرای چندین نمونه از برنامه به صورت همزمان

چطور يك حلقه را يك مرحله جلو ببرم؟

تولید رشته های منحصر بفرد

تبدیل درجه به رادیان و بالعکس

تبدیل آرایه از بایتها به یک رشته و بالعکس

تغییر رنگ قسمتی از richbox

تبدیل عدد و رشته به متناظر بولین آنها و بالعکس

تبدیل درجه حرارتها به یکدیگر (سلسیوس، فارنهایت، کلون)

به توان رساندن و جذر گرفتن

بدست آوردن قسمت صحیح یک عدد اعشاری

بدست آوردن حروف تشکیل دهنده ی یک رشته

بدست آوردن تک تک مقادیر از یک رشته ی مرکب مرزبندی شده

به دست آوردن تعداد خطوط متن در یک textbox

آغاز کار با کلاس Clipboard و دادن و گرفتن یک رشته متنی به/از آن

ایجاد رشته ای تصادفی از اعداد با طول مشخص

بدست آوردن تعداد خطوط یک رشته

کار با ایونت KeyPress در dataGridView

TextChanged در dataGridView

ایجاد متن و رشته های چند سطری

چرا و کی باید از کلاس StringBuilder به جای string استفاده کنیم ؟

تبدیل عکس به آیکن.....

تغییر زبان فرم های ویندوزی از فارسی به انگلیسی و از انگلیسی به فارسی

پیدا کردن معکوس یک رشته بصورت بازگشتی

چطور میشه کاری کرد که نشه توی TextBox فارسی نوشت و کلید های سوئیچی alt+shift هم بی تاثیر باشه

چک کردن ورودی کاربر برای مقادیر عددی

نمایش قرمز رنگ عبارت مورد جستجو در RichTextBox

ایجاد فرم‌های غیر مستطیلی

تبدیل اعداد به رشته

Register A Global HotKey

کپی آیتم‌های یک کامیو باکس به دیگری بدون استفاده از حلقه و در یک خط

استفاده از دیکشنری:

محاسبه ی فاصله بین دو زمان مشخص

کپی آیتم‌های یک کامیو باکس به دیگری

بدست آوردن زبان های نصب شده در سیستم:

عمل چک باکس مثل رادیو باتون

ایجاد محدودیت کاراکتری با حداقل کد(۲)

چینش تصادفی اعضای يك لیست یا آرایه به کمک LINQ

ایجاد محدودیت کاراکتری با حداقل کد

افزودن قطعه کدی به Toolbox

مثالی ساده از LINQ ، که از یک آرایه اعداد زوج را انتخاب میکند :

اینترفیس INotifyPropertyChanged برای ایجاد یک رویداد ژنریک جهت تشخیص پروپرتی تغییر یافته

اضافه کردن شماره سطر به دیتا گرید ویو

محاسبه توان بدون استفاده از حلقه های تکرار و بازگشت

تکست باکسی که فقط عدد قبول میکند

پاک کردن هم زمان تمام TextBox ها :

حل مشکل رسم اشکال گرافیکی در سی شارپ:

تولید رشته های تصادفی بصورت کاملاً دستی

نشان دادن کامل تاریخ و زمان سیستم با متود To String:

انتخاب و فعال شدن یک سطر در dataGridView با مقدار خاص

تست ارتباط با اینترنت از طریق پینگ کردن

بدست آوردن لیست برنامه های نصب شده بر روی سیستم

ارسال و اجرای مستقیم کوئری در LINQ To SQL

تشخیص روشن یا خاموش بودن کلید های CapsLock, NumLock, ScrollLock به روشی بسیار بسیار ساده

بدست آوردن سطر جاری در دیتاگریڈویو

بدست آوردن میزان زمان روشن بودن کامپیوتر

اجرای يك متود تنها با دانستن نام آن (بصورت رشته)

فقط یک نمونه از یک فرم اجرا شود:

تبدیل مینا با استفاده از Convert.ToString

ترسیم و نوشتن مستقیم بر دسکتاپ ویندوز

آموزش کار کردن با BackgroundWorker بصورت آسان و قدم به قدم

آشنایی با الگوهای طراحی : Singleton Pattern

تبدیل اتوماتیک دکمه Enter به Tab جهت انتقال فوکوس در کنترلها

تعدادی کد اعتبارسنجی (Validation) برای TextBox

اطلاعاتی در مورد موس

فراخوانی یک رویداد از رویداد دیگر

ساختن Connection String به روشی بسیار ساده

فقط یک نمونه از برنامه بتواند اجرا شود (با استفاده از WMI)

دستیابی به تاریخ شمسی با استفاده از PersianCalender

تبدیل رشته حاوی تاریخ شمسی به متناظر میلادی آن

آغاز کار با کامپوننت ErrorProvider

برعکس کردن ترتیب آیتمهای یک آرایه

Hash کردن یک رشته متنی با الگوریتم MD5

جلوگیری از رخ دادن Unhandled Exceptions

جایگزین کردن یک رشته درون متن یک textbox به روش اندیس دهی

ارسال تعداد متغیر پارامتر به یک متود

به دست آوردن مقدار یک Enum از عدد متناظر آن

کار با اشیاء بوسیله ی نام آنها (با کمک از Reflection)

refresh کردن صفحه نمایش (desktop)

روشی برای تشخیص عدد بودن یک string

اختصاص دادن رویداد به کترهای ساخته شده در runtime

```
private void Form1_Load(object sender, EventArgs e)
{
    List<Button> buttons = new List<Button>();

    for (int i = 0; i < 10; i++)
    {
        Button btn = new Button();

        btn.Size = new Size(120, 22);

        btn.Text = string.Format("Button{0}", i);

        btn.Location = new Point(10, i * 25 + 10);

        btn.Click += new EventHandler(btn_Click);

        buttons.Add(btn);
    }

    this.Controls.AddRange(buttons.ToArray());
}
```

```
void btn_Click(object sender, EventArgs e)
{
    for (int i = 0 ; i < 10 ; i++)
    {
        Button btn = sender as Button;

        btn.Location = new Point(btn.Left + i, btn.Top);

        Application.DoEvents();
    }
}
```

```
System.Diagnostics.Process.Start("mspaint.exe");
```

اجرا کردن یک فایل اجرایی با کدنویسی

چنانچه فایل اجرایی نیاز به آرگومان خط فرمان داشته باشد :

```
System.Diagnostics.Process.Start("mspaint.exe", "c:\\Test.bmp");
```

مثال بالا تصویر Test را در msPaint باز میکند.

اتصال به بانکهای اطلاعاتی اوراکل در سی شارپ

روشهای اتصال به بانکهای اوراکل :

Oracle .NET data provider

OLE DB .NET data provider

ODBC .NET data provider

در مثال زیر به تمام راههای بالا اشاره شده است.

```
using System;

using System.Data.OracleClient;

using System.Data.OleDb;

using System.Data.Odbc;

namespace ConnectOracle
{
    class Program Interprocess communication (IPC)
    {
        static void Main(string[] args)
        {
            // Connect using .NET data provider for Oracle

            string oracleConnectionString =

                "Data Source=ORCL;User Id=hr;Password=password;";

            using (OracleConnection connection =

                new OracleConnection(oracleConnectionString))
            {
                connection.Open( );

                // Return some information about the server.

                Console.WriteLine("---Microsoft .NET Provider for Oracle---");
            }
        }
    }
}
```

```

        Console.WriteLine("ConnectionString = {0}\n", oracleConnectionString);

        Console.WriteLine("State = {0}", connection.State);

        Console.WriteLine("DataSource = {0}", connection.DataSource);

        Console.WriteLine("ServerVersion = {0}", connection.ServerVersion);
    }

    // Connect using .NET data provider for OLE DB.

    string oledbConnectionString =

        "Provider=MSDAORA;Data Source=ORCL;User Id=hr;Password=password;";

    using (OleDbConnection connection =

        new OleDbConnection(oledbConnectionString))
    {

        connection.Open( );

        // Return some information about the server.

        Console.WriteLine("\n---Microsoft .NET Provider for OLE DB---");

        Console.WriteLine("ConnectionString = {0}\n", oledbConnectionString);

        Console.WriteLine("State = {0}", connection.State);

        Console.WriteLine("DataSource = {0}", connection.DataSource);

        Console.WriteLine("ServerVersion = {0}", connection.ServerVersion);
    }

    // Connect using .NET data provider for ODBC

    string odbcConnectionString = "Driver={Oracle in OraDb10g_home1};" +

        "Server=ORCL;uid=hr;pwd=password;";

    using (OdbcConnection connection =

        new OdbcConnection(odbcConnectionString))
    {

        connection.Open( );

        // Return some information about the server.

```

```

        Console.WriteLine("\n---Microsoft .NET Provider for ODBC---");

        Console.WriteLine("ConnectionString = {0}\n", odbcConnectionString);

        Console.WriteLine("State = {0}", connection.State);

        Console.WriteLine("DataSource = {0}", connection.DataSource);

        Console.WriteLine("ServerVersion = {0}", connection.ServerVersion);

    }

    Console.WriteLine("\nPress any key to continue.");

    Console.ReadKey( );

}

}

}

```

روش اول Oracle .NET data provider یا همان Native Oracle:

Oracle .NET data provider با استفاده از OCI (Oracle Call Interface) و از طریق ابزارهای ارتباطی برای کلاسیک (Oracle client connectivity software) به بانکهای اطلاعاتی اوراکل دسترسی دارد. این پروایدر به اوراکل ۷.۳.۴ (یا ورژنهای بالاتر) می تواند دسترسی داشته باشد، همچنین به Oracle 8i Release 3 (8.1.7) client software (یا ورژنهای بالاتر) نیاز دارد. تمام کلاسهای مورد نیاز در فضای نام System.Data.OracleClient قرار دارد

مثالی از رشته اتصال که از Integrated Security استفاده کرده است:

```
Data Source = ORCL; Integrated Security = yes;
```

و بدون استفاده از Integrated Security:

```
Data Source=ORCL;User Id=hr;Password=password;
```

معمولاً روش Native بهتر از دو روش دیگر است. چون این روش مخصوص این نوع بانک اطلاعاتی است و برای آن بهینه شده است (تمام لایه های غیر مرتبط با این نوع دیتا حذف شده است)

روش دوم OLE DB:

در این روش برای دسترسی به بانکهای اطلاعاتی اوراکل از MSDAORA (Oracle OLE DB provider) استفاده می کنیم.

```
Provider=MSDAORA;Data Source=myOracleDb;User Id=hr;Password=password;
```

معمولاً از این روش برای ایجاد یک پل ارتباطی با برنامه ای که از OLE DB استفاده می کند، بکار می رود.

روش سوم OLE DB:

و در پایان با استفاده از ODBC .NET data provider نیز می توان به بانکهای اوراکل متصل شد. به عنوان مثال رشته اتصال به بانک اطلاعاتی در Oracle 10g به صورت زیر است.

```
"Driver={Oracle in OraDb10g_home1};Server=ORCL;uid=hr;pwd=password;"
```

از این روش نیز برای ایجاد یک پل ارتباطی با برنامه ای که از ODBC استفاده می کند، بکار می رود.

بنابراین در صورت امکان برای اتصال به این بانکها از روش Native استفاده کنید.

نکته: اوراکل از یک فایل پیکر بندی (با نام TNSNAMES.ORA در محل بانک اطلاعاتی) برای تعیین نحوه ارتباط با آن استفاده می کند.

مثالی از محتویات این فایل برای alias ORCL :

```
ORCL =  
  
  (DESCRIPTION =  
  
    (ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.1.100)(PORT = 1521))  
  
    (CONNECT_DATA =  
  
      (SERVER = DEDICATED)  
  
      (SERVICE_NAME = orcl)  
  
    )  
  
  )
```

که این اتصال به طور پیش فرض از ارتباط TCP/IP با پورت ۱۵۲۱ صورت می گیرد.

اتصال به **SqlServer** با استفاده از IP Address

```
using System;  
  
using System.Data.SqlClient;  
  
namespace ConnectIPAddressSqlServer  
{  
  
    class Program  
    {  
  
        static void Main(string[] args)  
        {  
  
            string connectionString =  
  
                "Network Library=dbmssocn;Network Address=127.0.0.1;" +  
  
                "Integrated security=SSPI;Initial Catalog=AdventureWorks";  
  
  
            using (SqlConnection connection = new SqlConnection(connectionString))  
            {  
  
                connection.Open( );  
  
            }  
  
        }  
  
    }  
  
}
```

```

// Return some information about the server.

Console.WriteLine(

    "ConnectionState = {0}\nDataSource = {1}\nServerVersion = {2}",

    connection.State, connection.DataSource,

    connection.ServerVersion);

}

Console.WriteLine("\nPress any key to continue.");

Console.ReadKey( );

}

}

}

```

SQL Server network libraries فایل‌های dll هستند که عملیات شبکه مورد نیاز برای کلاینتها را انجام می دهند.

پروتکلها شبکه موجود برای SQL Server

- Shared Memory

- TCP/IP

استفاده از پروتکل TCP/IP برای ارتباط

- Named Pipes

SQL Server مکانیسم IPC را برای ارتباط بین کلاینت و سرور فراهم می کند.

- VIA

پروتکل Virtual Interface Adapter که توسط سخت افزارهای VIA مورد استفاده قرار می گیرد. از SQL Server 2005 به بعد دیگر این پروتکل پشتیبانی نمی شود.

- AppleTalk ADSP

- Banyan VINES

- Multiprotocol

به صورت اتوماتیک از بین پروتکل های موجود یکی را انتخاب می کند که از لحاظ کارایی همانند استفاده از native network library می باشد. که از پروتکل های TCP/IP Sockets, NWLink IPX/SPX, Named Pipes پشتیبانی می کند

- NWLink IPX/SPX

پروتکل شبکه های ناول

برای اطلاعات بیشتر در مورد پیکربندی پروتکل های شبکه به Microsoft SQL Server Books Online مراجعه کنید

استفاده از سوکت های TCP/IP برای SQL Server عملکرد و مقیاس پذیری برنامه را بهبود می بخشد (در زمانی که با حجم بالایی از داده سرو کار داریم). این مکانیسم از بعضی از مسائل امنیتی مربوط به named pipes اجتناب می ورزد.

در صورت استفاده از هر پروتکلی ، کلاینت و سرور با ید برای استفاده از TCP/IP پیکر بندی شوند.

SQL Server Configuration Manager/SQL Server Network Configuration/enabling TCP/IP in the Protocols
.subnode

توجه کنید که Instance پیش فرض SqlServer از پورت ۱۴۳۳ استفاده می کند. در صورتی که شما آنرا برای پورت دیگری پیکربندی کرده باشید Connection String را به صورت زیر تغییر دهید.

Network Address=(local),1450

Logoff، Restart، Shutdown کردن یک کامپیوتر بوسیله کد

```
[DllImport("user32.dll")]
public static extern int ExitWindowsEx(int uFlags, int dwReason);

// Log Off
ExitWindowsEx(0, 0);

// Reboot
ExitWindowsEx(2, 0);

// Shutdown
ExitWindowsEx(1, 0);

// Force LogOff
ExitWindowsEx(4, 0);

// Log Off
System.Diagnostics.Process.Start("ShutDown", "/l");

// Reboot
System.Diagnostics.Process.Start("ShutDown", "/r");

// Shutdown
System.Diagnostics.Process.Start("ShutDown", "/s");
```

Serialize کردن یا سریالیزه کردن آبجکت و فایل

```

using System.IO;

using System.Runtime.Serialization;

using System.Runtime.Serialization.Formatters.Binary;

public void SerializeObject(string filename, Object o)
{
    Stream stream = File.Open(filename, FileMode.Create);

    BinaryFormatter bFormatter = new BinaryFormatter();

    bFormatter.Serialize(stream, o);

    stream.Close();
}

```

Serialize کردن object و XML

```

XmlSerializer xmSer = new XmlSerializer(typeof(Person));

FileStream st = new FileStream("C:\\Test.xml", FileMode.OpenOrCreate);

Person p = new Person("Ali Rezaei", 44);

xmSer.Serialize(st, p);

st.Close();

```

Serialize کردن object و SOAP XML

برای Serialize کردن object تون توی فایل Soap می تونین از این کد استفاده کنین.

دقت کنین که باید System.Runtime.Serialization.Formatters.Soap رو Add Reference و using کنین.

```

SoapFormatter soap = new SoapFormatter();

FileStream st = new FileStream("C:\\Test.soap", FileMode.OpenOrCreate);

Person p = new Person("Ali Rezaei", 33);

soap.Serialize(st, p);

st.Close();

```

restart کردن (بستن و مجددا اجرا کردن) برنامه

```

private void button1_Click(object sender, EventArgs e)

```

```
{  
  
    Application.Restart();  
  
}
```

Resize کردن یک تصویر با کد نویسی

```
using System.Drawing.Drawing2D;
```

```
private static Image resizeImage(Image imgToResize, Size size)
```

```
{  
  
    int sourceWidth = imgToResize.Width;  
  
    int sourceHeight = imgToResize.Height;  
  
  
    float nPercent = 0;  
  
    float nPercentW = 0;  
  
    float nPercentH = 0;  
  
  
    nPercentW = ((float)size.Width / (float)sourceWidth);  
  
    nPercentH = ((float)size.Height / (float)sourceHeight);  
  
  
    if (nPercentH < nPercentW)  
        nPercent = nPercentH;  
  
    else  
        nPercent = nPercentW;  
  
  
    int destWidth = (int)(sourceWidth * nPercent);  
  
    int destHeight = (int)(sourceHeight * nPercent);  
  
  
    Bitmap b = new Bitmap(destWidth, destHeight);  
  
    Graphics g = Graphics.FromImage((Image)b);  
  
    g.InterpolationMode = InterpolationMode.HighQualityBicubic;
```

```

        g.DrawImage(imgToResize, 0, 0, destWidth, destHeight);

        g.Dispose();

        return (Image)b;
    }

```

خواندن از Registry ویندوز

با سلام

یه نکته کوچک، گرچه شاید خیلی از دوستان بدونند.

برای خواندن از رجیستری ویندوز می تونید از کد زیر استفاده کنید :

فرض کنید اطلاعات زیر در رجیستری موجود باشد :

```

[HKEY_CURRENT_USER\Software\TAW\BSE]

"DSN"="TAWReports"

"User"="TAW1"

"Password"="taw1.1"

"Server"="dbserver"

"IP"=" "

```

حال برای خواندن :

```

namespace CONAPP
{
    using System;

    using Microsoft.Win32;

    public class Class1
    {
        public Class1()
        {
            //
            // TODO: Add Constructor Logic here
            //
        }
    }
}

```

```

public static int Main(string[] args)
{

RegistryKey SUBKEY;

RegistryKey TAWKAY = RegistryKey.OpenRemoteBaseKey(Microsoft.Win32.RegistryHive.CurrentUser, "");

string subkey = "Software\\TAW\\BSE";

SUBKEY = TAWKAY.OpenSubKey(subkey);

object dsn = SUBKEY.GetValue("DSN");

object user = SUBKEY.GetValue("user");

object password = SUBKEY.GetValue("password");

object server = SUBKEY.GetValue("server");

return 0;

}

}

}

```

combobox کردن readonly

```

private void comboBox1_KeyPress(object sender, KeyPressEventArgs e)
{

e.Handled = true;

}

```

Ping کردن یک کامپیوتر تحت شبکه

```

using System.Net.NetworkInformation;

{

string remoteMachineNameOrIP = "127.0.0.1";

int timeout = 5;

Ping ping = new Ping();

// Pinging remote machine

PingReply reply = ping.Send(remoteMachineNameOrIP, timeout);

```

```
// Displaying the result

StringBuilder sb = new StringBuilder();

sb.Append("Address:" + reply.Address.ToString());

sb.Append("\nStatus:" + reply.Status.ToString());

sb.Append("\nRoundtripTime:" + reply.RoundtripTime.ToString());

MessageBox.Show(sb.ToString(), "Ping Result: " + remoteMachineNameOrIP);

}
```

picture box

میخواهم با کلیک بر روی ایتیم های لیست باکس (که آدرس عکس هستند) در picture box نمایش داده شود

```
PictureBox1.ImageLocation=listbox1.selecteditem.toString();
```

یکی دیگه:

```
PictureBox1.Image = Image.FromFile(ListBox1.SelectedItem.ToString)
```

Map کردن در ایونهای شبکه به کمک اجرای دستورات خط فرمان

```
using System.Diagnostics;
```

```
Process.Start("NET", @"USE U: \\127.0.0.1\Sinpin /PERSISTENT:YES");
```

Initilize کردن کالکشنها در انت ۳.۰ به بالا

میتونید کالکشنها رو در همان زمان تعریف مقداردهی نیز کنید.

مثال ۱-

```
List<string> names = new List<string> { "Jim", "Joe", "Bob", "Sam" };
```

مثال ۲- با فرض داشتن کلاسی مانند:

```
public class Person
{
    public string FirstName { get; set; }
    public string LastName { get; set; }
    public int Age { get; set; }
}
```

میتونیم بنویسیم :

```
List<Person> people = new List<Person> {
```

```

new Person { FirstName = "Scott", LastName = "Guthrie", Age = 32 },

new Person { FirstName = "Bill", LastName = "Gates", Age = 50 },

new Person { FirstName = "Susanne", LastName = "Guthrie", Age = 32 }

};

```

Standby و Hibernate کردن کامپیوتر

```

// Hibernate

Application.SetSuspendState(PowerState.Hibernate, true, true);

// Standby

Application.SetSuspendState(PowerState.Suspend, true, true);

```

فایل swf

اگر می‌خواهید این فایل فلش رو اضافه کنید فکر کنید این راه حل خوبی باشه :

۱. بخش toolbox محیط ویژوال استدیو رو باز کنید (همون منوی سمت چپ)

۲. روی اون کلیک راست کنید و choose item رو انتخاب کنید

۳. حالا به پنجره باز می‌شه به اسم choose toolbox items که شما tab سمت راستی یعنی com components رو باز می‌کنید.

۴. حالا توی لیست موجود بگردید دنبال گزینه shock wave flash object رو تیک بزنید و ok کنید.

۵. حالا پروژه رو rebuild کنید.

۶. حالا این کامپوننت توی toolbox در سمت چپ اضافه شده است.

۷. drag and drop و قضیه حله (اگر "Failed to import the ActiveX control. Please ensure it is properly registered" این error رو بهتون داد دوباره rebuild کنید. حالا اضافه می‌شه)

۸. حالا فایل فلش رو اضافه کنید.

با این کد:

```
axShockwaveFlash1.Movie = Application.StartupPath + @"flash\main.swf";
```

این کد رو می‌تونید توی رویداد load فرم اضافه کنید و یا در رویداد کلیک یک button.

که در این کد @"flash\main.swf" آدرس فایل فلشتون هست : flash پوشه ای هست که من در فولدر debug پروژه ساختم و فایل های فلشم را در اون کپی کردم. و main.swf هم نام اون فلشی هست که می‌خواهم نمایش بدم. تمام شد...

Anchor و Dock چیستند؟

Dock : آیا تا به حال منوها و Toolbar ها را دیده اید که به گوشه ای از فرم چسبیده اند... این همان خاصیت است که به ما امکان می دهد اشیاء را در حالت Dock نماییم ... در این حالت ها شیء به گوشه بالا , پایین , چپ , راست یا در کل شیء پدر ظاهر خواهد شد ... البته برای اشیایی مانند Label باید حتماً AutoSize را False نمایید تا بتوانید به طور کامل از امکانات Dock بهره ببرید.

Anchor : این گزینه به ما امکان ست کردن شیء را همراه با شیء پدر خواهد داد به طوری که اگر ابعاد شیء پدر تغییر کند موقعیت و سایز آن شیء فرزند هم می تواند تغییر نماید ... به صورت کلی هر ضلع شیء به ضلع شیء پدر وصله خواهد شد ...

منظور از شیء پدر هم می تواند : Table , Panel , GroupBox , Form و یا هر شیء ای که امکان پدر و فرزندی را به ما خواهد داد ..

DeSerialize کردن باینری آبجکت از فایل

```
using System.IO;

using System.Runtime.Serialization;

using System.Runtime.Serialization.Formatters.Binary;

public object DeserializeObject(string filename)
{
    object o;

    Stream stream = File.Open(filename, FileMode.Open);

    BinaryFormatter bFormatter = new BinaryFormatter();

    o =
        (ObjectToSerialize)bFormatter.Deserialize(stream);

    stream.Close();

    return o;
}
```

DeSerialize کردن object از XML

برای بازیابی شیء تون از یک فایل XML می تونین از این کد استفاده کنین:

```
XmlSerializer xmSer = new XmlSerializer(typeof(Person));

FileStream st = new FileStream("C:\\Test.xml", FileMode.Open);

Person p = xmSer.Deserialize(st) as Person;

st.Close();
```

DeSerialize کردن object از SOAP XML

برای بازیابی شیء تون از یک فایل SOAP می تونین از این کد استفاده کنین:

```
SoapFormatter soap = new SoapFormatter();
```



```

FileStream st = new FileStream("C:\\Test.soap", FileMode.Open);

Person p = soap.Deserialize(st) as Person;

st.Close();

```

Crop کردن یک تصویر (یک برش مستطیلی از تصویر)

```

private static Image cropImage(Image img, Rectangle cropArea)
{
    Bitmap bmpImage = new Bitmap(img);

    Bitmap bmpCrop = bmpImage.Clone(cropArea,
    bmpImage.PixelFormat);

    return (Image)(bmpCrop);
}

```

مثال از نحوه ی استفاده :

```

private void button1_Click(object sender, EventArgs e)
{
    pictureBox2.Image = cropImage(pictureBox1.Image, new Rectangle(10,10,100,100));
}

```

API SndPlaySound dar C# Mamnoon

```
[DllImport("winmm.dll")] public int sndPlaySound(string path, int flags) {}
```

```
const int SND_ASYNC = 0x1
```

```
const int SND_FILENAME = 0x20000
```

```
// And to play the sound:
```

```
sndPlaySound("C:\\MySound.wav", SND_ASYNC | SND_FILENAME);
```

حاشیه دار کردن کنترل TextBox با یک رنگ دلخواه

در کلاس TextBox خود (و یا سایر کنترل های دلخواه : کامبویاکس، لیست باکس و ...) بنویسید :

```
private static int WM_NCPAINT = 0x0085;
```

```
private static int WM_ERASEBKGDND = 0x0014;
```

```

private static int WM_PAINT = 0x000F;

[DllImport("user32.dll")]
static extern IntPtr GetDCEx(IntPtr hwnd, IntPtr hrgnclip, uint fdwOptions);

[DllImport("user32.dll")]
static extern int ReleaseDC(IntPtr hwnd, IntPtr hdc);

protected override void WndProc(ref Message m)
{
    base.WndProc(ref m);

    if (m.Msg == WM_NCPAINT || m.Msg == WM_ERASEBKGD || m.Msg == WM_PAINT)
    {
        IntPtr hdc = GetDCEx(m.HWnd, (IntPtr)1, 1 | 0x0020);

        if (hdc != IntPtr.Zero)
        {
            Graphics graphics = Graphics.FromHdc(hdc);

            Color borderColor = Color.Blue;

            Rectangle rectangle = new Rectangle(0, 0, this.Width, this.Height);

            ControlPaint.DrawBorder(graphics, rectangle, borderColor, ButtonBorderStyle.Solid);

            m.Result = (IntPtr)1;

            ReleaseDC(m.HWnd, hdc);
        }
    }
}

```

توجه داشته باشید که انجام اینکار صرفاً بوسیله override کردن رویداد paint و رسم یک مستطیل امکانپذیر نیست.

محوشدن تدریجی یک فرم با تغییر دادن خاصیت Opacity

```

private void button1_Click(object sender, EventArgs e)
{
    this.Opacity = 1;
}

```

```

for (int i = 0; i < 100; i++)
{
    this.Opacity -= 0.01;
    Application.DoEvents();
}
}

```

انتقال آیتمهای یک آرایه از اعداد به یک لیست باکس

```

int[] numbers = { 12, 23, 34, 45, 56, 67 };
Object[] oNumbers = new Object[numbers.Length];
numbers.CopyTo(oNumbers, 0);
listBox1.Items.AddRange(oNumbers);

```

پیاده سازی حالت ساده ی الگوی سینگلتون در یک کلاس

```

sealed class SingletonClass
{
    // Static members are lazily initialized.
    // .NET guarantees thread safety for static initialization
    private static readonly SingletonClass instance =
        new SingletonClass();

    // Note: constructor is private.
    private SingletonClass(){}

    public static SingletonClass GetInstance()
    {
        return instance;
    }
}

```

رسم یک چند ضلعی توپر روی یک فرم

```
private void Form1_Paint(object sender, PaintEventArgs e)
{
    Graphics g = e.Graphics;

    // Create pen.
    Pen blackPen = new Pen(Color.Black, 3);

    // Create points that define polygon.
    Point point1 = new Point(30, 50);
    Point point2 = new Point(100, 25);
    Point point3 = new Point(200, 5);
    Point point4 = new Point(250, 50);
    Point point5 = new Point(270, 100);
    Point point6 = new Point(250, 250);

    Point[] curvePoints = {point1, point2, point3, point4, point5, point6};

    // Draw polygon to screen.
    g.DrawPolygon(blackPen, curvePoints);

    // Fill polygon
    g.FillPolygon(Brushes.Red, curvePoints);
}
```

کشیدن نقاشی باتابع ساده گرافیکی و ذخیره خروجی در یک فایل

```
public void Draw2File ()
{
    Bitmap bmp = new Bitmap (500, 400);

    Graphics gr = Graphics.FromImage (bmp);

    Rectangle rect = new Rectangle (20, 20, 450, 350);

    gr.DrawRectangle (Pens.Red, rect);

    gr.FillRectangle(Brushes.Pink, rect);

    gr.DrawString ("Salam", new Font ("Arial", 10), Brushes.Red, 50, 50);

    gr.DrawArc (Pens.Red, 30, 30, 60, 60, 0, 180);

    gr.DrawEllipse(new Pen(Color.Yellow, 3), new Rectangle(200, 100, 200, 200));
}
```

```

bmp.Save ("C:\\1.bmp");

bmp.Dispose ();

gr.Dispose ();

}

```

ویرایش فقط یک سطر از DataGridView

```

{
for (int i = 0; i < dataGridView1.RowCount; i++)
{
dataGridView1.Rows[i].ReadOnly = true;
}
}

private void button1_Click(object sender, EventArgs e)
{
for (int i = 0; i < dataGridView1.RowCount; i++)
{
dataGridView1.Rows[i].ReadOnly = true;
}

dataGridView1.Rows[dataGridView1.CurrentRow.Index].ReadOnly = false;
}
}

```

انجام عملیات متداول روی فایلها و دایرکتوریها

```

using System.IO;

//To create a directory
Directory.CreateDirectory(@"C:\MyNewDir");

//To move a directory
Directory.Move(@"C:\MyNewDir", @"C:\MyMovedDir");

//To delete a directory
Directory.Delete(@"C:\MyMovedDir");

```

```

//To Delete a directory recursively

Directory.Delete(@"C:\MyNewDir", true);


//To Delete a File

File.Delete(@"C:\MyFile.Txt");


//To Move a File

File.Move(@"C:\MyFile.Txt", @"C:\MyOtherDir\MyFile.Txt");


//To Copy a file

File.Copy(@"C:\MyFile.Txt", @"C:\MyOtherDir\MyFile.Txt");


//To copy to a different file name is also possible

File.Copy(@"C:\MyFile.Txt", @"C:\MyOtherDir\MyNewFileName.Txt");


//To get information about a file, like the length

//You can also get the extension, directory, LastAccessedtime,

//LastModifiedTime, whether the file exists or not, the creation date,

//attributes of the file etc, from the FileInfo class

FileInfo FI = new FileInfo(@"C:\MyFile.Txt");

Console.WriteLine("File size of MyFile.Txt: {0}", FI.Length);


//copy example

String DateTemp = DateTime.Now;

File.Copy(@"P:\PRD\Products\AHM\prod.CD\Database\data.mdb",

    @"P:\PRD\Products\AHM\prod.CD\Database\"+ DateTemp + "-data.mdb");


using Microsoft.Win32;


// Create SubKey

```

انجام عملیات متداول در رجیستری ویندوز

```

Registry.LocalMachine.CreateSubKey(@"Software\Sinpin", RegistryKeyPermissionCheck.ReadWriteSubTree);

//Create Key and Set Value

RegistryKey reg = Registry.LocalMachine.OpenSubKey(@"Software\Sinpin", true);

reg.SetValue("DWord", "1", RegistryValueKind.DWord);

reg.SetValue("ExpandString", "1", RegistryValueKind.ExpandString);

reg.SetValue("QWord", "1", RegistryValueKind.QWord);

reg.SetValue("String", "1", RegistryValueKind.String);

reg.SetValue("Unknown", "1", RegistryValueKind.Unknown);

// Delete Key

reg.DeleteValue("DWord");

// Delete SubKey

Registry.LocalMachine.DeleteSubKey(@"Software\Sinpin");

// Read Key Value

string val = reg.GetValue("QWord").ToString();

// Retrieve All Keys

foreach (string s in reg.GetValueNames())

    MessageBox.Show(s);

```

انجام عملیات متداول بر رشته‌ها

حذف تمامی whiteSpace ها :

```

//STRIPS WHITE SPACES FROM BOTH START + FINISH

string Name = " String Manipulation " ;

string NewName = Name.Trim();

```

حذف whiteSpace ها (با حروفی خاص) از آخر رشته :

```

//STRIPS CHRS FROM THE END OF THE STRING

string Name = " String Manipulation " ;

//SET OUT CHRS TO STRIP FROM END

```

```
char[] MyChar = { ' ', 'n' };
string NewName = Name.TrimEnd(MyChar);
```

حذف whiteSpace ها (با حروفی خاص) از ابتدای رشته :

```
//STRIPS CHRS FROM THE START OF THE STRING
string Name = " String Manipulation " ;
//SET OUT CHRS TO STRIP FROM END
char[] MyChar = { ' ', 'S' };
string NewName = Name.TrimStart(MyChar);
```

جستجوی یک رشته ی در رشته ی دیگر :

```
string MainString = "String Manipulation";
string SearchString = "pul";
int FirstChr = MainString.IndexOf(SearchString);
```

جایگزین کردن بخشی از یک رشته :

```
string MainString "String Manipulatin";
string CorrectString = MainString.Replace("Manipulatin", "Manipulation");
```

حذف کردن بخشی از یک رشته :

```
string MainString = "Sllllltring Manipulation";
string NewString = MainString.Remove(1,4);
```

تفکیک یک رشته :

```
string MainString = "String Manipulation";
string [] Split = MainString.Split(new Char [] { ' ' });
//SHOW RESULT
MessageBox.Show(Convert.ToString(Split[0]));
MessageBox.Show(Convert.ToString(Split[1]));
```

انتقال آیتمهای Enum به یک ListBox

```
public enum Days { Sat = 1, Sun, Mon, Tue, Wed, Thu, Fri };
System.Type t = typeof(Days);
foreach (string s in Enum.GetNames(t))
{
    listBox1.Items.Add(s);
}
```


افزودن یک زیر منو دستورده نخواهد به context menu ویندوز اکسلور

```
private void btnAddMenu_Click(object sender, System.EventArgs e)
{
    RegistryKey regmenu = null;
    RegistryKey regcmd = null;

    try
    {
        regmenu = Registry.ClassesRoot.CreateSubKey(MenuName);

        if(regmenu != null)

            regmenu.SetValue("", this.txtName.Text);

        regcmd = Registry.ClassesRoot.CreateSubKey(Command);

        if(regcmd != null)

            regcmd.SetValue("", this.txtPath.Text);

    }

    catch(Exception ex)
    {

        MessageBox.Show(this, ex.ToString());

    }

    finally
    {

        if(regmenu != null)

            regmenu.Close();

        if(regcmd != null)

            regcmd.Close();

    }

}
```

افزودن و حذف کردن یک برنامه به Startup ویندوز از طریق کلیدهای رجیستری

```
using Microsoft.Win32;
```

```
{
```

```

RegistryKey key =
Registry.LocalMachine.OpenSubKey(@"Software\Microsoft\Windows\CurrentVersion\Run", true);

key.SetValue(_name, _path);

}

```

```

private void RemoveStartUpKey(string _name)

{

RegistryKey key =
Registry.LocalMachine.OpenSubKey(@"Software\Microsoft\Windows\CurrentVersion\Run", true);

key.DeleteValue(_name, false);

}

```

افزودن کنترل در زمان اجرا

مثال: یک تکست باکس به فرم اضافه میکند:

```

private void button1_Click(object sender, System.EventArgs e)

{

TextBox myText = new TextBox();

myText.Location = new Point(25,25);

this.Controls.Add (myText);

}

```

اعمال محدودیت برای ذخیره فایل توسط کاربر

مثلا به برنامه داریم که به کاربر قابلیت ذخیره ی فایل میدهد !

خوب بدلیل شرایط امنیتی بهتره دسترسی کاربر به درایو اصلی ویندوز محدود بشه ! برای اینکار بصورت زیر عمل میکنیم :

ابتدا Namespace های زیر رو استفاده میکنیم :

```

using System.Security.Permissions;

using System.Security;

using System.IO;

```

حالا قبل از namespace اصلی برنامه :

```

[assembly: FileIOPermissionAttribute(SecurityAction.RequestRefuse, Write = "C:\\")]

```

بدین صورت کاربر اجازه ی درج فایل در درایو C رو نداره .

مثلا اگه تو رویداد Click یکی از Button ها دستور زیر نوشته شده باشه .. کاربر با خطای مواجه خواهد شد :

```

StreamWriter sw;

```

```
try
```

```
{
```

```
sw = File.CreateText("C:\\\\test.txt");
```

```
sw.WriteLine("salam");
```

```
}
```

```
catch
```

```
{
```

```
MessageBox.Show("عدم دسترسی");
```

```
}
```

```
finally
```

```
{
```

```
sw.Close();
```

```
}
```

اعتبار سنجی آدرسهای ایمیل به کمک عبارات منظم

```
using System.Text.RegularExpressions;
```

```
public bool TestEmailRegex(string emailAddress)
```

```
{
```

```
string patternStrict = @"^(([^<>()[]\]\\.,;:\s@"
```

```
+ @"(\.["
```

```
+ @"((\[[0-9]{1,3}\.["
```

```
+ @"\.["
```

```
+ @"["
```

```
Regex reStrict = new Regex(patternStrict);
```

```
bool isStrictMatch = reStrict.IsMatch(emailAddress);
```

```
return isStrictMatch;
```

```
}
```

اطلاعاتی در مورد موس

شاید نیاز داشته باشید بفهمید کاربر از موس استفاده می‌کند یا نه؛ و در صورتی که استفاده می‌کند اطلاعاتی از موس (تعداد دکمه، Wheel) این اطلاعات و یا سایر اطلاعاتی را که از سیستم کاربر می‌خواهید می‌توانید در کلاس Enviroment بیابید.

```
SystemInformation.MousePresent.ToString()  
  
SystemInformation.MouseButtons.ToString()  
  
SystemInformation.MouseButtonsSwapped.ToString()  
  
SystemInformation.MouseWheelPresent.ToString()
```

اضافه و جدا کردن نام فایل از مسیر کامل

```
string filename = @"..\..\myfile.txt";  
  
string fullPath = @"c:\Temp";  
  
string filename = Path.GetFileName(filename);  
  
string fullPath = Path.Combine(fullPath, filename);  
  
// (fullPath is now "c:\Temp\myfile.txt")
```

استفاده کارآمد از دستور Using برای آزادسازی بهتر منابع

سلام به همه دوستان عزیز

مدیریت منابع و آزادسازی اشیاء یکی از بحث‌های مهم برنامه نویسی است. توصیه می‌کنم حتما مطالب این سرفصل رو پیگیری کنید. به هر حال زمانی که از یک کلاس، شی ساخته میشود حتما پس از استفاده می‌باید آنرا از بین برد و منابع مورد استفاده اش را آزاد کرد. هر چند که وقتی یک کلاس را خراب میکنید در همان لحظه توسط کامپایلر منابعش آزاد نمی‌شود. حال نمونه کد زیر را در نظر بگیرید

```
TextReader reader = new StreamReader(filename);  
  
String line ;  
  
While ((line=reader.ReadLine()) !=null)  
{  
  
Console.WriteLine(line);  
  
}  
  
Reader.close ;
```

این کد یک مشکل اساسی دارد. در واقع در برابر exception ایمن نیست. اگر فراخوانی ReadLine یا WriteLine مشکلی ایجاد کند آنگاه متد close فراخوانی نمیشود.

برای بهینه شدن این کد بدین شکل عمل میشود .

```
TextReader reader = new StreamReader(filename);

Try
{
    String line ;

    While ((line=reader.ReadLine()) !=null)
    {
        Console.WriteLine(line);
    }
}

Finally
{
    Reader.close ;
}
```

با این کد حتی اگر خطایی هم رخ دهد متد Close اجرا خواهد شد .

اما استفاده از Finally چند مشکل دارد .

- ۱ - اگر مجبور باشید چند منبع را آزاد کنید این روش ناکارآمد است .
- ۲ - فهم راه حل مشکل است و هر بار باید تکرار شود .
- ۳ - در خیلی از مواقع لازم است در کد تغییرات زیادی انجام دهید .
- ۴ - ارجاع به منبع حتی بعد از بلاک Finally هم باقی میماند . یعنی ممکن است بعد از آزاد شدن منبع تصادفاً مورد استفاده واقع شود . زیرا که منبع در سراسر برنامه قابل دسترسی است .

اما استفاده از دستور Using برای بهینه شدن این کد .

```
Using ( try variable = initialization ) embeddedStatement
```

این دستور معادل دستورات زیر است :

```
{
    Type variable = initialization ;

    Try
    {
        embeddedStatement
    }

    Finally
}
```

```

{
    If (variable != Null)
    {
        ((Disposable)variable).Dispose();
    }
}

```

پس بهترین راه برای اینکه مطمئن شوید متد Close همیشه فراخوانی شود به شکل زیر است :

```

Using(TextReader reader = new StreamReader(filename)
{
    String line ;
    While((line = reader.ReadLine()) != )
    {
        Console.WriteLine(line);
    }
}

```

مزایای این روش :

۱ - برای آزاد سازی چندین منبع به خوبی کار می کند .

۲ - منطق برنامه را خراب نمی کند

۳- از تکرار جلوگیری میکند .

۴ - بعد از دستور Using دیگر نمیتوان از منبع استفاده کرد . زیرا آن منبع در خارج از محدوده Using قابل دسترسی نیست . (* این مورد بهترین مزیت این دستور است زیرا یکی از پر اشکال ترین موارد در آزاد سازی منابع استفاده تصادفی دوباره از آن منبع است .)

ارسال متغیر به صفحه بعد با استفاده از url

نحوه ارسال به صفحه بعد

```
Response.Redirect("form2.aspx?دلیخواه=" + Convert.ToString("متغیر"));
```

نحوه استفاده از آن در صفحه دوم

```
string str = Request.QueryString.Get("متغیر");
```

ایجاد فرمهایی با اشکال غیر مستطیلی

ایجاد یک فرم دایره ای شکل :

```
private void Form1_Load(object sender, EventArgs e)
```

```

{
    this.Height = 350;

    this.Width = 350;

    //Creating circle path

    System.Drawing.Drawing2D.GraphicsPath path = new System.Drawing.Drawing2D.GraphicsPath();

    path.AddEllipse(0, 0, 300, 300);

    //Creating the region from the circle path

    this.Region = new Region(path);

    this.Show();
}

```

ایجاد طیف رنگ بامش از دو رنگ توسط ColorBlend

```

//define LinearGradientBrush with Init Prop

//

LinearGradientBrush lgBrush = new LinearGradientBrush(this.ClientRectangle,

    Color.White, Color.Blue,130);

//

// Define ColorBlend

//

ColorBlend cb = new ColorBlend(5); // 5 is count of color which sets to Brush

cb.Colors = new Color[] {Color.Red,Color.Blue,Color.Gold,Color.Black,Color.Brown,Color.Lime };

//define the point of each color

//all the Points must be between 0 and 1

// 0 and 1 should be exist in point collection

cb.Positions = new float[] {0.0f,0.2f,0.4f,0.6f,0.8f,1f };

// assign colorBlend to gradientBrush

lgBrush.InterpolationColors = cb;

//draw shape and fill it with gradientBrush

```

```
e.Graphics.FillRectangle(lgBrush, this.ClientRectangle);
```

ایجاد طیف رنگ (Gradient) بعنوان پشت زمینه ی کنترلها

```
using System.Drawing.Drawing2D;
```

```
using System.Drawing;
```

```
public void PaintGradient(Control _control, LinearGradientMode _direction, Color _gradientColorStart, Color _gradientColorEnd)
```

```
{
```

```
    LinearGradientBrush gradBrush;
```

```
    gradBrush = new LinearGradientBrush(new Rectangle(0, 0, _control.Width, _control.Height), _gradientColorStart, _gradientColorEnd, _direction);
```

```
    Bitmap bmp = new Bitmap(_control.Width, _control.Height);
```

```
    Graphics g = Graphics.FromImage(bmp);
```

```
    g.FillRectangle(gradBrush, new Rectangle(0, 0, _control.Width, _control.Height));
```

```
    _control.BackgroundImage = bmp;
```

```
    _control.BackgroundImageLayout = ImageLayout.Stretch;
```

```
}
```

حال این متود رو در رخداد Paint کنترلهای دلخواه (فرم، دکمه، پنل ...) بنویسید :

```
private void Form1_Paint(object sender, PaintEventArgs e)
```

```
{
```

```
    PaintGradient(this, LinearGradientMode.Vertical, Color.RoyalBlue, Color.DarkBlue);
```

```
}
```

خصوصاً ظاهر بهتر و مناسبتری برای دکمه های شما ایجاد میکند.

ایجاد تصویر Bitmap در زمان اجرا (روش دوم)

برای اینکار دو روش ساده وجود داره

یکی از روش ها قبلاً گفته شد که از حلقه استفاده میشد،

مشکلی که اون روش داشت این بود که خیلی خیلی زمانبر بود

اما با این روش در حداقل زمان اینکار انجام میشه.

```
using System.Drawing;
```



```

void New_file(PictureBox pb)
{
    bmp_image = new Bitmap(pb.Width, pb.Height);

    g = Graphics.FromImage(bmp_image);

    g.FillRectangle(Brushes.White, new Rectangle(0, 0, bmp_image.Width, bmp_image.Height));

    pb.Image = bmp_image;
}

```

ایجاد تصاویر Bitmap در زمان اجرا (روش اول)

```

private Image CreateBitmap()
{
    System.Drawing.Bitmap flag = new System.Drawing.Bitmap(10, 10);

    for (int x = 0; x < flag.Height; ++x)
        for (int y = 0; y < flag.Width; ++y)
            flag.SetPixel(x, y, Color.White);

    for (int x = 0; x < flag.Height; ++x)
        flag.SetPixel(x, x, Color.Red);

    return flag;
}

```

```

private void Form1_Load(object sender, EventArgs e)
{
    pictureBox1.Image = CreateBitmap();
}

```

ایجاد اشیا گرافیکی

برای رسم اشکال گرافیکی در فرم یا کنترل مراحل زیر را انجام دهید:

۱- تعریف متغیری از نوع Graphics

۲- ایجاد شی گرافیک با مند CreateGraphics

دستورات زیر را ببینید:

```
Graphics  
  
objGraph1;  
  
Graphics objGraph2;  
  
objGraph1 = textBox1.CreateGraphics();  
  
objGraph2 = label1.CreateGraphics();
```

اولین و دومین دستور، دو متغیر به نام های objGraph1 و objGraph2 را از نوع Graphics معرفی می کنند. سومین دستور کنترل textBox1 را به عنوان محیط رسم گرافیک مربوط به متغیر objGraph1 و چهارمین دستور هم همینطور. (Label1)

انجام محاسبات بر روی یک فیلد از DataTable

```
myDataset.Table["myTable"].Compute("Sum(myFiledname)","FilterCreatia");
```

انجام کارهای متداول روی درایوها (Partitions)

```
using System.IO;
```

بدست آوردن لیست تمامی درایوها :

```
listBox1.Items.AddRange(DriveInfo.GetDrives());
```

بدست آوردن لیست تمامی درایوها و نوع آنها :

```
foreach (DriveInfo drv in DriveInfo.GetDrives())  
  
    listBox1.Items.Add(drv.Name + ":" + drv.DriveType);
```

بدست آوردن میزان فضای کلی و خالی درایوها:

```
foreach (DriveInfo drv in DriveInfo.GetDrives())  
  
    if (drv.IsReady)  
  
    {  
  
        listBox1.Items.Add(drv.Name + " - free space: " + drv.AvailableFreeSpace);  
  
        listBox1.Items.Add(drv.Name + " - total size: " + drv.TotalSize);  
  
    }
```

بازو بسته کردن درایو CD-ROM

```
using System.Runtime.InteropServices;
```

```
[DllImport("winmm.dll", EntryPoint = "mciSendStringA", CharSet = CharSet.Ansi)]
```

```
protected static extern int mciSendString(string lpstrCommand,
StringBuilder lpstrReturnString,
int uReturnLength,
IntPtr hwndCallback);
```

باز کردن :

```
int ret = mciSendString("set cdaudio door open", null, 0, IntPtr.Zero);
```

بستن :

```
int ret = mciSendString("set cdaudio door closed", null, 0, IntPtr.Zero);
```

باز کردن یک سایت توسط internet explorer

```
System.Diagnostics.Process.Start("iexplore.exe", "www.barnamenevis.org");
```

و برای مثال در فایرفاکس :

```
System.Diagnostics.Process.Start("C:\Program Files\Mozilla Firefox\FireFox.exe",
"www.barnamenevis.org");
```

آیا کاربر جاری Administrator هست؟

برای اینکه متوجه بشویم که آیا کاربر جاری تون عضوی از گروه کاربران Administrator هست یا نه از این کد استفاده کنیم:

```
WindowsIdentity identity = WindowsIdentity.GetCurrent();
WindowsPrincipal principal = new WindowsPrincipal(identity);
string role = "BUILTIN\Administrators";
bool IsAdmin = principal.IsInRole(role);
```

ایجاد یک نام تصادفی برای فایل

```
string randomFileName = System.IO.Path.GetRandomFileName();
```

و برای ایجاد نام منحصر بفرد برای فایل‌های موقت :

```
string tfile = Path.GetTempFileName();
```

بدست آوردن نام کامپیوتر، نام کاربر جاری و... (بوسیله کلاس Environment)

```
string pcName = Environment.MachineName;
```

```

string currentUser = Environment.UserName;

string system32 = Environment.SystemDirectory;

string osVersion = Environment.OSVersion;

string domain = Environment.UserDomainName;

int cpuCount = Environment.ProcessorCount;

```

بدست آوردن نام شرکت سازنده پراسور (CPU Manufacturer)

ابتدا System.Management رو به References پروژه بیفزایید و سپس :

```

using System.Management;

public string GetCPUManufacturer()
{
    string cpuMan = String.Empty;

    //create an instance of the Managemnet class with the
    //Win32_Processor class
    ManagementClass mgmt = new ManagementClass("Win32_Processor");

    //create a ManagementObjectCollection to loop through
    ManagementObjectCollection objCol = mgmt.GetInstances();

    //start our loop for all processors found
    foreach (ManagementObject obj in objCol)
    {
        if (cpuMan == String.Empty)
        {
            // only return manufacturer from first CPU
            cpuMan = obj.Properties["Manufacturer"].Value.ToString();
        }
    }
}

```

```

return cpuMan;
}

```

بدست آوردن نام سیستم لوکال

```

string pcName = Environment.MachineName;

using System.Net;

string pcName = Dns.GetHostName();

```

بدست آوردن میزان فضای خالی یک درایو (پارتیشن)

ابتدا System.Management رو به References پروژه بیفزایید و سپس :

```

using System.Management;

{
    //check to see if the user provided a drive letter
    //if not default it to "C"
    if (drive == "" || drive == null)
    {
        drive = "C";
    }

    //create our ManagementObject, passing it the drive letter to the
    //DevideID using WQL
    ManagementObject disk = new ManagementObject("Win32_LogicalDisk.DeviceID=\"\" + drive + "\"");
    //bind our management object
    disk.Get();

    //return the free space amount
    return Convert.ToDouble(disk["FreeSpace"]);
}

```

بدست آوردن مسیر فولدرهای ویژه

برای مثال :

```
string path = Environment.GetFolderPath(Environment.SpecialFolder.ProgramFiles);
```

سایر مسیرها قابل دستیابی با این روش :

```
Desktop, MyDocuments, System, ...
```

بدست آوردن مسیر دایرکتوری جاری برنامه

```
MessageBox.Show(AppDomain.CurrentDomain.BaseDirectory);
```

یا:

```
MessageBox.Show(System.IO.Directory.GetCurrentDirectory());
```

یا:

```
MessageBox.Show(Application.StartupPath);
```

یا:

```
MessageBox.Show(System.IO.Path.GetDirectoryName(Application.ExecutablePath));
```

بدست آوردن مدت زمان بیکاری سیستم

```
struct LASTINPUTINFO
```

```
{  
    public uint cbSize;  
    public uint dwTime;  
}
```

```
[DllImport("User32.dll")]
```

```
private static extern bool GetLastInputInfo(ref LASTINPUTINFO plii);
```

```
public static uint GetIdleTime()
```

```
{  
    LASTINPUTINFO lastInPut = new LASTINPUTINFO();  
    lastInPut.cbSize = (uint)System.Runtime.InteropServices.Marshal.SizeOf(lastInPut);  
    GetLastInputInfo(ref lastInPut);  
}
```

```

        return ((uint)Environment.TickCount - lastInPut.dwTime);
    }

```

مثالی از نحوه ی استفاده :

```

private Timer CheckIdleTimer;

private void Form1_Load(object sender, System.EventArgs e)
{
    CheckIdleTimer = new Timer();

    CheckIdleTimer.Interval = 1000;

    CheckIdleTimer.Start();

    CheckIdleTimer.Tick += new EventHandler(CheckIdleTimer_Tick);
}

private void CheckIdleTimer_Tick(object sender, System.EventArgs e)
{
    this.Text = (GetIdleTime() / 1000).ToString();
}

```

بدست آوردن لیست تمامی ناحیه های زمانی (Time Zone)

این روش مختص دات نت ۳.۵ (به بعد) است.

```
using System.Collections.ObjectModel;
```

بدست آوردن Id و DisplayName نواحی :

```

ReadOnlyCollection<TimeZoneInfo> zones = TimeZoneInfo.GetSystemTimeZones();

foreach (TimeZoneInfo zone in zones)
{
    listBox1.Items.Add(zone.Id);

    listBox2.Items.Add(zone.DisplayName);
}

```

بدست آوردن لیست تمامی فوتهای نصب شده در سیستم

```
using System.Drawing.Text;
```

```

InstalledFontCollection fonts = new InstalledFontCollection();

foreach (FontFamily font in fonts.Families)

    listBox1.Items.Add(font.Name);

```

بدست آوردن لیست تمامی فرمهای باز در یک برنامه

مثال : تغییر رنگ پشت زمینه ی تمامی فرمهای باز در یک برنامه :

```

foreach (Form frm in Application.OpenForms)

    frm.BackColor = Color.Fuchsia;

```

بدست آوردن لیست IP های یک سیستم

```
using System.Net;
```

برای سیستم لوکال :

```

IPAddress[] ipList = Dns.GetHostAddresses(Dns.GetHostName());

foreach (IPAddress ip in ipList )

    MessageBox.Show(ip.ToString());

```

برای یک سیستم خاص با دانستن نام آن :

```

IPAddress[] ipList = Dns.GetHostAddresses("Sinpin");

foreach (IPAddress ip in ipList )

    MessageBox.Show(ip.ToString());

```

بدست آوردن خروجی دستورهای خط فرمان

```
using System.Diagnostics;
```

```

private static string CaptureCommandPromptOutput(string command, string argument)

{

    ProcessStartInfo info = new ProcessStartInfo(command, argument);

    info.RedirectStandardOutput = true;

    info.UseShellExecute = false;

    info.CreateNoWindow = true;

```



```

Process p = new Process();

p.StartInfo = info;

p.Start();

return p.StandardOutput.ReadToEnd();

}

```

مثال از نحوه استفاده :

```
richTextBox1.Text = CaptureCommandPromptOutput("cmd", "/c dir");
```

بدست آوردن حجم کلی یک درایو (پارتیشن)

ابتدا System.Management رو به References پروژه بیفزایید و سپس :

```

using System.Management;

public double getHDDSize(string drive)
{
    //check to see if the user provided a drive letter

    //if not default it to "C"

    if (drive == "" || drive == null)
    {
        drive = "C";
    }

    //create our ManagementObject, passing it the drive letter to the

    //DevideID using WQL

    ManagementObject disk = new ManagementObject("Win32_LogicalDisk.DeviceID=\"\" + drive + "\"");

    //bind our management object

    disk.Get();

    //return the HDD's initial size

    return Convert.ToDouble(disk["Size"]);
}

```

بدست آوردن جزئیات اطلاعات در مورد یک پروسه در حال اجرا

```
using System.Diagnostics;
```

```

Process myProcess = new Process();

// Get the process start information of notepad.

ProcessStartInfo myProcessStartInfo = new ProcessStartInfo("notepad.exe");

// Assign 'StartInfo' of notepad to 'StartInfo' of 'myProcess' object.

myProcess.StartInfo = myProcessStartInfo;

// Create a notepad.

myProcess.Start();

System.Threading.Thread.Sleep(1000);

ProcessModule myProcessModule;

// Get all the modules associated with 'myProcess'.

ProcessModuleCollection myProcessModuleCollection = myProcess.Modules;

Console.WriteLine("Properties of the modules associated "

    +"with 'notepad' are:");

// Display the properties of each of the modules.

for( int i=0;i<myProcessModuleCollection.Count;i++)

{

    myProcessModule = myProcessModuleCollection[i];

    Console.WriteLine("The moduleName is "

        +myProcessModule.ModuleName);

    Console.WriteLine("The " +myProcessModule.ModuleName + "'s base address is: "

        +myProcessModule.BaseAddress);

    Console.WriteLine("The " +myProcessModule.ModuleName + "'s Entry point address is: "

        +myProcessModule.EntryPointAddress);

    Console.WriteLine("The " +myProcessModule.ModuleName + "'s File name is: "

        +myProcessModule.FileName);

}

// Get the main module associated with 'myProcess'.

myProcessModule = myProcess.MainModule;

// Display the properties of the main module.

Console.WriteLine("The process's main moduleName is: "

    +myProcessModule.ModuleName);

Console.WriteLine("The process's main module's base address is: "

```

```

+myProcessModule.BaseAddress);

Console.WriteLine("The process's main module's Entry point address is: "

+myProcessModule.EntryPointAddress);

Console.WriteLine("The process's main module's File name is: "

+myProcessModule.FileName);

myProcess.CloseMainWindow();

```

بدست آوردن اطلاعاتی در مورد OS و CLR

```

// OS and CLR version information.

Console.WriteLine(Environment.NewLine);

Console.WriteLine("OS PlatformID : " +

    Environment.OSVersion.Platform);

Console.WriteLine("OS Major Version : " +

    Environment.OSVersion.Version.Major);

Console.WriteLine("OS Minor Version : " +

    Environment.OSVersion.Version.Minor);

Console.WriteLine("CLR Version : " + Environment.Version);

```

بدست آوردن اطلاعاتی از نسخه‌ی ویندوز موجود بوسیله‌ی توابع API

```

using System.Runtime.InteropServices;

// Define the structure and specify the layout type as sequential.

[StructLayout(LayoutKind.Sequential)]

public class OSVersionInfo

{

    public int dwOSVersionInfoSize;

    public int dwMajorVersion;

    public int dwMinorVersion;

    public int dwBuildNumber;

    public int dwPlatformId;

    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = 128)]

    public String szCSDVersion;

```

```

}

[DllImport("kernel32.dll")]
public static extern bool GetVersionEx([In, Out] OSVersionInfo osvi);

static void Main(string[] args)
{
    OSVersionInfo osvi = new OSVersionInfo();

    osvi.dwOSVersionInfoSize = Marshal.SizeOf(osvi);

    // Obtain the OS version info.
    GetVersionEx(osvi);

    // Display the version information.
    Console.WriteLine("Class size: " + osvi.dwOSVersionInfoSize);
    Console.WriteLine("Major Version: " + osvi.dwMajorVersion);
    Console.WriteLine("Minor Version: " + osvi.dwMinorVersion);
    Console.WriteLine("Build Number: " + osvi.dwBuildNumber);
    Console.WriteLine("Platform Id: " + osvi.dwPlatformId);
    Console.WriteLine("CSD Version: " + osvi.szCSDVersion);
    Console.WriteLine("Platform: " + Environment.OSVersion.Platform);
    Console.WriteLine("Version: " + Environment.OSVersion.Version);

    // Wait to continue.
    Console.WriteLine(Environment.NewLine);
    Console.WriteLine("Main method complete. Press Enter.");
    Console.ReadLine();
}

```

بدست آوردن اطلاعاتی (مانند: سرعت، نوع و...) از کانکشنهای شبکه ای موجود لوکال

```
using System.Net.NetworkInformation;
```

```
static void ShowConnectionInfo()
```

```

{

    // Only proceed if there is a network available.

    if (NetworkInterface.GetIsNetworkAvailable())

    {

        // Get the set of all NetworkInterface objects for the local

        // machine.

        NetworkInterface[] interfaces =

            NetworkInterface.GetAllNetworkInterfaces();

        // Iterate through the interfaces and display information.

        foreach (NetworkInterface ni in interfaces)

        {

            // Report basic interface information.

            Console.WriteLine("Interface Name: {0}", ni.Name);

            Console.WriteLine("    Description: {0}", ni.Description);

            Console.WriteLine("    ID: {0}", ni.Id);

            Console.WriteLine("    Type: {0}", ni.NetworkInterfaceType);

            Console.WriteLine("    Speed: {0}", ni.Speed);

            Console.WriteLine("    Status: {0}", ni.OperationalStatus);


            // Report physical address.

            Console.WriteLine("    Physical Address: {0}",

                ni.GetPhysicalAddress().ToString());


            // Report network statistics for the interface.

            Console.WriteLine("    Bytes Sent: {0}",

                ni.GetIPv4Statistics().BytesSent);

            Console.WriteLine("    Bytes Received: {0}",

                ni.GetIPv4Statistics().BytesReceived);


            // Report IP configuration.

            Console.WriteLine("    IP Addresses:");

            foreach (UnicastIPAddressInformation addr

                in ni.GetIPProperties().UnicastAddresses)

```

```

        {
            Console.WriteLine("          - {0} (lease expires {1})",
                addr.Address, DateTime.Now +
                new TimeSpan(0, 0, (int)addr.DhcpLeaseLifetime));
        }

        Console.WriteLine(Environment.NewLine);
    }
}

else
{
    Console.WriteLine("No network available.");
}

Console.ReadLine();
}

```

بدست آوردن Handle و عنوان پهنجره‌ی فعال جاری

```
using System.Runtime.InteropServices;
```

سپس تعریف :

```
// Declare external functions.
```

```
[DllImport("user32.dll")]
```

```
private static extern IntPtr GetForegroundWindow();
```

```
[DllImport("user32.dll")]
```

```
private static extern int GetWindowText(IntPtr hWnd,
```

```
    StringBuilder text, int count);
```

و نهایتاً افزودن سه لیبل و یک تایمر (مثلاً با نام tmrRefresh) به برنامه و فعال کردن رویداد Tick آن :

```
private void tmrRefresh_Tick(object sender, EventArgs e)
```

```
{
```

```
    int chars = 256;
```

```
    StringBuilder buff = new StringBuilder(chars);
```

```

// Obtain the handle of the active window.

IntPtr handle = GetForegroundWindow();

// Update the controls.

if (GetWindowText(handle, buff, chars) > 0)
{
    lblCaption.Text = buff.ToString();

    lblHandle.Text = handle.ToString();

    if (handle == this.Handle)
    {
        lblCurrent.Text = "True";
    }
    else
    {
        lblCurrent.Text = "False";
    }
}
}

```

بدست آوردن Handle تمامی پنجره های موجود در سیستم

```

using System.Runtime.InteropServices;

// The signature for the callback method.

public delegate bool CallBack(IntPtr hwnd, int lParam);

[DllImport("user32.dll")]

public static extern int EnumWindows(CallBack callback, int param);

[DllImport("user32.dll")]

public static extern int GetWindowText(IntPtr hWnd,

    StringBuilder lpString, int nMaxCount);

static void Main(string[] args)
{

```

```

// Request that the operating system enumerate all windows,
// and trigger your callback with the handle of each one.

EnumWindows(DisplayWindowInfo, 0);

Console.ReadLine();
}

// The method that will receive the callback. The second
// parameter is not used, but is needed to match the
// callback's signature.

public static bool DisplayWindowInfo(IntPtr hWnd, int lParam)
{
    int chars = 100;

    StringBuilder buf = new StringBuilder(chars);

    if (GetWindowText(hWnd, buf, chars) != 0)
    {
        Console.WriteLine(buf);
    }

    return true;
}

```

ترسیم یک نوشته در مرکز یک کنترل

```

private void DrawCenterString(object sender, PaintEventArgs e)
{
    Control ct = sender as Control;

    string text = "Middle String";

    SizeF stringSize = e.Graphics.MeasureString(text, ct.Font);

    e.Graphics.DrawString(text, Font,
        Brushes.Red,
        (ct.Width - stringSize.Width) / 2,
        (ct.Height - stringSize.Height) / 2);
}

```

به قسمت آبی رنگ که سایز نوشته رو بدست میاورد دقت کنید.

مثال - برای نوشتن در مرکز یک پنل :

```
private void panell_Paint(object sender, PaintEventArgs e)
{
    DrawCenterString(sender, e);
}
```

ترسیم و نوشتن مستقیم بر دسکتاپ ویندوز

```
using System.Drawing;
using System.Drawing.Drawing2D;
using System.Runtime.InteropServices;

[DllImport("user32")]
internal static extern IntPtr GetDC(IntPtr hwnd);

[DllImport("User32.dll")]
internal static extern void ReleaseDC(IntPtr dc);

public void PaintRectangleToScreen()
{
    IntPtr deskDC = GetDC(IntPtr.Zero);

    Graphics g = Graphics.FromHdc(deskDC);

    Font font = new Font("Arial Black", 36);

    DrawStringOnCenter(g, "Sale 1387 mobarak!", font, new Point(0, -25));

    font = new Font("Arial Black", 18);

    DrawStringOnCenter(g, "by: Sinpin", font, new Point(0, 25));

    Rectangle rect = new Rectangle(200, 300,
        Screen.PrimaryScreen.Bounds.Width - 400,
        Screen.PrimaryScreen.Bounds.Height - 600);

    g.FillRectangle(new SolidBrush(Color.FromArgb(100, Color.DodgerBlue))
        , rect);

    g.DrawRectangle(new Pen(Color.DodgerBlue, 3), rect);

    g.Dispose();
}
```

```

        ReleaseDC(deskDC);
    }

    private void DrawStringOnCenter(Graphics g, string str, Font font, Point offset)
    {
        SizeF size = g.MeasureString(str, font);

        g.DrawString(str, font, Brushes.White,

            new PointF(

                (Screen.PrimaryScreen.Bounds.Width - size.Width) / 2 + offset.X,

                (Screen.PrimaryScreen.Bounds.Height - size.Height) / 2 + offset.Y)

            );
    }

```

و مثالی از طریقه ی استفاده :

```

private void Form1_Load(object sender, EventArgs e)
{
    PaintRectangleToScreen();

    this.Close();
}

```

ترسیم برخی دکمه های ویژه به کمک GDI+

```

private void DrawCaptionButton(Graphics g, ref Rectangle rc,
CaptionButton button, ButtonState state)
{
    ControlPaint.DrawCaptionButton(g, rc, button, state);

    g.DrawString(state.ToString() + ", " + button.ToString(), Font,
    Brushes.Black, rc.Width + 5, rc.Y + (rc.Height -
    Font.Height) / 2);

    rc.Offset(0, rc.Height + 5);
}

```

```

private void Form1_Paint(object sender, PaintEventArgs e)
{

```

```

Rectangle rc = new Rectangle(0, 0, 25, 25);

DrawCaptionButton(e.Graphics, ref rc, CaptionButton.Close,
ButtonState.All);

DrawCaptionButton(e.Graphics, ref rc, CaptionButton.Help,
ButtonState.Checked);

DrawCaptionButton(e.Graphics, ref rc, CaptionButton.Maximize,
ButtonState.Flat);

DrawCaptionButton(e.Graphics, ref rc, CaptionButton.Minimize,
ButtonState.Inactive);

DrawCaptionButton(e.Graphics, ref rc, CaptionButton.Restore,
ButtonState.Normal);

DrawCaptionButton(e.Graphics, ref rc, CaptionButton.Close,
ButtonState.Pushed);
}

```

ترسیم انواع چک باکس با GDI+

```

private void DrawCheckBox(Graphics g, ref Rectangle rc, ButtonState state)
{
    ControlPaint.DrawCheckBox(g, rc, state);

    g.DrawString(state.ToString(), Font, Brushes.Black,
rc.Width + 5, rc.Y + (rc.Height - Font.Height) / 2);

    rc.Offset(0, rc.Height + 5);
}

private void Form1_Paint(object sender, PaintEventArgs e)
{
    Rectangle rc = new Rectangle(0, 0, 25, 25);

    DrawCheckBox(e.Graphics, ref rc, ButtonState.All);

    DrawCheckBox(e.Graphics, ref rc, ButtonState.Checked);

    DrawCheckBox(e.Graphics, ref rc, ButtonState.Flat);

    DrawCheckBox(e.Graphics, ref rc, ButtonState.Inactive);

    DrawCheckBox(e.Graphics, ref rc, ButtonState.Normal);

    DrawCheckBox(e.Graphics, ref rc, ButtonState.Pushed);
}

```

ترسیم انواع border های ۳ بعدی به کمک GDI+

```
{  
  
    ControlPaint.DrawBorder3D(g, rc, borderStyle);  
  
    g.DrawString(borderStyle.ToString(), Font, Brushes.Black,  
        rc.Width + 5, rc.Y + (rc.Height - Font.Height) / 2);  
  
    rc.Offset(0, rc.Height + 5);  
  
}  
  
private void Form1_Paint(object sender, PaintEventArgs e)  
{  
  
    Rectangle rc = new Rectangle(0, 0, 50, 25);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.Adjust);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.Bump);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.Etched);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.Flat);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.Raised);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.RaisedInner);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.RaisedOuter);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.Sunken);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.SunkenInner);  
  
    DrawBorder3D(e.Graphics, ref rc, Border3DStyle.SunkenOuter);  
  
}
```

تبدیل یک تصویر رنگی به معادل GrayScal آن

```
using System.Drawing.Imaging;  
  
{  
  
    //create a blank bitmap the same size as original  
  
    Bitmap newBitmap =  
  
        new Bitmap(original.Width, original.Height);
```

```

//get a graphics object from the new image
Graphics g = Graphics.FromImage(newBitmap);

//create the grayscale ColorMatrix
ColorMatrix colorMatrix = new ColorMatrix(
new float[][]{
    new float[] { .3f, .3f, .3f, 0, 0 },
    new float[] { .59f, .59f, .59f, 0, 0 },
    new float[] { .11f, .11f, .11f, 0, 0 },
    new float[] { 0, 0, 0, 1, 0 },
    new float[] { 0, 0, 0, 0, 1 } });

//create some image attributes
ImageAttributes attributes = new ImageAttributes();

//set the color matrix attribute
attributes.SetColorMatrix(colorMatrix);

//draw the original image on the new image
//using the grayscale color matrix
g.DrawImage(original,
new Rectangle(0, 0, original.Width, original.Height),
0, 0, original.Width, original.Height,
GraphicsUnit.Pixel, attributes);

//dispose the Graphics object
g.Dispose();

return newBitmap;
}

private void button1_Click(object sender, EventArgs e)
{

```

مثال -

```

        Bitmap b = (Bitmap) pictureBox1.Image;

        pictureBox2.Image = MakeGrayscale(b);
    }

```

تبدیل یک IP به HostName و بالعکس

```

using System.Net;

```

```

public string ConvertIP2HostName(string ip)
{
    IPEndPoint iphost = Dns.GetHostEntry(ip);

    return iphost.HostName;
}

```

```

public string HostNameToIP(string hostName)
{
    IPEndPoint iphost = System.Net.Dns.GetHostEntry(hostName);

    IPAddress[] addresses = iphost.AddressList;

    StringBuilder addressList = new StringBuilder();

    foreach (IPAddress address in addresses)
    {
        addressList.AppendFormat("IP Address: {0};", address.ToString());
    }

    return addressList.ToString();
}

```

نحوه ی استفاده :

```

private void Form1_Load(object sender, EventArgs e)
{
    MessageBox.Show( Class4.ConvertIP2HostName("127.0.0.1"));

    MessageBox.Show(Class4.HostNameToIP("laptop"));
}

```

تبدیل اتوماتیک دکمه Enter به Tab جهت انتقال فوکوس در کنترلرها

این کد را در کلاس مربوط به کنترل TextBox خود بنویسید :

```
protected override bool ProcessDialogKey(Keys keyData)
{
    if (keyData == Keys.Return)
        SendKeys.Send("{TAB}");
    return base.ProcessDialogKey(keyData);
}
```

و در مورد استفاده در کنترل‌های استاندارد :

```
private void textBox1_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Return)
        SendKeys.Send("{TAB}");
}
```

تبدیل string به int

```
Convert.ToInt32();
```

پر کردن یک listbox از رنگهای سیستمی

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

    private void Form1_Load(object sender, EventArgs e)
    {
        string[] colorNames;

        colorNames = System.Enum.GetNames(typeof(KnownColor));

        listBox1.Items.AddRange(colorNames);
    }
}
```

```

    }

    private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
    {
        KnownColor selectedColor;

        selectedColor = (KnownColor)System.Enum.Parse(typeof(KnownColor), listBox1.Text);

        this.BackColor = System.Drawing.Color.FromKnownColor(selectedColor);
    }
}

```

پُر کردن دو DataGrid به صورت master detail

این تیکه کد نیاز به دو DataGrid دارد و از متد SetDataBinding برای پر کردن DataGrid ها استفاده میکند

```

DataRelation myDataRelation;

myDataRelation = new DataRelation("CustOrd", ds.Tables["Customers"].Columns["CustomerID"],
ds.Tables["Orders"].Columns["CustomerID"]);

// Add the relation to the DataSet.

ds.Relations.Add(myDataRelation);

GridOrders.SetDataBinding(ds, "Customers");

GridDetails.SetDataBinding(ds, "Customers.CustOrd");

```

پخش کردن برخی اصوات و صداهای سیستمی تنها با یک خط!

```

// Play a beep with default frequency

// and duration (800 and 200, respectively)

Console.Beep();

// Play a beep with frequency as 200 and duration as 300

Console.Beep(200, 300);

SystemSounds.Asterisk.Play();

SystemSounds.Hand.Play();

```

یا:


```
SystemSounds.Exclamation.Play();
```

```
SystemSounds.Beep.Play();
```

```
SystemSounds.Question.Play();
```

در مورد دومی این یادتون نره:

```
using System.Media;
```

پخش فایل صوتی موجود در Resource

برای دسترسی به فایل‌های صوتی داخل Resource و پخش کردن اون‌ها همیشه مانند دسترسی به bitmap ها یا عمل کرد. برای این کار همیشه طبق مثال زیر عمل کرد.....

این مثال واسه وقتی که یک فایل wav در resource وجود داشته باشه. برای فایل‌های بیشتر همیشه از ArrayList استفاده کرد.....

اضافه کردن namespace ها:

```
using System.Reflection;
```

```
using System.IO;
```

```
using System.Media;
```

```
using System.Runtime.InteropServices;
```

اضافه کردن dll مربوط به Windows Media Player :

```
[DllImport("Winmm.dll")]
```

```
private static extern bool PlaySound(byte[] data, IntPtr hMod, UInt32 dwFlags);
```

```
private const UInt32 SND_ASYNC = 1;
```

```
private const UInt32 SND_MEMORY = 4;
```

کدهای مربوطه:

```
private Assembly a = Assembly.GetExecutingAssembly();
```

```
private string _sound = string.Empty;
```

```
private Stream stream;
```

```
private Int32 length;
```

```
private byte[] _data;
```

```
foreach (string resourceName in a.GetManifestResourceNames())
```

```
if (Path.GetExtension(resourceName).ToLower() == ".wav")
```

```
_sound = resourceName;
```

```
stream = a.GetManifestResourceStream(_sound);

length = (Int32)stream.Length;

_data = newbyte[length];

stream.Read(_data, 0, length);

PlaySound(_data, IntPtr.Zero, SND_ASYNC | SND_MEMORY);
```

پاک کردن یک پردازش از حافظه

مثال زیر ، Yahoo Messenger رو می بندد.

```
Process[] ps = Process.GetProcesses();

foreach(Process p in ps)
{
    if(p.ProcessName.ToLower() == "ypager")
    {
        p.Kill();
    }
}
```

پاک کردن فایل

```
System.IO.File.Delete("FilePath");
```

یا:

```
System.IO.FileInfo fi = new FileInfo("FilePath");

fi.Delete();
```

بهبود رسم شکلها با روانسازی منحنیها و زاویه ها

```
private void Form1_Paint(object sender, PaintEventArgs e)
{
    // حالت معمولی

    e.Graphics.DrawEllipse(new Pen(Color.Red, 10), 30, 30, 50, 50);

    // روانسازی به کمک SmoothingMode.AntiAlias
```

```
e.Graphics.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.AntiAlias;

e.Graphics.DrawEllipse(new Pen(Color.Blue, 10), 30, 100, 50, 50);

}
```

به دست آوردن لیست کاربران یک گروه خاص در سیستم

برای اینکه بتوانیم لیست کاربرانی که عضو گروه خاصی هستند رو به دست بیاریم از این کد استفاده کنیم:

```
DirectoryEntry localMachine = new DirectoryEntry( "WinNT://" +
Environment.MachineName);

DirectoryEntry admGroup = localMachine.Children.Find(txtGroupName.Text, "group");

object members = admGroup.Invoke("members", null);

foreach (object groupMember in (IEnumerable)members)
{
    DirectoryEntry member = new DirectoryEntry(groupMember);

    Console.WriteLine("UserName: {0}", member.Name);
}
```

برنامه ای مشابه ColorBalance فوتوشاپ

برنامه ای که مشابه ColorBalance فوتوشاپ عمل می کند

به کمک این برنامه می توانید تعادل رنگی قرمز، سبز، آبی، و زرد را در

عکس های مورد نظر خود تغییر دهید.

می دونم که کسی عجلشو از دست نداده که از این برنامه برای اینکار

استفاده کنه ولی نمونه بسیار خوبی برای اینکاره.

توی برنامه از ۴ کنترل TrackBar استفاده شده که به صورت لیستی به فرم

اضافه میشن، نام رنگ ها هم همینطور؛ ازین رو به سادگی و با ۳ تغییر بسیار ساده

می تونید تعداد رنگ ها را تغییر دهید.

برای اینکار مقدار CNT را تغییر داده و رنگ ها و نام های خود را به آرایه ها اضافه یا کم

کنید، طول فرم هم بسته به تعداد اونها تغییر می کنه.

تعریف متغیرها

```
static int CNT = 4;

public static List<TrackBar> tbs = new List<TrackBar>();
```

```

Color[] clr = new Color[] { Color.Red, Color.Green, Color.Blue, Color.Yellow};

string[] lb = new string[] { "R", "G", "B", "Y"};

Image img2, img_org;

```

ColorBalance

```

private static Image fill_layer_ARGB(PictureBox PictureBox, Image img, Color[] clr)
{
    PictureBox.Image = img;

    Bitmap bmp_img = new Bitmap(PictureBox.Image);

    Graphics ghp = Graphics.FromImage(bmp_img);

    LinearGradientBrush LineaBrush;

    for (int i = 0; i < CNT; i++)
    {
        LineaBrush = new LinearGradientBrush(new Rectangle(0, 0, bmp_img.Width, bmp_img.Height),
        Color.FromArgb(tbs[i].Value, clr[i]), Color.FromArgb(tbs[i].Value, clr[i]),
        LinearGradientMode.BackwardDiagonal);

        ghp.FillRectangle(LineaBrush, new Rectangle(0, 0, bmp_img.Width, bmp_img.Height));
    }

    return (Image)bmp_img;
}

```

اضافه کردن TrackBar ها و Label ها

```

void add_tracks()
{
    for (int i = 0; i < CNT; i++)
    {
        TrackBar tb = new TrackBar();

        tb.Size = new Size(114, 45);

        tb.Location = new Point(i * 114 + 214, 545);

        tb.Enabled = false;

        tb.Minimum = 0;

        tb.Maximum = 255;

        tb.Value = 0;

        tb.TickStyle = TickStyle.None;

        tb.ValueChanged += new EventHandler(tb_ValueChanged);

        tbs.Add(tb);
    }
}

```

```

Label l = new Label();

l.Text = lb[i];

l.Location = new Point(i * 114 + 260, 567);

this.Controls.Add(l);

}

this.Controls.AddRange(tbs.ToArray());

this.Width += (CNT - 4) * 110;

label1.Left = (int)((pictureBox1.Width - label1.Width) / 2);

}

```

برقراری ارتباط تلفنی (Dial up) توسط TAPI32

```

add Reference Microsoft.TAPI32

TAPI32Lib.RequestMakeCall rmc = new TAPI32Lib.RequestMakeCall ();

rmc.MakeCall("Home", "09173.....", "0", "none");

```

بررسی اتصال به اینترنت Dial up و ADSL

این قطعه کد رو داخل همین بخش دیدم و خاطرم نیست کی زحمتش رو کشیده بود

در صورتیکه مقدار بازگردانده شده از این تابع IsConnectedToInternet برابر true باشد ، سیستم به اینترنت متصل است

```

using System.Runtime.InteropServices;

// بررسی اتصال به اینترنت

[DllImport("wininet.dll")]
private extern static bool InternetGetConnectedState(out int Description, int ReservedValue);

//Creating a function that uses the API function...

bool IsConnectedToInternet()
{
    bool a;

    int Desc;

    a=InternetGetConnectedState(out Desc, 0);

    return a;
}

```

یک کلاس ژنریک برای سریالایز و دیسریالایز کردن به شکل XML

```
}

using System.IO;

using System.Xml.Serialization;

public class GenericSerialization<T> where T : new()
{
    public static bool Serialize(object _object, string _path)
    {
        XmlSerializer serializer = new XmlSerializer(typeof(T));

        FileStream fs = null;

        try
        {
            fs = new FileStream(_path, FileMode.Create);

            serializer.Serialize(fs, _object);

            return true;
        }
        catch
        {
            return false;
        }
        finally
        {
            fs.Close();
        }
    }

    public static T Deserialize(string _path)
    {
        T result;

        XmlSerializer serializer = new XmlSerializer(typeof(T));
```

```

        FileStream fs = new FileStream(_path, FileMode.OpenOrCreate);

        try
        {
            result = (T) serializer.Deserialize(fs);
        }
        catch
        {
            result = new T();
        }
        finally
        {
            fs.Close();
        }
        return result;
    }
}

```

تغییر دادن میزان تمکیک پذیری (Resulation) صفحه نمایش

```

public static void ChangeResolution(int a, int b)
{
    Screen screen = Screen.PrimaryScreen;

    int iWidth = a;

    int iHeight = b;

    DEVMODE1 dm = new DEVMODE1();

    dm.dmDeviceName = new String(new char[32]);

    dm.dmFormName = new String(new char[32]);

    dm.dmSize = (short)Marshal.SizeOf(dm);

    if (0 != User_32.EnumDisplaySettings(null, User_32.ENUM_CURRENT_SETTINGS, ref dm))
    {
        dm.dmPelsWidth = iWidth;
    }
}

```

```

dm.dmPelsHeight = iHeight;

int iRet = User_32.ChangeDisplaySettings(ref dm, User_32.CDS_TEST);

if (iRet == User_32.DISP_CHANGE_FAILED)
{
    MessageBox.Show("Unable to process your request");

    MessageBox.Show("Description: Unable To Process Your Request. Sorry For This
Inconvenience.", "Information", MessageBoxButtons.OK, MessageBoxIcon.Information);
}
else
{
    iRet = User_32.ChangeDisplaySettings(ref dm, User_32.CDS_UPDATEREGISTRY);

    switch (iRet)
    {
        case User_32.DISP_CHANGE_SUCCESSFUL:
        {
            break;

            //successfull change
        }

        case User_32.DISP_CHANGE_RESTART:
        {

            MessageBox.Show("Description: You Need To Reboot For The Change To
Happen.\n If You Feel Any Problem After Rebooting Your Machine\nThen Try To Change Resolution In Safe
Mode.", "Information", MessageBoxButtons.OK, MessageBoxIcon.Information);

            break;

            //windows 9x series you have to restart
        }

        default:
        {

```



```
        MessageBox.Show("Description: Failed To Change The Resolution.",  
"Information", MessageBoxButtons.OK, MessageBoxIcon.Information);
```

```
        break;
```

```
        //failed to change
```

```
    }
```

```
    }
```

```
    }
```

```
    }
```

```
}
```

تغییر خواص یک فایل

```
using System.IO;
```

مخفی و فقط خواندنی کردن یک فایل :

```
FileInfo file = new FileInfo(@"C:\test.txt");
```

```
file.Attributes = file.Attributes | FileAttributes.ReadOnly | FileAttributes.Hidden;
```

تغییر خاصیت (حذف حالت فقط خواندنی مثال قبل):

```
file.Attributes = file.Attributes & ~FileAttributes.ReadOnly;
```

تغییر Render کنترل های دات نت ۲۰۰۵

سلام خدمت همه دوستان عزیز.

یکی از امکاناتی که در دات نت ۲۰۰۵ قرار داده شده امکان Custom Render بعضی کنترل های دات نت هستش. مثلاً براحتی میشه برای ToolStrip یا MenuStrip رندر دلخواه و ایجاد کرد. اما چجوری؟؟؟ برای توضیح یه مثال میزنم:

داخل #C یک پروژه جدید ایجاد کنید از نوع Windows Application. روی فرم یک کنترل ToolStrip و یک Button قرار بدید و یک کلاس به پروژه اضافه کنید و نام کلاس رو MyRender بزارید. وارد کلاس MyRender شده و خط `public class MyRender` رو به صورت زیر تغییر بدید:

```
public class MyRender : System.Windows.Forms.ToolStripRenderer
```

حالا داخل کلاس کدهای زیر رو بنویسید:

```
protected override void OnRenderToolStripBackground(System.Windows.Forms.ToolStripRenderEventArgs e)
```

```
{
```

```
    base.OnRenderToolStripBackground(e);
```

```
    Graphics g = e.Graphics;
```

```
    using (LinearGradientBrush b = new LinearGradientBrush(new Rectangle(0, 0,  
e.ToolStrip.Bounds.Width, e.ToolStrip.Bounds.Height), Color.White, Color.Silver, 90))
```

```
{
```

```
        g.FillRectangle(b, new Rectangle(0, 0, e.ToolStrip.Bounds.Width,  
e.ToolStrip.Bounds.Height));
```

```

    }

    }

    protected override void
    OnRenderButtonBackground(System.Windows.Forms.ToolStripItemRenderEventArgs e)

    {

        base.OnRenderButtonBackground(e);

        Graphics g = e.Graphics;

        if (e.Item.Selected)

        {

            if (e.Item.Pressed)

            {

                using (LinearGradientBrush b = new LinearGradientBrush(new Rectangle(0, 0,
                e.Item.Bounds.Width, e.Item.Bounds.Height), Color.Silver, Color.White, 90))

                {

                    g.FillRectangle(b, new Rectangle(0, 0, e.Item.Bounds.Width,
                    e.Item.Bounds.Height));

                }

            }

            else

            {

                using (LinearGradientBrush b = new LinearGradientBrush(new Rectangle(0, 0,
                e.Item.Bounds.Width, e.Item.Bounds.Height), Color.LightSkyBlue, Color.LightSkyBlue, 90))

                {

                    g.FillRectangle(b, new Rectangle(0, 0, e.Item.Bounds.Width,
                    e.Item.Bounds.Height));

                }

            }

        }

    }

}

```

حالا به فرم برنامه برگشته و کدهای زیر رو داخل کنترل button بنویسید:

```

toolStrip1.RenderMode = ToolStripRenderMode.ManagerRenderMode;

ToolStripManager.Renderer = new MyRender();

```

پس از اجرای برنامه و کلیک روی button خواهید دید که استایل toolStrip تغییر خواهد کرد. میتوان با اضافه کردن سایر قسمت های مربوط داخل کلاس استایل مورد نظر رو ایجاد کرد. این یکی از توانایی های جدید دات نت ۲۰۰۵ میباشد که باعث میشود کاربر نیازی به ایجاد یک کامپوننت جدید برای رندر کنترل نداشته باشد و فقط با ایجاد یک کلاس داخل برنامه این کار رو براحتی انجام بده. این کار رو برای تمام کنترل هایی که خاصیت RenderMode دارند همیشه انجام داد.

موفق باشید.

تعریف یک متغیر بدو تایی کردن نوع آن در دات نت فریم ورک ۳.۰ به بعد

با استفاده از کلمه کلیدی var :

برای مثال نوشتن :

```
var answer = 42;

var s = "this is a string.";

var names = new string[]{"Joe", "Bob", "Sam"};
```

معادل است با :

```
int answer = 42;

string s = "this is a string.";

string[] names = new string[]{"Joe", "Bob", "Sam"};
```

پس از مقداردهی نوع متغیر مشخص خواهد شد.

اینها درست هستند :

```
var a = "See", b = "Spot", c = "run";

string[] names = new string[]{"Jim", "Joe", "Bob"};

var b = new[] {1, 2.0, 3.5, 4.75}; // double[]

var a = new[] {1, 2, 3, 4, 5}; // int[]
```

```
foreach (var name in names)
{
    ...
}
```

اما اینها اشتباه هستند :

```
var x = null;

var a = 22, b = "Spot", c = 9.5;

var c = new[] {1, "Name", 2, "Address"};
```

چند نکته :

بلافاصله بعد از تعریف، متغیر باید مقداردهی شود در غیر اینصورت خطای زمان کامپایل رخ میدهد.

عمل تعیین نوع تنها یکبار انجام شده و پس از آن قابل تغییر نیست.

تشخیص روشن بودن Num Lock و Caps Lock

```
[DllImport("user32.dll", CharSet=CharSet.Auto,
ExactSpelling=true,CallingConvention=CallingConvention.Winapi)]
```

```
public static extern short GetKeyState(int keyCode);
```

```
bool isCapsLock = (((ushort) GetKeyState(0x14 /*VK_CAPITAL*/)) & 0xffff) != 0;
```

```
bool isNumLock = (((ushort) GetKeyState(0x90 /*VK_NUMLOCK*/)) & 0xffff) != 0;
```

تشخیص خالی بودن pictureBox

```
if (pictureBox1.Image == null)
```

Method overloading چیست؟

تعریف Method overloading : به امکان تعریف چندین متود با نام یکسان اما متفاوت درامضا (نوع و تعداد پارامترها) در یک کلاس گفته می شود. در واقع امکان داشتن ورژنهای مختلفی از یک متود درون یک کلاس. موارد زیر در Method overloading مطرح می باشند:

برای overload کردن یک متود باید متودهایی با signature های متمایز از یکدیگر ایجاد کنیم. signature هر متود شامل نوع , ترتیب و تعداد پارامترها می باشد.

return type یا مقدار برگشتی هر متود شامل signature متود نمی شود, در c# مجاز نیستیم ۲ متود با signature یکسان اما با مقادیر برگشتی متفاوت داشته باشیم.

مثال :

```
public class MyMathClass
{
    public int AddNum(int x)
    {
        return x + x;
    }

    public int AddNum(int x, int y)
    {
        return x + y;
    }

    public float AddNum(float x)
    {
        return x + x;
    }
}
```

تضمین آنکه چطور همیشه پس از اتمام کار بایک آبجکت حتما از حافظه خارج (dispose) شود

عادت کنید که همیشه از using استفاده کنید.

برای مثال :

```
using (FileStream FS = new FileStream("Test.txt", FileMode.Create))  
{  
    FS.WriteByte((byte)1);  
    FS.WriteByte((byte)2);  
    FS.WriteByte((byte)3);  
    using (StreamWriter SW = new StreamWriter(FS))  
    {  
        SW.WriteLine("some text.");  
    }  
}
```

که اگر بخواهید مثال بالا را به بدون استفاده از using پیاده سازی کنید باید چیزی شبیه به تیکه کد زیر بنویسید :

```
FileStream FS = new FileStream("Test.txt", FileMode.Create);  
  
try  
{  
    FS.WriteByte((byte)1);  
    FS.WriteByte((byte)2);  
    FS.WriteByte((byte)3);  
    StreamWriter SW = new StreamWriter(FS);  
  
    try  
    {  
        SW.WriteLine("some text.");  
    }  
  
    finally  
    {  
        if (SW != null)  
        {  
            ((IDisposable)SW).Dispose();  
        }  
    }  
}
```

```

    }

}

finally
{
    if (FS != null)
    {
        ((IDisposable)FS).Dispose();
    }
}

```

همینطور که میبینید استفاده از using کد شما را راحتتر، خواناتر و کوتاه تر میکند.

abstract class یا کلاس مجرد چیست؟

abstract class, کلاس غیر قطعی و ناتمامی است که در سطح فرزندانش باید پیاده سازی آن تکمیل گردد.

ویژگیهای abstract class :

در اعلان کلاس abstract از کلمه کلیدی abstract قبل از class استفاده می کنیم.

اگر کلاسی تنها یک رفتار و یا خاصیت غیر قطعی داشته باشد آن کلاس تبدیل به یک کلاس مجرد و غیر قطعی میگردد.

abstract class ها non-instantiatable هستند. در واقع به علت ناتمام بودن پیاده سازیشان نمی توانیم از آنها instance بسازیم.

abstract class ها حداقل یک یا بیشتر متود یا پروپرتی abstract دارند. abstract member ها بدون هیچگونه پیاده سازی فقط declare می شوند. abstract member ها قابل تعریف در کلاسی غیر از کلاس abstract نیستند.

کلاسهای فرزند abstract class باید تمامی abstract member ها را پیاده سازی کنند.

از کلمه کلیدی sealed در اعلان abstract class نمی توانیم استفاده کنیم. در واقع abstract class باید توسط فرزندانش به ارث رود.

متودهای مجرد و غیر قطعی که نیاز به override شدن دارند، نیازی به کلمه کلیدی virtual ندارند. در واقع کلمه کلیدی virtual در کلمه abstract نهفته است.

خواندن خط به خط یک رشته چند سطری

```

using System.IO;

string s = @"salam

dooste

man";

using (StringReader sr = new StringReader(s))

```

```
{
    while (sr.Peek() > 0)
        MessageBox.Show(sr.ReadLine());
};
```

بافر: نشان دادن مقدار کپی شدن به فایل: .

با استفاده از بافر می توان روی مقدار کپی شدن به فایل مدیریت کرد و مانور داد

موارد مورد نیاز برای این مثال:

۱. یک Progress Bar

۲. یک Button

این کد متد اصلی ما می باشد

```
public void CopyFile(string FileSource, string FileDestination)
{
    int NumRead;
    long FileLength;

    System.IO.FileStream From = new System.IO.FileStream(FileSource, System.IO.FileMode.Open);
    System.IO.FileStream To = new System.IO.FileStream(FileDestination,
        System.IO.FileMode.CreateNew);

    byte[] buffer = new byte[1024];
    FileLength = From.Length;
    progressBar1.Minimum = 0;
    progressBar1.Maximum = (int)FileLength;

    while (FileLength>0)
    {
        System.IO.BinaryReader Reader = new System.IO.BinaryReader(From);
        NumRead = Reader.Read(buffer, 0, 1024);
        FileLength = FileLength - NumRead;
        System.IO.BinaryWriter Writer = new System.IO.BinaryWriter(To);
        Writer.Write(buffer, 0, NumRead);
        progressBar1.Value = progressBar1.Value + NumRead;
    }
}
```

```

        Writer.Flush();

    }

    From.Close();

    To.Close();

    if (progressBar1.Value>99)
    {
        progressBar1.Value = 0;

        MessageBox.Show("Copy Finished successfully");
    }
}

```

از این کد برای فراخوانی متد اصلی در رویداد Button استفاده می شود

```

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.ShowDialog();

    folderBrowserDialog1.ShowDialog();

    System.IO.FileInfo d=new System.IO.FileInfo(openFileDialog1.FileName.ToString());

    CopyFile(openFileDialog1.FileName.ToString(),
    folderBrowserDialog1.SelectedPath.ToString()+"\\"+d.Name.ToString());
}

```

خواندن و نوشتن فایل به صورت unicode با استفاده از richTextBox

```

// read text file

richTextBox1.LoadFile(@"C:\Log.txt", RichTextBoxStreamType.UnicodePlainText);

// save text file

richTextBox1.SaveFile(@"C:\Log.txt", RichTextBoxStreamType.UnicodePlainText);

```

تغییر نام دادن (Rename) یک فایل

```

using System.IO;

```



```

public static void RenameFile(string originalName, string newName)
{
    File.Move(originalName, newName);
}

```

```

public static void Rename(FileInfo originalFile, string newName)
{
    originalFile.MoveTo(newName);
}

```

جایگزین کردن فرم با کلیک بر روی هر قسمت از آن - بدون استفاده از توابع API

```

private bool dragging;
private Point pointClicked;
private void Form1_MouseDown(object sender, MouseEventArgs e)
{
    if (e.Button == MouseButtons.Left)
    {
        // Turn drag mode on and store the point clicked.
        dragging = true;
        pointClicked = new Point(e.X, e.Y);
    }
    else
    {
        dragging = false;
    }
}
private void Form1_MouseMove(object sender, MouseEventArgs e)
{
    if (dragging)
    {
        Point pointMoveTo;
        // Find the current mouse position in screen coordinates.
        pointMoveTo = this.PointToScreen(new Point(e.X, e.Y));
    }
}

```

```

        // Compensate for the position the control was clicked.
        pointMoveTo.Offset(-pointClicked.X, -pointClicked.Y);

        // Move the form.
        this.Location = pointMoveTo;
    }
}

private void Form1_MouseUp(object sender, MouseEventArgs e)
{
    dragging = false;
}

```

جایگاه کردن فرم با کلیک بر روی هر قسمت از آن - با استفاده از توابع API

```

using System.Runtime.InteropServices;

[DllImport("user32.dll")]
static extern int ReleaseCapture();

[DllImport("user32.dll", EntryPoint = "SendMessageA")]
static extern int SendMessage(int hwnd, int wMsg, int wParam, object lParam);

private const int WM_NCLBUTTONDOWN = 161;

private void Form1_MouseDown(object sender, MouseEventArgs e)
{
    ReleaseCapture();

    SendMessage(this.Handle.ToInt32(), WM_NCLBUTTONDOWN, 2, 0);
}

```

تغییر نام دادن (Rename) یک دایرکتوری

```

using System.IO;

public static void RenameDirectory(string originalName, string newName)
{

```

```

try
{
    // "rename" it
    Directory.Move(originalName, newName);
}
catch (IOException ioe)
{
    // most likely given the directory exists or isn't empty
    Console.WriteLine(ioe.ToString());
}
}

```

تغییر بنا از ۱۴۰۱ به ۱۴۰۲

```

MessageBox.Show(System.Convert.ToDecimal(0xA).ToString());

```

تغییر زمان و تاریخ سیستم بوسیله کدنویسی

```

using System.Runtime.InteropServices;

```

```

public struct SYSTEMTIME
{
    public short wYear;
    public short wMonth;
    public short wDayOfWeek;
    public short wDay;
    public short wHour;
    public short wMinute;
    public short wSecond;
    public short wMilliseconds;
}

```

```

[DllImport("kernel32.dll", SetLastError = true)]

```

```
public static extern bool SetSystemTime([In] ref SYSTEMTIME st);
```

و یک مثال از نحوه ی استفاده :

```
private void Form1_Load(object sender, EventArgs e)
{
    SYSTEMTIME st = new SYSTEMTIME();

    st.wYear = 2003; // must be short

    st.wMonth = 5;

    st.wDay = 22;

    st.wHour = 0;

    st.wMinute = 0;

    st.wSecond = 0;

    SetSystemTime(ref st); // invoke this method.
}
```

تغییر زبان پیش فرض ویندوز بصورت دینامیک در برنامه

```
public struct LanguageHelper
{
    public static void SetLanguage2English()
    {
        InputLanguage.CurrentInputLanguage = InputLanguage.FromCulture(new CultureInfo("en-US"));
    }

    public static void SetLanguage2Farsi()
    {
        InputLanguage.CurrentInputLanguage = InputLanguage.FromCulture(new CultureInfo("fa-IR"));
    }
}
```

چگونه فقط یک نسخه از برنامه اجرا شه!!!!

با سلام

بعضی وقتها می خواهیم فقط یک نسخه از برنامه ما اجرا شود .

برای این کار ابتدا تمام پروسسهای ویندوز را چک می کنیم تا اگه برنامه ای با همان نام پیدا شد ، محل اجرای آن را چک میکند و

```
using System;

using System.Runtime.InteropServices;

using System.Windows.Forms;

using System.Diagnostics;

using System.Reflection;


public class OneInstnace
{
    [STAThread]

    public static void Main()
    {
        //Get the running instance.

        Process instance = RunningInstance();

        if (instance == null)
        {
            //There isn't another instance, show our form.

            Application.Run (new Form());
        }
        else
        {
            //There is another instance of this process.

            HandleRunningInstance(instance);
        }
    }

    public static Process RunningInstance()
    {
        Process current = Process.GetCurrentProcess();

        Process[] processes = Process.GetProcessesByName (current.ProcessName);

        //Loop through the running processes in with the same name

        foreach (Process process in processes)
        {
```

```

//Ignore the current process

if (process.Id != current.Id)

{

//Make sure that the process is running from the exe file.

if (Assembly.GetExecutingAssembly().Location.Replace("/", "\\") ==

current.MainModule.FileName)

{

//Return the other process instance.

return process;

}

}

}

//No other instance was found, return null.

return null;

}

```

```

public static void HandleRunningInstance(Process instance)

{

//Make sure the window is not minimized or maximized

ShowWindowAsync (instance.MainWindowHandle , WS_SHOWNORMAL);

//Set the real intance to foreground window

SetForegroundWindow (instance.MainWindowHandle);

}

```

```

[DllImport("User32.dll")]

```

```

private static extern bool ShowWindowAsync(

IntPtr hWnd, int cmdShow);

[DllImport("User32.dll")] private static extern bool

SetForegroundWindow(IntPtr hWnd);

```

```
private const int WS_SHOWNORMAL = 1;

}
```

چطوری میشه از توابع API استفاده کرد

باید با استفاده از اتریبیوت DllImport در System.Runtime.InteropServices فایل‌های که تابع مورد نظر توشه رو به برنامه اضافه کنی. بعد تابع رو تعریف کنی.

مثلا برای اضافه کردن تابع Beep که در فایل kernel32.dll است می‌تونیم از کد زیر استفاده کنی

```
[System.Runtime.InteropServices.DllImport("kernel32")]

private static extern int Beep(int dwFreq, int dwDuration);
```

بعد از این میتونی مثل تابع‌های عادی از اوون استفاده کنی.

در ضمن اکیدا توصیه شده که در کدهای مدیریت شده در NET از API‌ها استفاده نکنی و حداقل امکان از توابع خود BCL استفاده کنی.

برای اطلاعات بیشتر هم می‌تونیم به فصل هشتم کتاب .NET Developing Windows Based Applications with MS Press مراجعه کنی

چطوری تونم IP خودمو به دست بیارم؟

```
namespace NKUtilities

{

    using System;

    using System.Net;

    public class DNSUtility

    {

        public static int Main (string [] args)

        {

            String strHostName = new String ("");

            if (args.Length == 0)

            {

                // Getting Ip address of local machine...

                // First get the host name of local machine.

                strHostName = DNS.GetHostName ();

                Console.WriteLine ("Local Machine's Host Name: " + strHostName);

            }

        }

    }

}
```

```

    }

    else

    {

        strHostName = args[0];

    }


    // Then using host name, get the IP address list..

    IPHostEntry ipEntry = DNS.GetHostByName (strHostName);

    IPAddress [] addr = ipEntry.AddressList;

    for (int i = 0; i < addr.Length; i++)

    {

        Console.WriteLine ("IP Address {0}: {1} ", i, addr[i].ToString ());

    }

    return 0;

}

}

```

چرخیدن یک تصویر با زاویه دلخواه

```

private Bitmap rotateImage(Bitmap b, float angle)

{

    //create a new empty bitmap to hold rotated image

    Bitmap returnBitmap = new Bitmap(b.Width, b.Height);

    //make a graphics object from the empty bitmap

    Graphics g = Graphics.FromImage(returnBitmap);

    //move rotation point to center of image

    g.TranslateTransform((float)b.Width / 2, (float)b.Height / 2);

    //rotate

    g.RotateTransform(angle);

    //move image back

    g.TranslateTransform(-(float)b.Width / 2, -(float)b.Height / 2);

    //draw passed in image onto graphics object

```



```

        g.DrawImage(b, new Point(0, 0));

        return returnBitmap;
    }

```

مثال استفاده :

```

private void button1_Click(object sender, EventArgs e)
{
    Bitmap b = (Bitmap) pictureBox1.Image;

    pictureBox2.Image = rotateImage(b, 60);
}

```

جستجوی فایل در هارد

```

int num = 0;

string drives = "";

System.Collections.ArrayList arr = new System.Collections.ArrayList();

this.listBox1.Items.Clear();

arr.Add("d:\\");

//Get Client Drives List

foreach(string d in System.IO.Directory.GetLogicalDrives())
{
    drives += d + "\n";
}

MessageBox.Show(drives); // Show Drives List

//Searching *.* in Selected Path

if(arr.Count != 0)
{
    for(int i = 0; i != arr.Count;i++)
    {

```

```

        foreach(string n in System.IO.Directory.GetDirectories(arr[num].ToString()))
        {
            arr.Add(n);
        }

        foreach(string f in System.IO.Directory.GetFiles(arr[num].ToString(),"*.*"))
        {
            listBox1.Items.Add(f);

            this.labell1.Text = f.ToString();

            Application.DoEvents();
        }

        num++;
    }
}

```

```

MessageBox.Show(this.listBox1.Items.Count.ToString());

```

مدل بهینه شده ی این کد:

```

private void button1_Click(object sender, System.EventArgs e)
{
    System.Threading.Thread Se = new System.Threading.Thread(
        new System.Threading.ThreadStart(this.Search));

    n.Start();
}

private void Search()
{
    this.listBox1.Items.Clear();

    int num = 0;

    System.Collections.ArrayList arr = new System.Collections.ArrayList();

    arr.Add("d:\\");

    string drives = "";

    foreach(string d in System.IO.Directory.GetLogicalDrives())
    {
        drives += d + "\n";
    }
}

```

```

    }

    MessageBox.Show(drives);

    if(arr.Count != 0)
    {
        for(int i = 0; i != arr.Count;i++)
        {
            foreach(string n in System.IO.Directory.GetDirectories(arr[num].ToString()))
            {
                arr.Add(n);
            }

            foreach(string f in System.IO.Directory.GetFiles(arr[num].ToString(),"*.*))
            {
                listBox1.Items.Add(f);

                this.labell1.Text = f.ToString();

                Application.DoEvents();
            }

            num++;
        }
    }

    MessageBox.Show(this.listBox1.Items.Count.ToString());
}

```

جدا کردن یک تکه از عکس

```

Bitmap bmp = (Bitmap) pictureBox1.Image;

Bitmap new_bitmap = new Bitmap(32, 32);

for(int i = 0; i < 32; i++)
    for (int j = 0; j < 32; j++)
    {
        new_bitmap.SetPixel(i, j, bmp.GetPixel(i, j));
    }

pictureBox2.Image = new_bitmap;

```

خواندن خواص (Attribute) یک فایل

```
using System.IO;

public static void DisplayFileTimestamps(string path)
{
    Console.WriteLine(File.GetCreationTime(path));
    Console.WriteLine(File.GetLastAccessTime(path));
    Console.WriteLine(File.GetLastWriteTime(path));
}

public static void DisplayFileInfoTimestamps(FileInfo fileInfo)
{
    Console.WriteLine(fileInfo.CreationTime.ToString());
    Console.WriteLine(fileInfo.LastAccessTime.ToString());
    Console.WriteLine(fileInfo.LastWriteTime.ToString());
}

public static void DisplayFileHiddenAttribute(string path)
{
    if (File.Exists(path))
    {
        FileInfo fileInfo = new FileInfo(path);
        // Display whether this file is hidden
        Console.WriteLine("Is file hidden? = " +
            ((fileInfo.Attributes & FileAttributes.Hidden) ==
            FileAttributes.Hidden));
    }
}
```

دستکاری خواص (Attribute) یک فایل

```
using System.IO;
```

```

public static void ModifyFileTimestamps(string path)
{
    File.SetCreationTime(path, DateTime.Parse(@"May 10, 2003"));
    File.SetLastAccessTime(path, DateTime.Parse(@"May 10, 2003"));
    File.SetLastWriteTime(path, DateTime.Parse(@"May 10, 2003"));
}

public static void ModifyTimestamps(FileInfo fileInfo, DateTime dt)
{
    fileInfo.CreationTime = dt;
    fileInfo.LastAccessTime = dt;
    fileInfo.LastWriteTime = dt;
}

public static void MakeFileHidden(FileInfo fileInfo)
{
    // Modify this file's attributes
    fileInfo.Attributes |= FileAttributes.Hidden;
}

```

بازو بستہ کر دین CD Drive

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Runtime.InteropServices;

namespace OpenCloseDrive
{
    public partial class Form1 : Form

```

```

{
    #region Fields

    //Why did i put this here?

    String rt = String.Empty;

    /// <summary>

    /// Importing this god forbidden dll

    /// </summary>

    [DllImport("winmm.dll", EntryPoint = "mciSendStringA")]

    public static extern void mciSendStringA(String lpstrCommand, String lpstrReturnString, long
uReturnLength, long hwndCallback);

    #endregion

    #region Ctor

    public Form1()

    {

        InitializeComponent();

    }

    #endregion

    #region openb_Click

    private void openb_Click(object sender, System.EventArgs e)

    { mciSendStringA("set CDAudio door open", rt, 127, 0); }//i remember, hehe.

    #endregion

    #region closeb_Click

    private void closeb_Click(object sender, System.EventArgs e)

    { mciSendStringA("set CDAudio door closed", rt, 127, 0); }// YODA FOR L

    #endregion

}
}

```

کرفتن یک فایل از طریق دیاگنک OpenFileDialog و دستکاری نام فایل

```
using System.IO;
```

```
OpenFileDialog openFileDialog = new OpenFileDialog();
```

```

if (openFile.ShowDialog() != DialogResult.OK)

    return;

// get filename from OpenFileDialog
string fileName = openFile.FileName;

// get only filename without extension and path
string onlyname = Path.GetFileNameWithoutExtension(fileName);

// get extension of filename
string extension = Path.GetExtension(fileName);

// get directory of specific filename
string directory = Path.GetDirectoryName(fileName);

// change the extension of specific filename
string newFileName = Path.ChangeExtension(fileName, "bmp");

// get the root partition of specific file
string root = Path.GetPathRoot(fileName);

```

ایندکسر (Indexers)

ایندکسرها نوعی از پراپرتیها هستند که به شما اجازه میدهند با اشیاء کلاس خود آرایه وار رفتار کنید.

تعریف یک ایندکسر نیز مانند تعریف یک پراپرتی است و در حالت کلی داریم :

```

<Modifier> <Return type> this[arguments]
{
    get { }

    Set { }
}

```

برای مثال - با فرض داشتن کلاسی مانند :

```

class Employee
{

```

```

private string[] name = new string[10];

public string this[int index]
{
    get
    {
        return name[index];
    }
    set
    {
        name[index] = value;
    }
}
}

```

میتوان نوشت :

```

Employee emp = new Employee();

emp[0] = "Joydip";

emp[1] = "Manashi";

emp[2] = "Jini";

```

چگونه یک کلاس را مهر و موم کنیم؟

کافی است در اعلان کلاس و بعد از Access modifier از کلمه کلیدی sealed استفاده کنیم.

```
public sealed class MyClass{}
```

ذخیره DataGrid در اکسل Excel

ابتدا در Add Reference و در تب Com گزینه های زیر را add کنید

Microsoft Excel 11 object Lib-1

-Microsoft Office 11 object lib 2

```

Excel.ApplicationClass excel = new ApplicationClass();

excel.Application.Workbooks.Add(true);

int ColumnIndex = 0;

foreach (DataGridViewColumn col in dgv.Columns)
{

```



```

        ColumnIndex++;

        excel.Cells[1, ColumnIndex] = col.HeaderText;

    }

    int rowIndex = 0;

    string val;

    pb.Maximum = dgv.Rows.Count;

    foreach (DataGridViewRow row in dgv.Rows)
    {

        rowIndex++;

        ColumnIndex = 0;

        foreach (DataGridViewColumn col in dgv.Columns)
        {

            ColumnIndex++;

            if (row.Cells[ColumnIndex - 1].Value == null)

                val = "";

            else

                val = row.Cells[ColumnIndex - 1].Value.ToString();

            excel.Cells[rowIndex + 1, ColumnIndex] = val;

        }

    }

    excel.Visible = true;

```

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

```

private void TextBox_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Up)
    {
        if (grid.RowCount != 0)
        {
            if (grid.CurrentRow != null)

```

```

        {
            if (grid.CurrentRow.Index > 0)
            {
                grid.CurrentCell = grid[0, grid.CurrentRow.Index - 1];
            }
        }
    }
}

else if (e.KeyCode == Keys.Down)
{
    if (grid.RowCount != 0)
    {
        if (grid.CurrentRow != null)
        {
            if (grid.CurrentRow.Index < grid.RowCount)
            {
                grid.CurrentCell = grid[0, grid.CurrentRow.Index + 1];
            }
        }
    }
}
}
}

```

دمای CPU

اگر از مادر بورد asus استفاده می کنید و نرم افزار مربوطه ی آنرا نصب کرده اید کد زیر می تواند به شما کمک کند :

```

using System;

using System.Runtime.InteropServices;

class Class1
{
    [DllImport("asus.dll")]

    public static extern Int32 GetCPUTemperature();

    [DllImport("asus.dll")]

```

```

public static extern Int32 GetMBTemperature();

static void Main(string[] args)
{
    Int32 cpuTemp = GetCPUTemperature();

    Console.WriteLine("CPU Temperature: {0}C/{1:F0}F",
cpuTemp, Fahrenheit(cpuTemp));

    Int32 mbTemp = GetMBTemperature();

    Console.WriteLine("Motherboard Temperature:
{0}C/{1:F0}F", mbTemp, Fahrenheit(mbTemp));
}

static double Fahrenheit(Int32 temperature)
{
    double dblTemp = (Convert.ToDouble(temperature)* 1.8) +
32.0;

    return dblTemp;
}
}

```

یک سری اطلاعات دیگر در مورد پروسسور (در ویندوز اکس پی) :

کلا از کلید HKEY_LOCAL_MACHINE\HARDWARE\DESCRIPTION\System\CentralProcessor می شود.

```

using System;

class Class1
{
    static void Main(string[] args)
    {
        // 0 is the first processor, and 1 would be second
        processor and so on

        string keyName =
"HARDWARE\DESCRIPTION\System\CentralProcessor\0";

        //create instance of registrykey object with the
registry classes opensubkey method

        Microsoft.Win32.RegistryKey regKey =

```

```

Microsoft.Win32.Registry.LocalMachine.OpenSubKey(keyName);

//Display processor name

string valueName = "ProcessorNameString";

Console.WriteLine("{0} = {1}", valueName,
Convert.ToString(regKey.GetValue(valueName)));

//Display all value names and their data

Console.WriteLine("\nValues in key: {0}", keyName);

string[] valueNames = regKey.GetValueNames();

foreach (string valName in valueNames)

    Console.WriteLine("{0} = {1}", valName,
Convert.ToString(regKey.GetValue(valName)));

    }

}

```

دستری بریکه بندی سیستم در CLR

شاید براتون پیش بیاد که بخواهید به فایل WEB.Config و یا App.Config در CLR دسترسی پیدا کنید.

اگر بخواهید با استفاده از منوی Add Reference این کار رو انجام بدید خواهید دید که در فهرست Reference ها خبری از System.configuration نیست.

برای اینکار باید فایل پروژه ی خودتون رو که میتونه پسوند vbproj و یا csproj داشته باشه (بسته به اینکه با C# و یا VB نوشته شده) رو تو یک Editor متنی باز کنید و در بخش ItemGroup خط زیر رو اضافه کنید :

```
<Reference Include="System.configuration" />
```

فایل رو ذخیره کنید و پروژه رو دوباره باز کنید.

حالا می تونید به Application Configuration دسترسی داشته باشید.

دریافت فایل متنی از web در Windows Application

```

using System.Net;

using System.IO;

WebRequest request = FileWebRequest.Create("http://www.test.com/a.txt");

WebResponse response = request.GetResponse();

```

```

string LocalPath = "C:\\\\filename.txt";

byte[] buffer = new byte[response.ContentLength];

response.GetResponseStream().Read(buffer,0,Convert.ToInt32(response.ContentLength));

FileStream fs = new FileStream(LocalPath, FileMode.Create, FileAccess.Write);

fs.Write(buffer,0,Convert.ToInt32(buffer.Length));

response.Close();

fs.Close();

```

خواندن و نوشتن رنگ یک پیکسل به کمک کلاس Bitmap

```

using System.Drawing.Imaging;

Bitmap bmp = new Bitmap(pictureBox1.Image);

//read

Color c = bmp.GetPixel(1, 1);

//write

bmp.SetPixel(2, 2, c);

private void GetPixel_Example(PaintEventArgs e)
{

    // Create a Bitmap object from an image file.

    Bitmap myBitmap = new Bitmap("Grapes.jpg");

    // Get the color of a pixel within myBitmap.

    Color pixelColor = myBitmap.GetPixel(50, 50);

```

مثال:

```
// Fill a rectangle with pixelColor.

SolidBrush pixelBrush = new SolidBrush(pixelColor);

e.Graphics.FillRectangle(pixelBrush, 0, 0, 100, 100);

}
```

خواندن و نوشتن رنگ یک پیکسل به کمک توابع API

```
using System.Drawing;

using System.Runtime.InteropServices;

using System.Windows.Forms;

[DllImport( "user32.dll" )]

static extern IntPtr GetDC( IntPtr hWnd );

[DllImport( "user32.dll" )]

static extern int ReleaseDC( IntPtr hWnd, IntPtr hDC );

[DllImport( "gdi32.dll" )]

static extern int GetPixel( IntPtr hDC, int x, int y );

[DllImport( "gdi32.dll" )]

static extern int SetPixel( IntPtr hDC, int x, int y, int color );

static public Color GetPixel( Control control, int x, int y )

{

    Color color = Color.Empty;

    if (control != null)

    {

        IntPtr hDC = GetDC( control.Handle );

        int colorRef = GetPixel( hDC, x, y );

        color = Color.FromArgb(

            (int)(colorRef & 0x000000FF),

            (int)(colorRef & 0x0000FF00) >> 8,
```

```

        (int)(colorRef & 0x00FF0000) >> 16 );

    ReleaseDC( control.Handle, hDC );

}

return color;

}

static public void SetPixel( Control control, int x, int y, Color color )
{
    if (control != null)
    {
        IntPtr hDC = GetDC( control.Handle );

        int argb = color.ToArgb();

        int colorRef =

            (int)((argb & 0x00FF0000) >> 16) |

            (int)(argb & 0x0000FF00) |

            (int)((argb & 0x000000FF) << 16);

        SetPixel( hDC, x, y, colorRef );

        ReleaseDC( control.Handle, hDC );

    }

}

```

خالی کردن سطل بازیافت ویندوز (RecycleBin)

```

using System.Runtime.InteropServices;

enum RecycleFlags : uint
{
    SHERB_NOCONFIRMATION = 0x00000001,
    SHERB_NOPROGRESSUI = 0x00000001,
    SHERB_NOSOUND = 0x00000004
}

[DllImport("Shell32.dll", CharSet = CharSet.Unicode)]
static extern uint SHEmptyRecycleBin(IntPtr hwnd, string pszRootPath,

```

```
RecycleFlags dwFlags);
```

```
private void button1_Click(object sender, EventArgs e)
{
    uint result = SHEmptyRecycleBin(IntPtr.Zero, null, 0);
}
```

شفاف کردن پشت زمینه ی کنترلها

ایجاد شفافیت کامل پشت زمینه ی یک کنترل :

```
control.BackColor = Color.Transparent;
```

تعیین میزان شفافیت یک کنترل:

```
control.BackColor = Color.FromArgb(trancparency, control.BackColor);
```

در کد بالا مقدار transparency از ۰ (برای کاملاً شفاف) تا ۲۵۵ (کاملاً کدر) قابل تغییر است.

مثال : یک Label با شفافیت ۵۰٪

```
label1.BackColor = Color.FromArgb(127, label1.BackColor);
```

شروع آشنایی با LINQ بصورت ساده

مثال ۱ -

فرض کنیم کلاسی مانند زیر تعریف کردیم :

```
public class User
{
    public int Age { get; set; }

    public string UserName { get; set; }
}
```

حال در فرم خود بنویسید :

```
User[] users = new User[] {
    new User{UserName="Ahmad", Age=20},
    new User{UserName="Maryam", Age=17},
    new User{UserName="Ali", Age=29},
    new User{UserName="Hooman", Age=33},
    new User{UserName="Sara", Age=22},
};
```



```
var mySelect = from user in users where user.Age > 20 && user.Age < 30 select user.UserName;
```

```
foreach (string uname in mySelect)
```

```
    MessageBox.Show(uname);
```

مثال ۲ -

```
List<string> words = new List<string> {
```

```
    "Word1", "World", "Word2", "Word3", "World4" };
```

```
var wordQuery = from word in words where word == "World" select new { word };
```

```
foreach (var name in wordQuery)
```

```
    MessageBox.Show( "-> Hello " + name.word);
```

شیه سازی عمل کلیک ماوس روی یک کنترل با کدنویسی

```
private const UInt32 WM_LBUTTONDOWN = 0x201;
```

```
private const UInt32 WM_LBUTTONUP = 0x202;
```

```
[DllImport("user32.dll")]
```

```
private static extern int SendMessage(IntPtr handle,
```

```
    UInt32 message,int wParam,int lParam);
```

```
public static void SendClick(Control receiver)
```

```
{
```

```
    if (receiver != null)
```

```
    {
```

```
        SendMessage(receiver.Handle, WM_LBUTTONDOWN, 0, 0);
```

```
        SendMessage(receiver.Handle, WM_LBUTTONUP, 0, 0);
```

```
    }
```

```
}
```

مثال :

```
private void button1_Click(object sender, EventArgs e)
{
    SendClick(button2);
}
```

```
private void button2_Click(object sender, EventArgs e)
{
    MessageBox.Show("2");
}
```

سیاه و سفید کردن عکس با کد نویسی

توسط متد زیر می توانید هر عکسی را به حالت سیاه و سفید یا همون grayScale در بیارین

```
public Image GrayScaleImage(Graphics graph, Image img, int left, int top)
{

    ColorMatrix colorMix = new ColorMatrix();

    colorMix.Matrix00 = 1 / 3f;
    colorMix.Matrix01 = 1 / 3f;
    colorMix.Matrix02 = 1 / 3f;
    colorMix.Matrix10 = 1 / 3f;
    colorMix.Matrix11 = 1 / 3f;
    colorMix.Matrix12 = 1 / 3f;
    colorMix.Matrix20 = 1 / 3f;
    colorMix.Matrix21 = 1 / 3f;
    colorMix.Matrix22 = 1 / 3f;

    ImageAttributes imgAttrib = new ImageAttributes();
    imgAttrib.SetColorMatrix(colorMix);
```

```

graph.DrawImage(img, new Rectangle(left, top, img.Width,
    img.Height), 0, 0, img.Width, img.Height,
    GraphicsUnit.Pixel, imgAttrib);

Bitmap bmp = new Bitmap(img);

return bmp;

}

```



سریالیزیشن چیست؟

: Serialize

فرآیندی برای تبدیل یک آبجکت و یا گرافی متشکل از چند آبجکت به یک حالت خطی (و جریان وار) از بایت ها برای انتقال و یا ذخیره سازی در محلی دیگر است.

: Deserialize

عمل برعکس سریالیزیشن است؛ یعنی دوباره ساختن آبجکت از روی جریانی از بایتها .

انواع فرمت ها در سریالیزیشن :

۱- Binary : بصورت باینری سریالایز میکند. (فشرده ترین و efficient ترین حالت است)

۲- SOAP : بصورت soap سریالایز میکند. (برای ارسال در شبکه و جایی که از تحت دات نت بودن کلاینتها مطمئن نیستیم)

3- XML : بصورت xml سریالایز میکند. (برای ارسال تحت شبکه - مزیت : خوانایی)

برخی از موارد کاربرد سریالیزیشن:

۱- کپی یک انشعاب از آبجکتها

۲- انتقال اطلاعات در شبکه

۳- نگهداری state ها و snapshot های مختلف در یک برنامه (برای موارد حساس : مثل تراکنشهای بانکی و ...)

۴- پیاده سازی Shallow Copy و Deep Copy

سایه دار کردن یک فرم

چطور میتوان یک فرم رو سایه دار کرد ؟

کد زیر را در فرم مورد نظر بنویسید :

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

    private const int CS_DROPSHADOW = 0x00020000;

    protected override CreateParams CreateParams
    {
        get
        {
            CreateParams p = base.CreateParams;
            p.ClassStyle |= CS_DROPSHADOW;
            return p;
        }
    }
}
```

دقت داشته باشید که برای نمایش سایه باید تیک مربوط به نمایش سایه ی ماوس را زده باشید

ساده ترین راه برای جلوگیری از Not Respond شدن برنامه در حلقه های طولانی

برنامه تمامی message های درون message queue فعلی (از قبیل رخدادها و ...) را پردازش میکند.

```
Application.DoEvents();
```

ساختن یک سند word و باز کردن آن در MSWord

ابتدا روی پروژه کلیک راست کرده، Add Reference و اسمبلیهای زیر را را اضافه نمایید:

Microsoft.Office.Core

Microsoft.Vbe.Interop

Microsoft.Office.Interop.Word

سپس :

```
using Word = Microsoft.Office.Interop.Word;
```

```
private static object n = Type.Missing;
```

```
static void Main(string[] args)
```

```
{
```

```
    // Start Word in the background.
```

```
    Word.ApplicationClass app = new Word.ApplicationClass();
```

```
    app.DisplayAlerts = Word.WdAlertLevel.wdAlertsNone;
```

```
    // Create a new document (this is not visible to the user).
```

```
    Word.Document doc = app.Documents.Add(ref n, ref n, ref n,
```

```
        ref n);
```

```
    // Add a heading and two lines of text.
```

```
    Word.Range range = doc.Paragraphs.Add(ref n).Range;
```

```
    range.InsertBefore("Test Document");
```

```
    string style = "Heading 1";
```

```
    object objStyle = style;
```

```
    range.set_Style(ref objStyle);
```

```

range = doc.Paragraphs.Add(ref n).Range;

range.InsertBefore("Line one.\nLine two.");

range.Font.Bold = 1;


// Show a print preview, and make Word visible.
doc.PrintPreview();

app.Visible = true;
}

```

ساختن لیستی از کنترل‌ها در زمان اجرا

مثال: ساختن ۱۰ دکمه درون یک لیست ژنریک و اختصاص آن به لیست کنترل‌های یک فرم

```

{

List<Button> buttons = new List<Button>();

for (int i = 0; i < 10; i++)

{

    Button btn = new Button();

    btn.Size = new Size(120, 22);

    btn.Text = string.Format("Button{0}", i);

    btn.Location = new Point(10, i * 25 + 10);

    buttons.Add(btn);

}

this.Controls.AddRange(buttons.ToArray());

}

```

روش ارسال ایمیل به چندین گیرنده

```

using System.Net.Mail;


class SendEmail

{

    public static void SendMessage(string subject, string messageBody, string fromAddress, string
toAddress, string ccAddress)

    {

```

```

    MailMessage message = new MailMessage();

    SmtplibClient client = new SmtplibClient();

    message.From = new MailAddress(fromAddress);

    // Allow multiple "To" addresses to be separated by a semi-colon
    if (toAddress.Trim().Length > 0)
    {
        foreach (string addr in toAddress.Split(';'))
        {
            message.To.Add(new MailAddress(addr));
        }
    }

    // Allow multiple "Cc" addresses to be separated by a semi-colon
    if (ccAddress.Trim().Length > 0)
    {
        foreach (string addr in ccAddress.Split(';'))
        {
            message.CC.Add(new MailAddress(addr));
        }
    }

    message.Subject = subject;

    message.Body = messageBody;

    client.Host = "YourMailServer";

    client.Send(message);
}
}

```

کار کردن با کلاس TimeZoneInfo

این کلاس مختص دات نت ۳.۵ (به بعد) است و جهت استفاده باید System.Core رو نیز به References پروژه ی خود بیافزایید.

مثال - بدست آوردن تاریخ و زمان فعلی در توکیو :

```
TimeZoneInfo tzSource = TimeZoneInfo.Local;

TimeZoneInfo tzDestination = TimeZoneInfo.FindSystemTimeZoneById("Tokyo Standard Time");

string sourceTime = TimeZoneInfo.ConvertTime(DateTime.Now,
    tzSource, tzDestination).ToShortTimeString();

MessageBox.Show(sourceTime);
```

کار کردن با کلاس EventLog

از این کلاس معمولا جهت Log کردن عملیات یک برنامه استفاده میشود.

(برای مشاهده آن به Event Viewer ویندوز مراجعه نمایید)

```
using System.Diagnostics;

// Create the source, if it does not already exist.
if(!EventLog.SourceExists("MySource"))
{
    //An event log source should not be created and immediately used.
    //There is a latency time to enable the source, it should be created
    //prior to executing the application that uses the source.
    //Execute this sample a second time to use the new source.
    EventLog.CreateEventSource("MySource", "MyNewLog");
    Console.WriteLine("CreatedEventSource");
    Console.WriteLine("Exiting, execute the application a second time to use the source.");
    // The source is created. Exit the application to allow it to be registered.
    return;
}

// Create an EventLog instance and assign its source.
EventLog myLog = new EventLog();

myLog.Source = "MySource";

// Write an informational entry to the event log.
myLog.WriteEntry("Writing to event log.");
```


قفل کردن کامپیوتر از طریق کدنویسی

```
[DllImport("user32.dll")]  
  
public static extern void LockWorkStation()  
  
// Lock workstation  
  
LockWorkStation();
```

قطع کردن اتصال اینترنت و ایل آپ از طریق دستورات خط فرمان

```
System.Diagnostics.Process.Start("rasdial", "/disconnect");
```

قرار دادن یک کنترل داخل منو

احتمالا دیدید که توی بعضی نرم افزار ها داخل یک منو آیتم یک کنترل وجود داره. مثلا یه منو که توی اون یک ComboBox نشون داده می شود. یا هر کنترل دیگری شبیه به اون.

برای اینکار یه کلاسی وجود داره به نام ToolStripControlHost

شما می تونین با ساختن یک object از این کلاس و پاس کردن کنترل مورد نظر در Constructor اون و در نهایت اضافه کردن اون به مثلا Menu یا ContextMenu و یا Toolbar از اون استفاده کنین.

```
private void Form1_Load(object sender, EventArgs e)  
{  
    MonthCalendar picker = new MonthCalendar();  
    picker.DateSelected += new DateRangeEventHandler(picker_DateSelected);  
    ToolStripControlHost host = new ToolStripControlHost(picker);  
    fileToolStripMenuItem.DropDownItems.Insert(2, host);  
}  
  
void picker_DateSelected(object sender, DateRangeEventArgs e)  
{  
    MonthCalendar picker = ((MonthCalendar)sender);  
    this.Text = picker.SelectionStart.ToString("yyyy/MMM/dd");  
}
```

قرار دادن شورتکات یک برنامه بروی Desktop و Start Menu

ابتدا روی پروژه کلیک راست کرده، Add Reference و در Tab مربوط به COM مورد Windows Script Host Object Model را اضافه نمایید.
سپس :

```
using IWshRuntimeLibrary;

public static void CreateShortcut(string destination)
{
    WshShell wshShell = new WshShell();
    object destFolder = (object)destination;

    string fileName = Path.Combine(
        (string)wshShell.SpecialFolders.Item(ref destFolder),
        "Notepad.lnk"
    );

    IWshShortcut shortcut =
        (IWshShortcut)wshShell.CreateShortcut(fileName);

    shortcut.TargetPath = Path.Combine(
        Environment.GetFolderPath(Environment.SpecialFolder.System),
        "notepad.exe"
    );

    shortcut.WorkingDirectory =
        Environment.GetFolderPath(Environment.SpecialFolder.Personal);

    shortcut.Description = "Notepad Text Editor";
    shortcut.Hotkey = "CTRL+ALT+N";

    // Configure Notepad to always start maximized.
    shortcut.WindowStyle = 3;

    // Configure the shortcut to display the first icon in Notepad.exe.
    shortcut.IconLocation = "notepad.exe, 0";

    shortcut.Save();
}
```

نحوه ی استفاده :

```
CreateShortcut("Desktop");

CreateShortcut("StartMenu");
```

نقطه یک نمونه از برنامه بتواند اجرا شود (با استفاده از WMI)

بتدا System.Management رو به References پروژه بیفزایید و سپس :

```
using System.Reflection;

using System.Management;

[STAThread]

static void Main()

{

    string[] parts =

        Assembly.GetExecutingAssembly().Location.Split("\\".ToCharArray());

    string appName = parts[parts.Length - 1];

    string query = "select name from CIM_Process where name = '"

        + appName + "'";

    ManagementObjectSearcher searcher = new ManagementObjectSearcher(query);

    int runcount = 0;

    foreach (ManagementObject item in searcher.Get())

    {

        runcount++;

        if (runcount > 1) break;

    }

    if (runcount == 1)

    {

        Application.Run(new Form1());

    }

    else

    {

        System.Windows.Forms.MessageBox.Show("قبلا اجرا شده");

        Application.Exit();

    }

}
```

```
}
```

نقطه یک نمونه از برنامه تولید اجرا شود (با استفاده از Process)

```
using System.Diagnostics;
```

این تغییرات را در کلاس مربوط به Startup برنامه اعمال کنید :

```
static class Program
```

```
{
```

```
    [STAThread]
```

```
    static void Main()
```

```
    {
```

```
        if (IsPrevInstance())
```

```
            return;
```

```
        Application.EnableVisualStyles();
```

```
        Application.SetCompatibleTextRenderingDefault(false);
```

```
        Application.Run(new Form1());
```

```
    }
```

```
private static bool IsPrevInstance()
```

```
{
```

```
    string processName = Process.GetCurrentProcess().ProcessName;
```

```
    Process[] instances = Process.GetProcessesByName(processName);
```

```
    if (instances.Length > 1)
```

```
        return true;
```

```
    else
```

```
        return false;
```

```
}
```

```
}
```

نقطه یک نمونه از برنامه تولید اجرا شود (با استفاده از Mutex)

```
using System.Threading;
```

متود Main از کلاس Startup پروژه رو به این شکل تغییر بدید :

```
[STAThread]

static void Main()

{

    bool ok;

    Mutex m = new Mutex(true, "YourNameHere", out ok);

    if (! ok)

    {

        MessageBox.Show("Another instance is already running.");

        return;

    }

    Application.EnableVisualStyles();

    Application.SetCompatibleTextRenderingDefault(false);

    Application.Run(new Form1());    // or whatever was there

    GC.KeepAlive(m);                // important!

}
```

فرستادن پارامتر به برنامه email ویندوز

توسط متد زیر می توانید مستقیما برنامه outlook ویندوز را باز کرده و مکان های مناسب ان را مقدار دهی کنید

```
private static void SendMail(String mailto,String cc,String subject,String body)

{

    String command = String.Format("mailto:{0}?cc={1}&subject={2}&body={3}", mailto, cc, subject,

body);

    System.Diagnostics.Process.Start(command);

}
```

یک مثال

```
private void button1_Click(object sender, EventArgs e)

{

    SendMail("rkiani88@yahoo.com","rkiani88@hotmail.com",

        "this a sample subject for sample mail",

        "Hi All, this is a sample code for sending parameters to outlook exprees");

}
```

فراخوانی یک رویداد از رویداد دیگر

```
private void Form1_Load(object sender, EventArgs e)
{
    MessageBox.Show(sender.ToString());
}
```

9

```
private void button1_Click(object sender, EventArgs e)
{
    System.EventArgs arg = new System.EventArgs();
    Form1_Load(button1.Text, arg);
}
```

فارسی کردن کیبورد

```
InputLanguage.CurrentInputLanguage = InputLanguage.FromCulture(new System.Globalization.CultureInfo("fa-IR"));
```

غیرفعال کردن دکمه Close یک فرم

```
using System.Runtime.InteropServices;

private const int SC_CLOSE = 0xF060;
private const int MF_GRAYED = 0x1;

[DllImport("user32.dll")]
private static extern IntPtr GetSystemMenu(IntPtr hWnd, bool bRevert);

[DllImport("user32.dll")]
private static extern int EnableMenuItem(IntPtr hMenu, int wIDEnableItem, int wEnable);

private void Form1_Load(object sender, System.EventArgs e)
{

```

```
EnableMenuItem(GetSystemMenu(this.Handle, false), SC_CLOSE, MF_GRAYED);
}
```

عوض کردن نشانگر ماوس (cursor) روی یک کنترل

استفاده از کورسرهاى پیشفرض محیط :

```
button1.Cursor = Cursors.Hand;
```

لود کردن یک کورسر سفارشی از فایل :

```
button1.Cursor = new Cursor(@"C:\WINDOWS\Cursors\hibeam.cur");
```

عوض کردن تصویر پشت زمینه desktop با کدنویسی

```
using System.Runtime.InteropServices;
using System.Drawing;
using Microsoft.Win32;
```

```
public class Wallpaper
{
    const int SPI_SETDESKWALLPAPER = 20;
    const int SPIF_UPDATEINIFILE = 0x01;
    const int SPIF_SENDWININICHANGE = 0x02;

    [DllImport("user32.dll", CharSet = CharSet.Auto)]
    static extern int SystemParametersInfo(
        int uAction, int uParam, string lpvParam, int fuWinIni);

    public enum Style : int
    {
        Tiled, Centered, Stretched
    }
}
```

```

public void SetWallpaper(string path, Style style)
{
    RegistryKey key = Registry.CurrentUser.OpenSubKey(
        "Control Panel\\Desktop", true);

    switch( style )
    {
        case Style.Stretched :

            key.SetValue(@"WallpaperStyle", "2") ;

            key.SetValue(@"TileWallpaper", "0") ;

            break;

        case Style.Centered :

            key.SetValue(@"WallpaperStyle", "1") ;

            key.SetValue(@"TileWallpaper", "0") ;

            break;

        case Style.Tiled :

            key.SetValue(@"WallpaperStyle", "1") ;

            key.SetValue(@"TileWallpaper", "1") ;

            break;

    }

    SystemParametersInfo(SPI_SETDESKWALLPAPER, 0, path,

        SPIF_UPDATEINIFILE | SPIF_SENDWININICHANGE);

}
}

```

نحوه استفاده:

```

private void setButton_Click(object sender, EventArgs e)
{
    Wallpaper w = new Wallpaper();

    w.SetWallpaper(fileName,

        (Wallpaper.Style)Enum.Parse(typeof(Wallpaper.Style), "Tiled"));

}

```


عدم نیاز به نوشتن انواع توابع سازنده (ctor) در دات نت فریمورک ۳.۰ به بعد

فرض کنید کلاسی برای کاربران خود به این شکل تعریف کردید :

```
public class User
{
    public string Password { get; set; }
    public string UserName { get; set; }
    public DateTime LastLogon { get; set; }
}
```

حال در برنامه میتوانید به اشکال زیر آبجکت آن را مقدار دهی کنید :

```
User u1 = new User { UserName="sinpin", Password="123", LastLogon=DateTime.Now };
```

```
User u2 = new User { UserName = "sinpin", Password = "123" };
```

```
User u3 = new User { UserName = "sinpin" };
```

به همین منوال میتوانید برای هر کلاسی استفاده کنید. مثلاً ساخت یک نمونه از Form2 با تغییر برخی مشخصات :

```
new Form2 { Text = "Test", BackColor = Color.Red }.Show();
```

```
new Form2 { StartPosition = FormStartPosition.CenterScreen }.Show();
```

طریقه تغییر دادن خاصیت FontStyle یک کنترل در زمان اجرا

```
private void Form1_Load(object sender, EventArgs e)
{
    textBox1.Font = new Font(textBox1.Font, FontStyle.Bold | FontStyle.Italic | FontStyle.Underline);
}
```

ضبط (رکورد) صدا از طریق میکروفون

```
using Microsoft.VisualBasic.Devices;
using Microsoft.VisualBasic;
using System.Runtime.InteropServices;
```

تابع API زیر را به کلاس برنامه بیافزایید :

```
[DllImport("winmm.dll", EntryPoint = "mciSendStringA",  
CharSet = CharSet.Ansi, SetLastError = true, ExactSpelling = true)]
```

```
private static extern int mciSendString(string lpstrCommand,  
string lpstrReturnString, int uReturnLength, int hwndCallback);
```

دکمه با نامهای زیر بسازید :

Record

SaveStop

Read

در زمان کلیک شدن دکمه Record بنویسید :

```
// record from microphone  
  
mciSendString("open new Type waveaudio Alias recsound", "", 0, 0);  
  
mciSendString("record recsound", "", 0, 0);
```

و هنگام کلیک شدن دکمه SaveStop

```
// stop and save  
  
mciSendString("save recsound c:\\record.wav", "", 0, 0);  
  
mciSendString("close recsound ", "", 0, 0);  
  
Computer c = new Computer();  
  
c.Audio.Stop();
```

و برای دکمه Read نیز :

```
Computer computer = new Computer();  
  
computer.Audio.Play("c:\\record.wav", AudioPlayMode.Background);
```

و اوار کردن لیست باکس به اسکرول تایم آیتم مشخص

مثال - برای آخرین آیتم :

```
// Set the TopIndex property of the ListBox to ensure the  
  
// most recently added items are visible.  
  
listBox1.TopIndex = listBox1.Items.Count - 1;  
  
listBox1.SelectedIndex = listBox1.Items.Count - 1;
```

نوشتن یک Event برای تعدادی کنترل و انجام کار مشابه بر روی آنها

ممکنه توی فرمتون تعداد زیادی TextBox باشه و بخواید هر بار که اون فوکوس میشه متن داخلش انتخاب بشه تا اگه کاربر بخواد اون رو عوض کنه نیازی به Del و BackSpace نداشته باشه (و یا اعمال دیگه)

میشه برای هر TextBox یه Event نوشت و اون کار رو انجام بدید

حالا اگه تعداد TextBox ها زیاد باشه برای هر کنترل یک Event

میشه برای همه اونها یک Event نوشت تا اینکار رو برای همه اونها انجام بده :

اول باید همه TextBox هایی رو که می خواید انتخاب کنین بعد یه Event براشون بزارین تا همشون از یه Event استفاده کنند بعد این کد رو داخل اون Event می گذارید.

```
private void textBox1_Enter(object sender, EventArgs e)
{
    TextBox txtbx = sender as TextBox;

    txtbx.Text.SelectAll();
}
```

نوار تبلیغاتی

من یک نوار تبلیغاتی برای اینترنت اکسپلورر می خوام بسازم که علاوه بر تبلیغات بشه خبر یا ... رو برای کاربره نرم افزار قرار داد. یعنی یک چیز مثل google bar با این تفاوت که سرچ نیست و فقط اخبار و تبلیغات توش درج می شه.

خوب برایه این کار فکر میکنم نیاز به یک هاست دارم

بقیه اش دیگه نمی دونم

این یه نمونه کد :

```
--- IObjectWithSite.cs ---

using System;

using System.Runtime.InteropServices;

namespace <YOUR_NAMESPACE_HERE>
{
    [ComVisible(true)]

    [InterfaceType(ComInterfaceType.InterfaceIsIUnknown)]

    [Guid("FC4801A3-2BA9-11CF-A229-00AA003D7352")]

    public interface IObjectWithSite
    {
        [PreserveSig] int SetSite([MarshalAs(UnmanagedType.IUnknown)] object site);

        [PreserveSig] int GetSite(ref Guid guid, out IntPtr ppvSite);
    }
}

--- BHO.cs ---
```

```

using System;

using System.Diagnostics;

using System.Runtime.InteropServices;

using Microsoft.Win32;

using SHDocVw;

namespace <YOUR_NAMESPACE_HERE>
{
    [ComVisible(true)]

    [Guid("<YOUR_GUID_HERE>")]

    [ClassInterface(ClassInterfaceType.None)]

    public class BHO : IObjectWithSite
    {
        private InternetExplorer explorer;

        #region ComRegisterFunction

        public static string BHOKEYNAME = "Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\Browser
Helper Objects";

        [ComRegisterFunction]

        public static void RegisterBHO(Type t)
        {
            RegistryKey key = Registry.LocalMachine.OpenSubKey(BHOKEYNAME, true);

            if (key == null)
            {
                key = Registry.LocalMachine.CreateSubKey(BHOKEYNAME);
            }

            string guidString = t.GUID.ToString("B");

            RegistryKey bhoKey = key.OpenSubKey(guidString, true);

            if (bhoKey == null)
            {
                bhoKey = key.CreateSubKey(guidString);
            }

            // NoExplorer:dword = 1 prevents the BHO to be loaded by Explorer

```

```

        string _name = "NoExplorer";

        object _value = (object)1;

        bhoKey.SetValue(_name, _value);

        key.Close();

        bhoKey.Close();
    }

    [ComUnregisterFunction]
    public static void UnregisterBHO(Type t)
    {
        RegistryKey key = Registry.LocalMachine.OpenSubKey(BHOKEYNAME, true);

        string guidString = t.GUID.ToString("B");

        if (key != null)
        {
            key.DeleteSubKey(guidString, false);
        }
    }

    #endregion

    #region IObjectWithSite Members

    public int SetSite(object site)
    {
        if (site != null)
        {
            explorer = (InternetExplorer)site;

            ShowBrowserBar(true);
        }
    }

    return 0;
}

    public int GetSite(ref Guid guid, out IntPtr ppvSite)
    {
        IntPtr punk = Marshal.GetIUnknownForObject(explorer);

        int hr = Marshal.QueryInterface(punk, ref guid, out ppvSite);

        Marshal.Release(punk);

        return hr;
    }

```

```

    }

    #endregion

    #region Private Functions

    private void ShowBrowserBar(bool bShow)
    {
        object pvaClsid = (object)(new Guid("GUID_OF_YOUR_BANDOBJECT_HERE").ToString("B"));

        object pvarShow = (object)bShow;

        object pvarSize = null;

        if (bShow) /* hide Browser bar before showing to prevent erroneous behavior of IE*/
        {
            object pvarShowFalse = (object>false;

            explorer.ShowBrowserBar(ref pvaClsid, ref pvarShowFalse, ref pvarSize);

        }

        explorer.ShowBrowserBar(ref pvaClsid, ref pvarShow, ref pvarSize);

    }

    #endregion

}

}

```

نمایش تصاویر در Windows picture and fax viewer از طریق کدنویسی

```

//Open with the 'Windows picture and fax viewer':

System.Diagnostics.Process.Start(@"C:\Windows\system32\rundll32.exe "

    , @"C:\Windows\system32\shimgvw.dll,ImageView_Fullscreen " + filename);

```

اینهم که آسونه اما شاید واسه دوستانی مفید باشه :

```

//Open with the 'Microsoft Paint':

System.Diagnostics.Process.Start(@"C:\Windows\system32\MSPaint.exe "

    , filename);

```

معرفی کلاسی باسه متود برای جستجو در EventLog

```

using System;

using System.Collections;

using System.Diagnostics;

public sealed class EventLogSearch
{
    private EventLogSearch() { }

    public static EventLogEntry[] FindTimeGeneratedAtOrBefore(
        IEnumerable logEntries, DateTime timeGeneratedQuery)
    {
        ArrayList entries = new ArrayList();

        foreach (EventLogEntry logEntry in logEntries)
        {
            if (logEntry.TimeGenerated <= timeGeneratedQuery)
            {
                entries.Add(logEntry);
            }
        }

        EventLogEntry[] entriesArray = new EventLogEntry[entries.Count];
        entries.CopyTo(entriesArray);

        return (entriesArray);
    }

    public static EventLogEntry[] FindTimeGeneratedAtOrAfter(
        IEnumerable logEntries, DateTime timeGeneratedQuery)
    {
        ArrayList entries = new ArrayList();

        foreach (EventLogEntry logEntry in logEntries)
        {
            if (logEntry.TimeGenerated >= timeGeneratedQuery)
            {
                entries.Add(logEntry);
            }
        }
    }
}

```

```

        }

    }

    EventLogEntry[] entriesArray = new EventLogEntry[entries.Count];

    entries.CopyTo(entriesArray);

    return (entriesArray);
}

public static EventLogEntry[] FindUserName(IEnumerable logEntries,
    string userNameQuery)
{
    ArrayList entries = new ArrayList();

    foreach (EventLogEntry logEntry in logEntries)
    {
        if (logEntry.UserName == userNameQuery)
        {
            entries.Add(logEntry);
        }
    }

    EventLogEntry[] entriesArray = new EventLogEntry[entries.Count];

    entries.CopyTo(entriesArray);

    return (entriesArray);
}
}

```

معرفی روشی بهینه جهت بدست آوردن درخت کامل یک دایرکتوری

```

using System.IO;

public IEnumerable<FileSystemInfo> GetAllFilesAndDirectories(string dir)
{
    DirectoryInfo dirInfo = new DirectoryInfo(dir);

    Stack<FileSystemInfo> stack = new Stack<FileSystemInfo>();

    stack.Push(dirInfo);
}

```



```

while (dirInfo != null || stack.Count > 0)
{
    FileSystemInfo fileInfo = stack.Pop();

    DirectoryInfo subDirectoryInfo = fileInfo as DirectoryInfo;

    if (subDirectoryInfo != null)
    {
        yield return subDirectoryInfo;

        foreach (FileSystemInfo fsi in subDirectoryInfo.GetFileSystemInfos())
        {
            stack.Push(fsi);

            dirInfo = subDirectoryInfo;
        }
    }
    else
    {
        yield return fileInfo;

        dirInfo = null;
    }
}
}

```

محو شدن تدریجی یک فرم با تغییر دادن خاصیت Opacity

```

private void button1_Click(object sender, EventArgs e)
{
    this.Opacity = 1;

    for (int i = 0; i < 100; i++)
    {
        this.Opacity -= 0.01;

        Application.DoEvents();
    }
}

```

محدود کردن تکرار باکس به پذیرش اعداد ۰-۹

```

if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))

```

```
e.Handled = true;
```

نمونه ی پیشرفته تر:

```
bool TextCheck = false;

bool KeyPressed = false;

private void textBox1_TextChanged(object sender, EventArgs e)
{
    if (!KeyPressed)
    {
        if (!TextCheck)
        {
            TextCheck = true;

            string Text = textBox1.Text;

            textBox1.Text = "";

            for (int i = 0; i < Text.Length; i++)
            {
                if (Convert.ToInt16(Text[i]) > 47 && Convert.ToInt16(Text[i]) < 58)
                {
                    textBox1.Text += Text[i];

                    textBox1.Select(textBox1.TextLength, 0);

                    TextCheck = false;
                }
            }

            KeyPressed = false;
        }
    }

private void textBox1_KeyPress(object sender, KeyPressEventArgs e)
{
    KeyPressed = true;

    e.Handled = Char.IsDigit(e.KeyChar) == false && Char.IsControl(e.KeyChar) == false;

    if (!Char.IsDigit(e.KeyChar) && ((int)e.KeyChar) != 8)

        KeyPressed = false;
}
```

```
using System.IO;

public long CalculateDirectorySize(DirectoryInfo directory, bool includeSubdirectories)
{
    long totalSize = 0;

    // Examine all contained files.
    FileInfo[] files = directory.GetFiles();

    foreach (FileInfo file in files)
    {
        totalSize += file.Length;
    }

    // Examine all contained directories.
    if (includeSubdirectories)
    {
        DirectoryInfo[] dirs = directory.GetDirectories();

        foreach (DirectoryInfo dir in dirs)
        {
            totalSize += CalculateDirectorySize(dir, true);
        }
    }

    return totalSize;
}

private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show(CalculateDirectorySize(new DirectoryInfo( @"C:\WINDOWS\System32"),
    true).ToString());
}
```

از کلاس stopwatch که در فضای نام System.Diagnostics قرار دارد هم می‌تونید استفاده کنید.

```
// Create an instance of the new Stopwatch class

Stopwatch myWatch = new Stopwatch();

// Start the timer

myWatch.Start();

//Code Here...

// Now we can stop the timer and display the elapsed time along

myWatch.Stop();

elapsedTimeTextLabel.Text = myWatch.ElapsedMilliseconds.ToString() ;
```

گرفتن و تسخیر کردن (Capture) تصویر صفحه نمایش

```
private void button1_Click(object sender, EventArgs e)
{
    this.BackgroundImage= CaptureScreen();
}

private Image CaptureScreen()
{
    Bitmap screen = new Bitmap(Screen.PrimaryScreen.Bounds.Width,
        Screen.PrimaryScreen.Bounds.Height);

    using (Graphics g = Graphics.FromImage(screen))
    {
        g.CopyFromScreen(0, 0, 0, 0, screen.Size);
    }

    return screen;
}
```

```
using Microsoft.Win32;

RegistryKey kcu = Registry.CurrentUser;

kcu.OpenSubKey("Software", true).OpenSubKey("Microsoft", true).OpenSubKey("Windows",
true).OpenSubKey("CurrentVersion", true).OpenSubKey("Run", true).SetValue("Your_name", "Your_adr.exe");
```

کلاس hatchBrush

این کلاس یکی دیگر از کلاس های کار با گرافیک است که در فضای نام System.Drawing.Drawing2D قرار گرفته است.

این کلاس از Enum ای به نام HatchStyle استفاده می کند تا استایل های از پیش تعریف شده ای را برای شما ترسیم کند.

HatchStyle دارای ۵۷ نوع استایل مختلف می باشد که همگی آن ها همراه با نام آن ها در شکل Attach شده ترسیم شده اند.

توضیحات بیشتر را با دیدن عکس متوجه خواهید شد. اگر سایز عکس کوچک است، برنامه را دانلود کنید و از طریق برنامه مشاهده کنید.

کد برنامه ای که این عکس را ترسیم می کند :

```
//create a hBrush

System.Drawing.Drawing2D.HatchBrush hBrush = new
HatchBrush((HatchStyle)(Enum.Parse(typeof(HatchStyle), senderItem.Tag.ToString(), true)), Color.Red,
Color.Green);

//draw FillRectangle With hBrush on the Source Item

e.Graphics.FillRectangle(hBrush, senderItem.ClientRectangle);

//Calculate Size Of Text

SizeF textSize = e.Graphics.MeasureString(senderItem.Tag.ToString().Trim(), this.Font);

int X_Loc = (int)(senderItem.Width - textSize.Width) / 2;

int Y_Loc = (int)(senderItem.Height - textSize.Height) / 2;
```

```
//Draw text
```

```
e.Graphics.DrawString(senderItem.Tag.ToString().Trim(), this.Font, new  
SolidBrush(this.ForeColor), X_Loc, Y_Loc);
```

کشیدن نقاشی با توابع ساده گرافیکی و ذخیره خروجی در یک فایل

```
public void Draw2File ()
```

```
{
```

```
    Bitmap    bmp = new Bitmap (500, 400);
```

```
    Graphics gr = Graphics.FromImage (bmp);
```

```
    Rectangle rect = new Rectangle (20, 20, 450, 350);
```

```
    gr.DrawRectangle (Pens.Red, rect);
```

```
    gr.FillRectangle(Brushes.Pink, rect);
```

```
    gr.DrawString ("Salam", new Font ("Arial", 10), Brushes.Red, 50, 50);
```

```
    gr.DrawArc (Pens.Red, 30, 30, 60, 60, 0, 180);
```

```
    gr.DrawEllipse(new Pen(Color.Yellow, 3), new Rectangle(200, 100, 200, 200));
```

```
    bmp.Save ("C:\\\\1.bmp");
```

```
    bmp.Dispose ();
```

```
    gr.Dispose ();
```

```
}
```

کشیدن لایه شفاف با رنگ دلخواه بر روی عکس

یادمه که قبلا یکی از دوستان دنبال روشی برای اینکار می گشت

برای اینکار حدود ۱۰۰ خط نوشته بود که در اون از حلقه های تو در تو استفاده کرده بود و عملا جواب نمی داد

با این چند خط به راحتی می تونید اینکارو انجام بدید

```
using System.Drawing.Drawing2D;
```

```
private static Image fill_layer(PictureBox Pic_Box, Image img, Color Layer_color, int Percent)
```

```
{
```

```
//به عنوان یک آپ استفاده می شود img از
```

```
//percent درصد شفافیت است
```

```

Pic_Box.Image = img;

Bitmap bmp_img = new Bitmap(Pic_Box.Image);

Graphics ghp = Graphics.FromImage(bmp_img);

LinearGradientBrush LineaBrush;

LineaBrush = new LinearGradientBrush(new Rectangle(0, 0, bmp_img.Width, bmp_img.Height),
Color.FromArgb(Percent, Layer_color), Color.FromArgb(Percent, Layer_color),
LinearGradientMode.BackwardDiagonal);

ghp.FillRectangle(LineaBrush, new Rectangle(0, 0, bmp_img.Width, bmp_img.Height));

return (Image)bmp_img;

}

```

```

pictureBox1.Image = fill_layer(pictureBox1,pictureBox2.Image, Color,Value);

```

کپی کردن فایل‌های درون یک دایرکتوری (فقط فایلها)

```

using System.IO;

```

فایلها را در مسیر C:\CopiedFolder کپی میکند :

```

olderBrowserDialog fbd = new FolderBrowserDialog();

fbd.Description = "Find Folder to Copy.";

if (fbd.ShowDialog() == DialogResult.OK)
{
    string[] files = Directory.GetFiles(fbd.SelectedPath);

    Directory.CreateDirectory(@"C:\CopiedFolder");

    for (int i = 0; i < files.Length; i++)
    {
        string tmpFileExt = Path.GetExtension(files[i]);

        string tmpFileName = Path.GetFileNameWithoutExtension(files[i]);

        File.Copy(files[i], @"C:\CopiedFolder\" + tmpFileName + tmpFileExt, true);
    }
}

```

کپی کامل یک دایرکتوری و تمامی زیرشاخه های آن بصورت بازگشتی

```

using System.IO;

```

```

void CopyDirectory(DirectoryInfo source, DirectoryInfo destination)
{
    if (!destination.Exists)
        destination.Create();

    // Copy all files.
    FileInfo[] files = source.GetFiles();

    foreach (FileInfo file in files)
        file.CopyTo(Path.Combine(destination.FullName, file.Name));

    // Process subdirectories.
    DirectoryInfo[] dirs = source.GetDirectories();

    foreach (DirectoryInfo dir in dirs)
    {
        // Get destination directory.
        string destinationDir = Path.Combine(destination.FullName, dir.Name);

        // Call CopyDirectory() recursively.
        CopyDirectory(dir, new DirectoryInfo(destinationDir));
    }
}

```

یافتن متادیراکزیم و مینیم Primitive Type های عددی

```

Int16.MaxValue;
Int16.MinValue;

Int64.MaxValue;
Int64.MinValue;

Double.MaxValue;
Double.MinValue;

```

هرس کردن یک رشته تثنی

حذف حروف خاص


```
private void PruningCharacters()
{
    string foo = "---TEST--";

    Console.WriteLine(foo.Trim(new char[] { '-' })); // Displays "TEST"

    foo = ",-TEST-,-";

    Console.WriteLine(foo.Trim(new char[] { '-', ',', ' ' })); // Displays "TEST"

    foo = "---TEST--";

    Console.WriteLine(foo.TrimStart(new char[] { '-' })); // Displays "TEST--"

    foo = ",-TEST-,-";

    Console.WriteLine(foo.TrimStart(new char[] { '-', ',', ' ' })); // Displays "TEST-,-"

    foo = "---TEST--";

    Console.WriteLine(foo.TrimEnd(new char[] { '-' })); // Displays "--TEST"

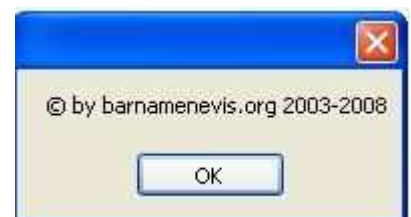
    foo = ",-TEST-,-";

    Console.WriteLine(foo.TrimEnd(new char[] { '-', ',', ' ' })); //Displays ",-TEST"
}
```

نحوه استفاده کاراکترهای خاص (مثلاً علامت کپی رایت)

```
MessageBox.Show("\xA9 by barnamenevis.org 2003-2008");
```

نتیجه ای مانند زیر (درون عکس) بدست خواهد آمد:



کرفتن کد اسکریپت و یونیکد کاراکترها

تبدیل به یونیکد

```
String ucode = String.Format("{0:x4}", (int)('a'));
```

```
// ucode = 0061
```

تبدیل به اسکی

```
String acode = (((int)('a')).ToString());
```

```
//acode=97
```

گرد کردن و رند کردن یک مقدار اعشاری

رُند کردن :

```
int x = (int)Math.Round(2.5555); // x == 3
```

گرد کردن تا دو رقم اعشار :

```
decimal x = Math.Round(2.5555, 2); // x == 2.56
```

اعداد فارسی در TextBox

```
private void textBox1_KeyPress(object sender, KeyPressEventArgs e)
```

```
{
```

```
    if ((int)e.KeyChar >= 48 && (int)e.KeyChar < 58)
```

```
        e.KeyChar = (char)(1632 + int.Parse(e.KeyChar.ToString()));
```

```
}
```

غیر فعال کردن یک رویداد در زمان اجرا

گاهی لازم است در زمان اجرا یک رویداد رو موقتاً و یا برای همیشه غیر فعال کنیم. برای اینکار با استفاده از -= ایونت هندلر مورد نظر را از رویداد حذف میکنیم.

مثال - فرض کنید که در جایی لازم است مقدار یک TextBox رو عوض کنیم بدون آنکه بخواهیم رویداد TextChanged آن تحریک شود (البته در اینجا بصورت موقت و بعد از تخصیص مقدار آن را به حالت اول برمیگردانیم) :

```
private void button1_Click(object sender, EventArgs e)
```

```
{
```

```
    // remove event handler
```

```
    textBox1.TextChanged -= new EventHandler(textBox1_TextChanged);
```

```
    textBox1.Text = "salam";
```

```
    // add event handler
```

```

        textBox1.TextChanged += new EventHandler(textBox1_TextChanged);
    }

    private void textBox1_TextChanged(object sender, EventArgs e)
    {
        ....

        MessageBox.Show("TextChanged");

        ....
    }

```

شمارش تعداد آیتم های ایست باکس

(۱) استفاده از متود count

```
listBox1.Items.Count
```

(۲) با حلقه Foreach :

```

int f = 0;

foreach (object obj in listBox1.Items)
{
    if (obj != null)
    {
        f++;
    }
}

//f = number of listbox items

```

استفاده از Place Holder ها در کار بارشده ها

فرض کنیم که :

```
int a = 3, b = 5;
```

حال میتوانیم به جای نوشتن :

```
string result = a.ToString() + "+" + b.ToString() + "=" + (a + b).ToString();
```

بسادگی بنویسیم :

```
string result = string.Format("{0}+{1}={2}", a, b, a + b);
```

استفاده از Place Holder ها (حالت دوم) ضمن خوانایی بیشتر برنامه، معمولا کدنویسی کمتری نیز طلب میکند.

روش بررسی خالی بودن یک رشته

```
string str;  
  
if (string.IsNullOrEmpty(str))  
{  
    ...  
}
```

یا:

```
string str;  
  
if (str.Trim() == "")  
{  
    ...  
}
```

یا:

```
string str;  
  
if (str == string.Empty)  
{  
    ...  
}
```

خواندن خط به خط یک رشته چند سطری

```
using System.IO;  
  
string s = @"salam  
dooste  
man";  
  
using (StringReader sr = new StringReader(s))  
{  
    while (sr.Peek() > 0)  
        MessageBox.Show(sr.ReadLine());  
};
```

نحوه برداشتن تیک از آیتم های کنترل checkedListBox

```
for (int i = 0; i < checkedListBox1.Items.Count; i++)
{
    if (checkedListBox1.GetItemChecked(i))
    {
        checkedListBox1.SetItemChecked(i, false);
    }
}
```

حالت های مختلف نمایش تاریخ و زمان با کمک String Formatting

```
static void Main(string[] args)
{
    DateTime CurrTime = DateTime.Now;

    Console.WriteLine("DateTime display listing specifier and result:\n");

    Console.WriteLine("d = {0:d}", CurrTime);

    // Short date mm/dd/yyyy

    Console.WriteLine("D = {0:D}", CurrTime);

    // Long date day, month dd, yyyy

    Console.WriteLine("f = {0:f}", CurrTime);

    // Full date/short time day, month dd, yyyy hh:mm

    Console.WriteLine("F = {0:F}", CurrTime);

    // Full date/full time day, month dd, yyyy HH:mm:ss AM/PM

    Console.WriteLine("g = {0:g}", CurrTime);

    // Short date/short time mm/dd/yyyy HH:mm

    Console.WriteLine("G = {0:G}", CurrTime);
}
```

```

// Short date/long time mm/dd/yyyy hh:mm:ss

Console.WriteLine("M = {0:M}", CurrTime);

// Month dd

Console.WriteLine("R = {0:R}", CurrTime);

// ddd Month yyyy hh:mm:ss GMT

Console.WriteLine("s = {0:s}", CurrTime);

// yyyy-mm-dd hh:mm:ss can be sorted!

Console.WriteLine("t = {0:t}", CurrTime);

// Short time hh:mm AM/PM

Console.WriteLine("T = {0:T}", CurrTime);

// Long time hh:mm:ss AM/PM

Console.WriteLine("u = {0:u}", CurrTime);

// yyyy-mm-dd hh:mm:ss universal/sortable

Console.WriteLine("U = {0:U}", CurrTime);

// day, month dd, yyyy hh:mm:ss AM/PM

Console.WriteLine("Y = {0:Y}", CurrTime);

// Month, yyyy

Console.WriteLine();

Console.WriteLine("DateTime.Month = " + CurrTime.Month);

// number of month

Console.WriteLine("DateTime.DayOfWeek = " + CurrTime.DayOfWeek);

// full name of day

```

```
Console.WriteLine("DateTime.TimeOfDay = " + CurrTime.TimeOfDay);

Console.Read();    // wait
}
```

چطور یک حلقه را یک مرحله جلو ببرم؟

```
int j = 10;

02.

03.

for (int i = 0; i < 100; i++)

04.

{

05.

j += j;

06.

07.

if (j > 100)

08.

break;

09.

}
```

```
1.

for (int i = 1; i < 10; i++)

2.

{

3.

if ((i % 2) != 0)
```

4.

```
continue;
```

5.

6.

```
System.Console.WriteLine ( "i = " + i);
```

7.

```
}
```

8.

```
}
```

جلوگیری از اجرای چندین نمونه از برنامه به صورت همزمان

مقدمه:

یکی از مشکلاتی که معمولا برنامه نویسان برنامه های تحت ویندوز با آن روبرو هستند، اجرای فقط یک نمونه از برنامه در زمان اجرا می باشد.

اجرای چند نمونه از برنامه می تواند مشکلاتی را در مورد دسترسی همزمان به اطلاعات و به اشتراک گزاردن منابع و غیره به وجود آورد. حتی ممکن است که کاربر لایسنس اجرای یک نمونه از برنامه را در هر زمان خریده باشد و در این صورت نباید بتواند چندین نمونه از برنامه را همزمان اجرا نماید.

برای انجام این عمل روش های متفاوتی وجود دارد که هر کدام دارای مشکلاتی می باشند. در این مقاله به معرفی یکی از بهترین روش ها می پردازم.

شروع:

در این مقاله ما از کلاسی به نام Mutex استفاده می کنیم که در فضای نامی System.Threading قرار دارد و جزو کلاس هایی می باشد که جهت مدیریت دسترسی Thread ها و حتی Process های مختلف به حافظه استفاده می شود.

قبل از اینکه به نوشتن قطعه کدهای مربوطه بپردازم، یادآوری می کنم که معمولا کلاس Program در برنامه های تحت ویندوز شبیه قطعه کد زیر می باشد.

```
static class Program
{
    [STAThread]

    static void Main()
    {
        Application.EnableVisualStyles();
        Application.SetCompatibleTextRenderingDefault(false);
        Application.Run(new Barber());
    }
}
```


اکنون قصد داریم که یک شرط بگذاریم تا اگر یک نمونه از برنامه در حال اجرا است و کاربر برنامه را مجدداً اجرا کرد، به کاربر پیامی دهیم که یک نمونه از برنامه در حال اجرا است و از اجرا شدن نمونه دیگری از برنامه جلوگیری کنیم.

قطعه کد کلاس Program را به شکل زیر تغییر می دهیم. قسمت های تغییر یافته به شکل bold شده، قابل مشاهده هستند.

```
static class Program

{

    static Mutex mutex = new Mutex(true, "{39d5e2c9-8791-4f77-alf4-d0e81f1b1681}");

    [STAThread]

    static void Main()

    {

        if (mutex.WaitOne(TimeSpan.Zero, true))

        {

            Application.EnableVisualStyles();

            Application.SetCompatibleTextRenderingDefault(false);

            Application.Run(new Form1());

            mutex.ReleaseMutex();

        }

        else

        {

            MessageBox.Show("در حال حاضر یک نمونه از برنامه در حال اجرا می باشد");

        }

    }

}
```

اکنون به بررسی قطعه کد بالا می پردازیم.

ابتدا یک نمونه شی استاتیک از نوع کلاس Mutex ایجاد نموده ایم و یک نام منحصر به فرد به آن نسبت داده ایم. این نام در حقیقت یک شناسه GUID می باشد.

سپس داخل متد Main که برنامه ما از آنجا اجرا می شود یک شرط گذاشته ایم.

خروجی متد WaitOne از کلاس Mutex یک مقدار منطقی می باشد. در صورتی که برای اولین بار این متد فراخوانی شود، مقدار بازگشتی آن true می باشد و تا زمانی که متد ReleaseMutex فراخوانی نشده باشد، خروجی تابع WaitOne برای سایر فراخوانی ها برابر false خواهد بود.

بنابراین هنگامی که برای اولین بار تابع `WaitOne` فراخوانی می شود مقدار `true` را بر می گرداند و برنامه اجرا می شود و تا زمانی که برنامه در حال اجرا می باشد، اگر نمونه های دیگری از برنامه را اجرا کنیم، مقدار `false` را برمی گرداند.

دقت کنید که فراموش نکنید که حتما `Mutex` را پس از خروج از برنامه `ReleaseMutex` کنید. همانطور که ما در بالا این عمل را انجام داده ایم.

برای درک بهتر، برنامه بالا را اجرا کنید و عملکرد آن را ملاحظه نمایید.

فعال نمودن نسخه ی در حال اجرای برنامه :

با اعمالی که تاکنون انجام دادیم، می توانیم به کاربر اعلام کنیم که در حال حاضر یک نمونه از برنامه در حال اجرا می باشد. ولی حالت ایده آل تری نیز وجود دارد.

بهتر است هنگامی که کاربر برای بار چندم قصد اجرای برنامه را دارد، نسخه ای از برنامه ای که در حال حاضر در حال اجرا می باشد را به صورت `TopMost` به کاربر نمایش دهیم زیرا ممکن است که برنامه در حالت `Minimize` بوده و کاربر آن را مشاهده ننماید.

برای انجام این کار کلاسی به نام `NativeMethods` می نویسیم که حاوی چند `API` سیستمی می باشد و از آن ها در برنامه استفاده خواهیم نمود.

```
internal class NativeMethods
{
    public const int HWND_BROADCAST = 0xffff;

    public static readonly int WM_SHOWME = RegisterWindowMessage("WM_SHOWME");

    [DllImport("user32")]

    public static extern bool PostMessage(IntPtr hwnd, int msg, IntPtr wparam, IntPtr lparam);

    [DllImport("user32")]

    public static extern int RegisterWindowMessage(string message);
}
```

تذکر :

با توجه به اینکه این `API` ها به صورت ثابت مورد استفاده قرار می گیرند و نیاز به تغییری در آن ها نیست، از تشریح عملکرد آن ها در برنامه خودداری می کنیم.

اکنون کلاس `Program` را به شکل زیر تغییر می دهیم. قسمت های تغییر نموده به صورت **bold** مشخص شده اند.

```
static class Program
{

    static Mutex mutex = new Mutex(true, "{39d5e2c9-8791-4f77-alf4-d0e81f1b1681}");

    [STAThread]

    static void Main()
    {
        if (mutex.WaitOne(TimeSpan.Zero, true))
        {
```

```

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

mutex.ReleaseMutex();

}

else

{

NativeMethods.PostMessage((IntPtr)NativeMethods.HWND_BROADCAST,

NativeMethods.WM_SHOWME,

IntPtr.Zero,

IntPtr.Zero);

}

}

}

```

خوب تنها یک کار دیگر باقی مانده است.

اکنون باید به فرم اصلی برنامه خود مراجعه نمایید (در اینجا Form1 می باشد) و دو متد زیر را به آن اضافه نمایید.

```

protected override void WndProc(ref Message m)

{

if (m.Msg == NativeMethods.WM_SHOWME)

{

ShowMe();

}

base.WndProc(ref m);

}

private void ShowMe()

{

if (WindowState == FormWindowState.Minimized)

{

WindowState = FormWindowState.Normal;

}

}

```

```
// get our current "TopMost" value (ours will always be false though)

bool top = TopMost;

// make our form jump to the top of everything

TopMost = true;

// set it back to whatever it was

TopMost = top;

}
```

کار تمام است. برنامه را اجرا نمایید و سعی کنید چند نمونه از آن را اجرا کنید و نتیجه را ملاحظه نمایید.

تولید رشته‌های منحصر بفرد

در ساده ترین حالت اینکار معمولا از طریق GUID انجام میشود :

```
private string GenerateId1()
{
    return Guid.NewGuid().ToString();
}
```

نمونه ای از خروجی:

```
c1eab2fa-63bb-426f-a2ff-dd87b03c0aa0
```

یک الگوریتم دیگر :

```
private string GenerateId2()
{
    long i = 1;

    foreach (byte b in Guid.NewGuid().ToByteArray())
    {
        i *= ((int)b + 1);
    }

    return string.Format("{0:x}", i - DateTime.Now.Ticks);
}
```

نمونه ای از خروجی:

```
4f2014c22f7c88ea
```

یک الگوریتم دیگر :

```
private long GenerateId3()
{

```

```
byte[] buffer = Guid.NewGuid().ToByteArray();

return BitConverter.ToInt64(buffer, 0);

}
```

نمونه ای از خروجی :

5209165259893891216

تغییر رنگ قسمتی از richbox

```
richTextBox1.Find("name");

richTextBox1.SelectionColor = Color.Red;

richTextBox1.DeselectAll();

richTextBox1.SelectionColor = Color.White;
```

تبدیل عدد ورشته به متناظر بولین آنها و بالعکس

تبدیل یک متغیر بولین به نوع صحیح :

```
bool flag = false;

int i = (flag ? 1 : 0);
```

تبدیل یک عدد صحیح به متناظر بولین آن :

```
int i = 1;

bool flag = (i == 1 ? true : false);
```

تبدیل یک رشته به متناظر بولین آن :

```
string str = "Yes";

bool flag = str.ToLower() == "yes" ? true : false;
```

تبدیل درجه حرارت به یکدیگر (سلسیوس، فارنهایت، کلوین)

```
public static double CelsiusToFahrenheit(double celsius)

{

    return (1.8 * celsius) + 32;

}

public static double FahrenheitToCelsius(double fahrenheit)

{
```

```

        return 1.8 * (fahrenheit - 32);
    }

    public static double CelsiusToKelvin(double celsius)
    {
        return celsius + 273;
    }

    public static double KelvinToCelsius(double kelvin)
    {
        return kelvin - 273;
    }

    public static double FahrenheitToKelvin(double fahrenheit)
    {
        return CelsiusToKelvin(FahrenheitToCelsius(fahrenheit));
    }

    public static double KelvinToFahrenheit(double kelvin)
    {
        return CelsiusToFahrenheit(KelvinToCelsius(kelvin));
    }

```

تبدیل درجه به رادیان و بالعکس

```

    public static double ConvertDegreesToRadians(double degrees)
    {
        return ((Math.PI / 180) * degrees);
    }

    public static double ConvertRadiansToDegrees(double radians)
    {

```

```

return ((180 / Math.PI) * radians);
}

```

تبدیل آرایه از مایتهای یک رشته و بالعکس

```

//You have a byte[] representing some binary information, such as a bitmap.
// You need to encode this data into a string so that it can be sent over
// a binary-unfriendly transport, such as email.

```

```

public string Base64EncodeBytes(byte[] inputBytes)
{
    return (Convert.ToBase64String(inputBytes));
}

```

```

//You have a String that contains information such as a bitmap encoded
// as base64. You need to decode this data (which may have been embedded in an
// email message) from a String into a byte[] so that you can access
// the original binary.

```

```

public byte[] Base64DecodeString(string inputStr)
{
    byte[] decodedByteArray = Convert.FromBase64String(inputStr);
    return (decodedByteArray);
}

```

به دست آوردن تعداد خطوط متن در یک textbox

API مربوطه

```

[DllImport("user32.dll", CharSet = CharSet.Auto, SetLastError = false)]
static extern IntPtr SendMessage(IntPtr hWnd, uint Msg, IntPtr wParam, IntPtr lParam);

const uint EM_GETLINECOUNT = 0xBA;

IntPtr wp=IntPtr.Zero;

IntPtr lp=IntPtr.Zero;

```

طریقه استفاده

```
IntPtr lines = SendMessage(textBox1.Handle, EM_GETLINECOUNT, wp, lp);

MessageBox.Show(lines.ToInt32().ToString());
```

به توان رساندن و جذر گرفتن

```
double i = Math.Pow(4, 2); // = 16;

double j = Math.Pow(4, .5); // = 2;
```

بدست آوردن قسمت صحیح یک عدد اعشاری

```
decimal d = 123.234M;

decimal i = Math.Truncate(d)
```

بدست آوردن حروف تشکیل دهنده ی یک رشته

```
string testStr = "Sinpin";

foreach (char c in testStr)

    MessageBox.Show(c.ToString());

string testStr = "Sinpin";

for (int counter = 0; counter < testStr.Length; counter++)

    MessageBox.Show(testStr[counter].ToString());
```

بدست آوردن تک تک متاویز از یک رشته ی مرکب مرزبندی شده

```
//Using the Split instance method on the String class, you can place the delimited

//information into an array in as little as a single line of code.
```

```
private void GetItemsFromDelimitedString()

{

    string delimitedInfo = "100,200,400,3,67";

    string[] discreteInfo = delimitedInfo.Split(new char[] { ',' });

    foreach (string Data in discreteInfo)

        MessageBox.Show(Data);

}
```



```
using System.Text.RegularExpressions;

public static long LineCount2(string source, bool isFileName)

{
    if (source != null)
    {
        string text = source;

        long numOfLines = 0;

        if (isFileName)
        {
            using (FileStream FS = new FileStream(source, FileMode.Open,
            FileAccess.Read, FileShare.Read))
            {
                using (StreamReader SR = new StreamReader(FS))
                {
                    while (text != null)
                    {
                        text = SR.ReadLine();

                        if (text != null)
                        {
                            ++numOfLines;
                        }
                    }
                }
            }

            return (numOfLines);
        }
        else
        {
            Regex RE = new Regex("\n", RegexOptions.Multiline);

            MatchCollection theMatches = RE.Matches(text);

            return (theMatches.Count + 1);
        }
    }
}
```

```

    }

}

else

{

    // Handle a null source here.

    return (0);

}

}

```

آغاز کار با کلاس Clipboard و دادن و گرفتن یک رشته متنی به / از آن

```

//clear clipboard

Clipboard.Clear();

//Set Text to clipboard

if (!string.IsNullOrEmpty(textBox1.Text))

    Clipboard.SetText(textBox1.Text);

//Get text from clipboard

MessageBox.Show(Clipboard.GetText());

```

ایجاد رشته ای تصادفی از اعداد با طول مشخص

```

private string GenerateRandomString(int size)

{

    Random r = new Random();

    string legalChars = "1234567890";

    StringBuilder sb = new StringBuilder();

    for (int i = 0; i < size; i++)

        sb.Append(legalChars.Substring(r.Next(0, legalChars.Length - 1), 1));

    return sb.ToString();

}

```

مثال زیر در هر بار کلیک دکمه یک رشته ی تصادفی و ۶ رقمی از اعداد را نمایش میدهد:

```

private void button1_Click(object sender, EventArgs e)

```

```
{
    MessageBox.Show(GenerateRandomString(6));
}
```

جهت تغییر دادن حروف رشته، متغیر legalChars را تغییر دهید. مثلا برای ترکیب حروف بزرگ با اعداد :

```
string legalChars = "1234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ";
```

ایجاد متن ورشته های چند سطری

```
string myString1 = "This is the first line of my string.\n" +
    "This is the second line of my string.\n" +
    "This is the third line of the string.\n";
```

```
string myString2 = @"This is the first line of my string.
    This is the second line of my string.
    This is the third line of the string.";
```

چرا و کی باید از کلاس StringBuilder به جای string استفاده کنیم؟

آبجکتهای string در دات نت immutable (غیرقابل تغییر) هستند: به این معنا که پس از مقدار گرفتن، دیگر قابل تغییر نیستند.

به مثال زیر دقت کنید :

```
string temp = "a";
temp = temp + "b";
temp += "c";
temp += "d";
```

در این مثال ابتدا یک آبجکت از نوع string میسازید، اما در خطوط بعدی با هر بار تغییر مقدار آن؛ آبجکتهای قدیمی فراموش و یک آبجکت جدید از نوع string - با مقدار جدید - ساخته میشود.

به این ترتیب همیشه انتظار داشت که کد زیر چه سرریز وحشتناکی روی منابع سیستم میتونه داشته باشه :

```
string s2 = new String('x', Int32.Parse(args[0]));

int loops = Int32.Parse(args[1]);

for (int j = 0; j < 100000000; j++) {
    string s = "";

    for (int i = loops; i > 0; --i)
```

```
s += s2;

}
```

بصورت کلی در حالتیکه اعمال تغییرات زیاد روی رشته ها نیاز باشد بهتر است آبجکتی از کلاس `StringBuilder` ساخته و با خاصیت `Append` آن کار کنیم.

مثال - پیاده سازی همان کد قبلی :

```
string s2 = new String('x', Int32.Parse(args[0]));

int loops = Int32.Parse(args[1]);

for (int j = 0; j < 10000000; j++) {

    StringBuilder sb = new StringBuilder();

    for (int i = loops; i > 0; --i)

        sb.Append(s2);

    sb.ToString();
}
```

کار با اینونت `KeyPress` و `dataGridView`

```
private void dataGridView1_EditingControlShowing(object sender,
DataGridViewEditingControlShowingEventArgs e)

{

    e.Control.KeyPress += DataGridKeyPress;

}

private void DataGridKeyPress(object sender, KeyPressEventArgs e)

{

}

}
```

`DataGridView` در `TextChanged`

```
private void dataGridView1_EditingControlShowing(object sender,
DataGridViewEditingControlShowingEventArgs e)

{

    e.Control.TextChanged += MyDataGridTextChanged;

}

private void MyDataGridTextChanged(object sender, EventArgs e)
```

```
{
    MessageBox.Show(((Control)sender).Text);
}
```

پیدا کردن معکوس یک رشته بصورت بازگشتی

سلام ، دوستان تابعی که در زیر نوشتم معکوس (مقلوب) یک رشته را می یابد ،

کد مورد نظر :

```
void str_rev(string str,ref string res)
{
    //finds reverse of string str and put result in res

    if(str.Length == 1)
    {
        res += str;
    }
    else
    {
        res += str[str.Length - 1].ToString();
        str_rev(str.Substring(0, str.Length - 1),ref res);
    }
}
```

نحوه استفاده :

```
private void button1_Click(object sender, EventArgs e)
{
    string res = "";
    str_rev("SALAR", ref res);
    MessageBox.Show(res);
}
```

چطور می‌توان کاری کرد که نشه توی TextBox فارسی نوشت و کلیدهای سوئیچی alt+shift هم بی‌تاثیر باشد

برای اینکه کلیدهای سوئیچی هیچ تاثیری نداشته باشند خصوصیت key preview فرم را با True مقدار دهی میکنیم و کد زیر رو برای خصوصیت key down فرم می نویسیم

```
if(e.Alt && e.shift)
```

```
}
```

```
InputLanguage.currentInputLanguage=InputLanguage.FromCulture(system.Globalization.CultureInfo.GetCultureInfo("fa"));
```

تبدیل عکس به آیکن.....

```
Size size = new Size(32, 32);
```

```
Bitmap bmp = new Bitmap(Image.FromFile(pathStr), size);
```

```
Icon temp = Icon.FromHandle(bmp.GetHicon());
```

```
FileStream newIcon = new FileStream("c:\\sa.ico", FileMode.Create);
```

```
temp.Save(newIcon);
```

```
newIcon.Close();
```

تغییر زبان فرم های ویندوزی از فارسی به انگلیسی و از انگلیسی به فارسی

```
private void func_SetLanguagePersian()
```

```
{
```

```
System.Globalization.CultureInfo MyCulture = new System.Globalization.CultureInfo("fa-IR");
```

```
InputLanguage MyL;
```

```
string BackL;
```

```
MyL = InputLanguage.CurrentInputLanguage;
```

```
MyCulture = MyL.Culture;
```

```
BackL = MyCulture.Name;
```

```
MyCulture = new System.Globalization.CultureInfo("fa-IR");
```

```
InputLanguage.CurrentInputLanguage =  
System.Windows.Forms.InputLanguage.FromCulture(MyCulture);
```

```
}
```

```
private void func_SetLanguageEnglish()
```

```
{
```

```

System.Globalization.CultureInfo MyCulture = new System.Globalization.CultureInfo("en");

InputLanguage MyL;

string BackL;

MyL = InputLanguage.CurrentInputLanguage;

MyCulture = MyL.Culture;

BackL = MyCulture.Name;

MyCulture = new System.Globalization.CultureInfo("en");

InputLanguage.CurrentInputLanguage =
System.Windows.Forms.InputLanguage.FromCulture(MyCulture);

}

```

چک کردن ورودی کاربر برای مقادیر عددی

یکی از روش های گرفتن مقادیر عددی برای ورودی استفاده از `MaskedTextBox` هست. روش دیگه استفاده از متد `TryParse` هست. این تابع دو پارامتر برای ورودی مگیره، یکی رشته ای که باید تبدیل بشه و دیگری متغییری که عدد تبدیل شده باید داخلش ذخیره بشه. این تابع مقداری از نوع `bool` بر میگردونه.

```

Console.WriteLine("Enter a number for check: ");

int i = 0;

string inputNumber = Console.ReadLine();

if (int.TryParse(inputNumber, out i))
{
    Console.WriteLine("Valid format");
}
else
{
    Console.WriteLine("Not valid format");
}

Console.ReadKey()

```

نمایش قرمز رنگ عبارت مورد جستجو در `RichTextBox`

توضیح: رنگ رو میتونید خودتون عوض کنید.

```

int start = 0;

int indexOfSearchText = 0;

```

```

private void button1_Click(object sender, EventArgs e)
{
    int startindex = 0;

    if (txtSearch.Text.Length > 0)
        startindex = FindMyText(txtSearch.Text.Trim(), start, rtb.Text.Length);

    if (startindex >= 0)
    {
        rtb.SelectionColor = Color.Red;

        int endindex = txtSearch.Text.Length;

        rtb.Select(startindex, endindex);

        start = startindex + endindex;
    }
}

```

```

private int FindMyText(string txtToSearch, int searchStart, int searchEnd)
{
    if (searchStart > 0 && searchEnd > 0 && indexOfSearchText >= 0)
    {
        rtb.Undo();
    }

    int retVal = -1;

    if (searchStart >= 0 && indexOfSearchText >= 0)
    {
        if (searchEnd > searchStart || searchEnd == -1)
        {

```



```

        indexOfSearchText = rtb.Find(txtToSearch, searchStart, searchEnd,
RichTextBoxFinds.None);

```

```

        if (indexOfSearchText != -1)
        {

            retVal = indexOfSearchText;

        }

    }

    return retVal;

}

```

تبدیل اعداد به رشته

```

public class NumberToString
{
    private static string[] yakan = new string[10] { "صفر", "یک", "دو", "سه", "چهار", "پنج", "شش", "هفت", "هشت", "نه" };

    private static string[] dahgan = new string[10] { "بیست", "سی", "چهل", "پنجاه", "شصت", "هفتاد", "هشتاد", "نود", "صد" };

    private static string[] dahyek = new string[10] { "ده", "یازده", "دوازده", "سیزده", "چهارده", "پانزده", "شانزده", "هفده", "هجده", "نوزده" };

    private static string[] sadgan = new string[10] { "یکصد", "دوصد", "سیصد", "چهارصد", "پانصد", "ششصد", "هشتصد", "نهمصد", "صد" };

    private static string[] basex = new string[5] { "هزار", "میلیون", "میلیارد", "تریلیون", " " };

    private static string getnum3(int num3)
    {
        string s = "";

        int d3, d12;

        d12 = num3 % 100;

```

```

d3 = num3 / 100;

if (d3 != 0)

    s = sadgan[d3] + " ۹ ";

if ((d12 >= 10) && (d12 <= 19))
{
    s = s + dahyek[d12 - 10];
}
else
{
    int d2 = d12 / 10;

    if (d2 != 0)

        s = s + dahgan[d2] + " ۹ ";

    int d1 = d12 % 10;

    if (d1 != 0)

        s = s + yakan[d1] + " ۹ ";

    s = s.Substring(0, s.Length - 3);
};

return s;
}

```

```

public string num2str(string snum)
{
    string stotal = "";

    if (snum == "0")
    {
        return yakan[0];
    }

    else
    {
        snum = snum.PadLeft(((snum.Length - 1) / 3 + 1) * 3, '0');

        int L = snum.Length / 3 - 1;

```

```

for (int i = 0; i <= L; i++)
{
    int b = int.Parse(snum.Substring(i * 3, 3));

    if (b != 0)

        stotal = stotal + getnum3(b) + " " + basex[L - i] + " و ";

}

stotal = stotal.Substring(0, stotal.Length - 3);

}

return stotal;

}
}

```

ایجاد فرم های غیر مستطیلی

یک روش استاندارد برای ایجاد فرم های غیر مستطیلی (در این مثال مثلثی شکل).

```

private void btnClose_Click(object sender, EventArgs e)
{
    this.Close();
}

private void frmMain_Load(object sender, EventArgs e)
{
    System.Drawing.Drawing2D.GraphicsPath MyPaht = new System.Drawing.Drawing2D.GraphicsPath();

    MyPaht.AddPolygon(new Point[] { new Point(0, 0), new Point(0, this.Height), new
    Point(this.Width, 0) });

    Region MyRegion = new Region(MyPaht);

    this.Region = MyRegion;
}

```

کپی آیتم های یک کامبواکس به دیگری بدون استفاده از حلقه و در یک خط

```

comboBox2.Items.AddRange(comboBox1.Items.Cast<Object>().ToArray());

```

```
Dictionary<datatype1, datatype2> name = new Dictionary<datatype1, datatype2>();
```

محاسبه فاصله بین دو زمان مشخص

انجام این کار خیلی ساده هست. از کلاس TimeSpan استفاده میکنیم.

به مثال زیر توجه کنید:

دو تا شیء از نوع DateTime میسازم، با تاریخ و ساعت دقیق:

```
DateTime myStartTime=new DateTime(2009,1,1,12,13,14,157);
```

```
DateTime myEndTime=new DateTime(2010,9,12,13,14,15,165);
```

حالا يك شیء از نوع TimeSpan میسازم و اونو اینطوری مقدار دهی میکنم:

```
TimeSpan myDifference=myEndTime-myStartTime;
```

الان داخل myDifference فاصله ی زمانی میان دو زمان مد نظرمون قرار داره. کافیه تا برای دسترسی به محتوای اون، از خصوصیات خودش استفاده کنیم.

مثلاً داریم:

```
int totalDays=myDifference.Days ; //تعداد روزهای میان دو تاریخ مذکور
```

```
int totalHours=myDifference.Hours; //تعداد ساعات میان دو تاریخ مذکور
```

```
int totalMinutes=myDifference.Minutes; //تعداد دقیقه های میان دو تاریخ مذکور
```

```
int totalSeconds=myDifference.Seconds; //تعداد ثانیه های میان دو تاریخ مذکور
```

```
int totalMilliseconds=myDifference.Milliseconds; //تعداد میلی ثانیه های میان دو تاریخ مذکور
```

کلاس TimeSpan متدهای مفید دیگه ای هم داره.

موفق باشید.

(با تشکر از danial82 عزیز که متذکر شدند دوست عزیزمون sinpin قبلاً این کلاس رو معرفی کرده بودند. از sinpin عزیز هم ممنونم.)

Register A Global HotKey

با استفاده از این روش بسیار ساده و عالی، بدون هیچ زحمت و توجع پیچیده و API های... و بدون هیچ نیازی به Hook کردن صفحه کلید و... برای برنامه تون یک کلید میانبر تعریف کنید که هر موقع و هر جای ویندوز که اون کلید فشار داده شد، براتون یک کاری انجام بده.

این کار خیلی ساده هست:

۱: یک پروژه ی ساده ی Windows Application ایجاد کنید.

۲: کلاس Form1 رو بدین ترتیب تغییر بدید:

```
public partial class Form1 : Form
{
    const int HOTKEY_ID = 31197;    //any number to be used as an id within this app

    const int WM_HOTKEY = 0x0312;

    public enum KeyModifiers        //enum to call 3rd parameter of RegisterHotKey easily
    {
        None = 0,

        Alt = 1,

        Control = 2,

        Shift = 4,

        Windows = 8
    }

    //API Imports
    [DllImport("user32.dll", SetLastError = true)]

    public static extern bool RegisterHotKey(

        IntPtr hWnd, // handle to window

        int id, // hot key identifier

        KeyModifiers fsModifiers, // key-modifier options

        Keys vk     // virtual-key code

    );

    [DllImport("user32.dll", SetLastError = true)]

    public static extern bool UnregisterHotKey(

        IntPtr hWnd, // handle to window

        int id       // hot key identifier

    );

    protected override void WndProc(ref Message msg)
```

```

{

    // Listen for operating system messages.

    switch (msg.Msg)
    {

        case WM_HOTKEY:

            // this is the block the app turns in if the hotkey has been pressed

            //so do your f@cking hotkey stuff here :-D

            MessageBox.Show("F11 Pressed!");

            break;

        }

        base.WndProc(ref msg);

    }

    //globalKeyboardHook myHookKeyBoard = new globalKeyboardHook();

    public Form1()
    {

        InitializeComponent();

        bool bcheck = RegisterHotKey(Handle, HOTKEY_ID, KeyModifiers.None, Keys.F11);

    }

}

```

خیلی کارهایش باز برای راحت تر شدن استفاده از این روش انجام داد. مثلاً اینکه بصورت یک DLL درش بیاریم، یا توئی یک کلاس بنویسیم و.....

کپی آیتم های یک کامبو باکس به دیگری

روش دیگر برای کپی آیتم های یک کامبو باکس به دیگری در موقعی که بخواهیم از کلاس استفاده کنیم

```

BindingSource bs = new BindingSource(comboBox1, "Items");

comboBox2.DataSource = bs

```

بدست آوردن زبان های نصب شده در سیستم:

```

string s;

foreach (InputLanguage Language in InputLanguage.InstalledInputLanguages)

```

```
{
s = Language.LayoutName;
listBox1.Items.Add(s);
}
```

غل چک باکس مثل رادیو باتون

```
private void checkBox2_CheckedChanged(object sender, EventArgs e)
{
if (checkBox2.Checked)
{
checkBox1.Checked = false;
// و بقیه
}
}

private void checkBox1_CheckedChanged(object sender, EventArgs e)
{
if (checkBox1.Checked)
{
checkBox2.Checked = false;
}
}
```

چینش تصادفی اعضای یک لیست یا آرایه به کمک LINQ

با فرض داشتن لیستی از اعداد مانند :

```
private List<int> numbers = new List<int> { 2, 12, 4, 6, 11 };
```

میتوانیم بنویسیم :

```
numbers = numbers.OrderBy(i => System.Guid.NewGuid()).ToList<int>();
```

ایجاد محدودیت کاراکتری با حد اقل کد

برای ایجاد محدودیت در نوع پذیرش کاراکتر در یک TextBox می توان از دوروداد keypress و keydown به این صورت استفاده کرد.

به عنوان مثال فقط می خواهیم اعداد پذیرفته شوند:

تعریف یک متغیر boolean در سطح فرم:

```
bool KeyHandle = false;
```

رخداد keydown :

```
private void Numeric_KeyDown(object sender, KeyEventArgs e)
```

```
{  
  
if ((e.KeyCode == Keys.Back) || (e.KeyCode >= Keys.D0 && e.KeyCode <= Keys.D9) || (e.KeyCode >=  
Keys.NumPad0 && e.KeyCode <= Keys.NumPad9))  
  
KeyHandle = false;  
  
else  
  
KeyHandle = true;  
  
}
```

حال در رخداد KeyPress براساس محتوای منطقی متغیر رخداد را Handle می کنیم:

```
private void Numeric_KeyPress(object sender, KeyPressEventArgs e)
```

```
{  
  
if (KeyHandle == true)  
  
{  
  
e.Handled = true;  
  
}
```

ایجاد محدودیت کاراکتری با حداقل کد (۲)

به منظور دریافت فقط کاراکتر عدد در کنترل‌های ویرایشی می‌توانید از کد زیر استفاده کنید. با استفاده از این کد معین می‌کنید که کنترل ویرایشی مورد نظر شما فقط عدد صحیح (غیر منفی) دریافت می‌کند یا مجوز درج و دریافت علامت ممیز (.) را نیز دارد یا خیر؟ برای اینکار یک کلاس استاتیک با نام Tools که سه مقدار استاتیک از (یک نوع String و دو نوع bool) ایجاد شده است.

```
static class Tools
```

```
{  
  
    // Boolean flag used to determine when a character other than a number is entered.  
  
    private static bool nonNumberEntered = false;  
  
    public static string text="0";  
  
    public static bool isDecimalAllow=false;
```

تابع زیر نوع کاراکتر ارسال شده به تاکس با کهایسی مشخص شده توسط شما را تعیین می‌کند//

```
public static void DetermineTypeOfCharacter(object sender, KeyEventArgs e)  
{  
  
    if (isDecimalAllow)  
  
        goto decimalAllow;  
  
    else goto decimalNotAllow;
```



```

decimalAllow:

    if (text.Contains("."))

    {

        // Initialize the flag to false.

        nonNumberEntered = false;

        // Determine whether the keystroke is a number from the top of the keyboard.

        if (e.KeyCode < Keys.D0 || e.KeyCode > Keys.D9)

        {

            // Determine whether the keystroke is a number from the keypad.

            if (e.KeyCode < Keys.NumPad0 || e.KeyCode > Keys.NumPad9)

            {

                // Determine whether the keystroke is a backspace.

                if (e.KeyCode != Keys.Back)

                {

                    // A non-numerical keystroke was pressed.

                    // Set the flag to true and evaluate in KeyPress event.

                    nonNumberEntered = true;

                    return;

                }

            }

        }

    }

    else

    {

        nonNumberEntered = false;

        if (e.KeyCode < Keys.D0 || e.KeyCode > Keys.D9)

        {

            if (e.KeyCode < Keys.NumPad0 || e.KeyCode > Keys.NumPad9)

            {

                // Determine whether the keystroke is a backspace.

                if (e.KeyCode != Keys.Back)

                {

```

```

        //Determine whether the keystroke is a Decimal.

        if (e.KeyCode != Keys.Decimal)

            // A non-numerical keystroke was pressed.

            // Set the flag to true and evaluate in KeyPress event.

            nonNumberEntered = true;

            return;

        }

    }

}

```

decimalNotAllow:

```

        nonNumberEntered = false;

        if (e.KeyCode < Keys.D0 || e.KeyCode > Keys.D9)

        {

            if (e.KeyCode < Keys.NumPad0 || e.KeyCode > Keys.NumPad9)

            {

                if (e.KeyCode != Keys.Back)

                {

                    nonNumberEntered = true;

                    return;

                }

            }

        }

    }

```

تابع زیر از ورود کاراکترهای غیر مجاز در داخل تکس باکسهای مورد نظر شما جلوگیری می کند//

```

public static void PreventCharacterEnter(object sender, KeyPressEventArgs e)

```

```

{

    // Check for the flag being set in the KeyDown event.

    if (nonNumberEntered == true)

    {

        // Stop the character from being entered into the control since it is non-numerical.

        e.Handled = true;

        System.Media.SystemSounds.Beep.Play();

    }

}

```

```

    }

    }

    در کد فوق دو تابع استاتیک (DetermineTypeOfCharacter(object sender, KeyEventArgs e) و public static void
    PreventCharacterEnter(object sender, KeyPressEventArgs e) نیز وجود دارد. خوب فقط لازم است برای هر کنترل ویرایشی مانند
    TextBox مقادیر مورد نظر خود را ست کرده (مقدار استاتیک Tools.text جهت ارسال تکس موجود در کنترل ویرایشی و تعیین مقدار true یا false
    برای مقدار استاتیک Tools.isDecimalAllow) و در رویداد keyDown کنترل مورد نظر خود تابع DetermineTypeOfCharacter(object
    sender, KeyEventArgs e) و در رویداد keyPress نیز تابع دوم یعنی public static void PreventCharacterEnter(object
    sender, KeyPressEventArgs e) را صدا بزنید. توجه داشته باشید که دو مقدار استاتیک text از نوع string و isDecimalAllow از نوع
    بولین را جهت مجوز دریافت علامت ممیز (.) یا عدم مجوز آن در رویداد keyDown (مشابه با کد زیر) ست کنید:

private void textBox1_KeyDown(object sender, KeyEventArgs e)

{

    Tools.text = this.modifyNumberTextBox.Text;

    Tools.isDecimalAllow = false;

    Tools.DetermineTypeOfCharacter(sender, e);

}

private void modifyNumberTextBox_KeyPress(object sender, KeyPressEventArgs e)

{

    Tools.PreventCharacterEnter(sender, e);

}

از مقدار nonNumberEntered که از نوع bool است جهت چک کردن مقدار معتبر استفاده می شود. در کد فوق از دو lable و دستور goto نیز
استفاده شده است.

ضمناً برای زیبایی کار در صورت عدم ورود کاراکتر صحیح در کنترل ویرایشی مورد نظر با استفاده از کد زیر در رویداد استاتیک
public static void PreventCharacterEnter(object sender, KeyPressEventArgs e) صدای Beep هشدار لازم را به کاربر اعلام می کند:

System.Media.SystemSounds.Beep.Play();

```

افزودن قطعه کدی به Toolbox

سلام

VS این امکان رو به شما میده که کدی که از اون زیاد استفاده می کنید رو به Toolbox اضافه کنید تا در مواقع بعدی با یک کلیک ساده بتونید ازش استفاده کنید .

برای اینکار کافیه در پنجره کدنویسی ، قطعه کد مورد نظرتون رو انتخاب کرده و اون رو به Toolbox درگ کنید . همچنین با Right Click روی گزینه اضافه شده و انتخاب Rename Item می تونید نام قطعه کد مورد نظر رو تغییر بدید .

مثالی ساده از LINQ ، که از یک آرایه اعداد زوج را انتخاب میکند :

```

int[] a = { 2, 1, 3, 6, 5, 8 };

var QResults = from n in a

where n % 2 == 0

```

```

select n;

string s = "";

foreach (var item in QResults)
{
    s += item.ToString()+"\n";
}

MessageBox.Show(s);

```

اینترفیس INotifyPropertyChanged برای ایجاد یک رویداد در یک جهت تشخیص پروپرتی تغییر یافته

اینترفیس مذکور دارای یک رویداد به نام PropertyChanged می باشد.

توسط این رویداد می توانید، رویدادی در کلاس خود داشته باشید که بتوانید تشخیص دهید که کدام یک از property ها مقدار آن تغییر کرده است.

در واقع به صورت اتوماتیک کلاینت شما متوجه خواهد شد که داده تغییر کرده است

مثال

```

class Student : System.ComponentModel.INotifyPropertyChanged
{
    private string name;

    public string Name
    {
        get
        {
            return name;
        }

        set
        {
            if (value != name)
            {
                name = value;

                OnPropertyChanged("Name");
            }
        }
    }

    private string family;

```

```

public string Family
{
    get
    {
        return family;
    }
    set
    {
        if(value != family)
        {
            family = value;
            OnPropertyChanged("Family");
        }
    }
}

#region INotifyPropertyChanged Members

public event System.ComponentModel.PropertyChangedEventHandler PropertyChanged;

#endregion

void OnPropertyChanged(string propertyName)
{
    if (this.PropertyChanged != null)
    {
        this.PropertyChanged(this, new
System.ComponentModel.PropertyChangedEventArgs(propertyName));
    }
}
}

```

آرگومان e ، دارای خاصیتی به نام propertyName می باشد، که توسط آن ، نام property که مقدار آن تغییر کرده است را می توانید بدست آورید.

به نحوه raise کردن این ایونت در پروپرتی ها دقت کنید.

محاسبه توان بدون استفاده از حلقه های تکرار و بازگشت

کدی که از فرمول ریاضی زیر پیروی می‌کند :

```
private void button1_Click(object sender, EventArgs e)
{
    int a = Convert.ToInt32(textBox1.Text);//Number 1
    int b = Convert.ToInt32(textBox2.Text);//Number 2
    double res = Math.Pow(Math.E, b * Math.Log(a, Math.E));//Result of Power
    MessageBox.Show("Result of Power is = " + res.ToString());
}
```

تکست باکسی که فقط عدد قبول میکند

```
private void textBox1_TextChanged(object sender, EventArgs e)
{
    string str = textBox1.Text;
    int f=0;
    foreach (char c in str)
    {
        if (char.IsDigit(c) == true)
        {
            f++;
        }
        else
        {
            f--;
        }
    }
    if (f != str.Length)
    {
        MessageBox.Show("Error");
        textBox1.Clear();
    }
}
```

اضافه کردن شماره سطر به دیتا گرید ویو

در اونت RowPostPaint اگر دیتا گرید ویو تان RightToLeft بود بنویسید:

```
using (SolidBrush b = new SolidBrush(Color.Black))
{
    e.Graphics.DrawString((e.RowIndex).ToString(),
```

```

        e.InheritedRowStyle.Font, b,

        (sender as DataGridView).Width - e.RowBounds.Location.X - 15,

        e.RowBounds.Location.Y + 4);

    }

```

و اگر چپ به راست بود بنویسید:

```

using (SolidBrush b = new SolidBrush(Color.Black))

{

    e.Graphics.DrawString((e.RowIndex).ToString(),

    e.InheritedRowStyle.Font, b,

    e.RowBounds.Location.X + 15,

    e.RowBounds.Location.Y + 4);

}

```

نشان دادن کامل تاریخ و زمان سیستم با متود To String:

```

DateTime mytime = new DateTime();

string time;

mytime = DateTime.Now;

time = mytime.ToString("F");

MessageBox.Show(time);

```

تولید رشته های تصادفی بصورت کاملاً دستی

سلام دوستان ، این برنامه که نوشتم ، رشته های تصادفی بصورت کاملاً دستی ، تولید میکنه با هر طولی و بدون استفاده از امکانات سی شارپ !!!!

```

public string txt_Generator(int size)

{

    int[] a = new int[size];

    Random rnd = new Random();

    for (int i = 0; i < size; i++)

    {

        a[i] = 65 + rnd.Next() % 26;

    }

    string str = "";

    foreach (int j in a)

```

```
{
str += (char)(j);
}
return str;
}
```

انتخاب و فعال شدن یک سطر در dataGridView با مقدار خاص

```
for (int i = 0; i < dataGridView1.Rows.Count-1; i++)
{
    if (dataGridView1.Rows[i].Cells[ایندکس ستون مورد جستجو].Value.ToString() == "مقدار مورد
    نظر")
    {
        dataGridView1.Rows[i].Selected = true;
    }
}
```

```
for (int i = 0; i < dataGridView1.Rows.Count-1; i++)
{
    if (dataGridView1.Rows[i].Cells[2].Value.ToString() == "Johnson")
    {
        dataGridView1.Rows[i].Selected = true;
    }
}
```

پاک کردن هم زمان تمام TextBox ها:

بعضی وقتا تعداد TextBox ها نوی فرمتون خیلی زیاده و سخته که هر بار تمام اونها رو پاک کنید پس از روش زیر استفاده کنید تا تمام TextBox ها را یکجا Clear کنید.

```
public void ClearTextBoxs()
{
    foreach (Control txt in this.Controls)
    {
        if (txt.GetType().Name == "TextBox")
```



```

        txt.Text = "";
    }
}

```

حل مشکل رسم اشکال گرافیکی در سی شارپ:

شاید شما هم وقتی در NET برنامه نویسی گرافیکی می کنید به این مشکل برخورد داشته باشید.

وقتی می خواهید روی یک فرم یا هر کنترل دیگری چیزی رسم کنید ابتدا با استفاده از متد `CreateGraphics` آن را صدا می زنید و سپس چیزی روی آن رسم می کنید. یعنی:

```

Graphics gr = form1.CreateGraphics();

// you draw something here

```

ولی پس از رسم مشاهده می کنید که به یک بار رفت و آمد صفحه همه چیز پاک شده است .

برای حل این مشکل کافیست به صورت زیر عمل کنید :

```

Bitmap bmp = new Bitmap(form1.Width, form1.Height);

Graphics gr = Graphics.FromImage((Image)bmp);

gr.Clear(Color.White);

// draw something in gr

gr.Dispose();

form1.BackgroundImage = bmp;

```

بدست آوردن لیست برنامه های نصب شده بر روی سیستم

```

private string GetInstalledSoftware()
{
    string Software = null;

    string SoftwareKey = @"SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall";

    using (RegistryKey rk = Registry.LocalMachine.OpenSubKey(SoftwareKey))
    {
        foreach (string skName in rk.GetSubKeyNames())
        {
            using (RegistryKey sk = rk.OpenSubKey(skName))
            {
                try
                {
                    if (!(sk.GetValue("DisplayName") == null))

```

```

        {

            if (sk.GetValue("InstallLocation") == null)

                Software += sk.GetValue("DisplayName") + " - Install path not known\n";

//Nope, not here.

            else

                Software += sk.GetValue("DisplayName") + " - " +
sk.GetValue("InstallLocation") + "\n"; //Yes, here it is...

        }

    }

    catch (Exception ex)

    {

        //No, that exception is not getting away... :P

    }

}

}

return Software;

}

```

تست ارتباط با اینترنت از طریق پینگ کردن

```

public bool isConnectionAvailable()

{

    //build a list of sites to ping, you can use your own

    string[] sitesList = { "www.google.com", "www.microsoft.com" , "www.psychocoder.net" };

    Ping ping = new Ping();

    PingReply reply;

    bool _success = false;

    int notReturned = 0;

    try

    {

        for (int i = 0; i <= sitesList.Length; i++)

        {

            reply = ping.Send(sitesList[i], 10);

            if (reply.Status != IPStatus.Success)

```

```

        {
            notReturned += 1;
        }

        if (notReturned == sitesList.Length)
        {
            _success = false;

            throw new Exception(@"There doest seem to be a network/internet connection.\r\n
            Please contact your system administrator");
        }
        else
        {
            _success = true;
        }
    }
}

catch
{
    _success = false;
}

return _success;
}

```

بدست آوردن سطر جاری در دیتا گرید ویو

```

public DataRow GetCurrentRow(DataGridView dgv)
{
    DataRowView drv = null;

    try
    {
        if (dgv.CurrentRow == null) { return null; }

        if (dgv.CurrentRow.DataBoundItem == null) { return null; }

        drv = (DataRowView)dgv.CurrentRow.DataBoundItem;
    }
}

```

```

catch
{
    return null;
}

return drv.Row;
}

```

بدست آوردن میزان زمان روشن بودن کامپیوتر

```

public string getUptime()
{
    String strResult = String.Empty;

    strResult += Convert.ToString(Environment.TickCount / 86400000) + " days, ";

    strResult += Convert.ToString(Environment.TickCount / 3600000 % 24) + " hours, ";

    strResult += Convert.ToString(Environment.TickCount / 120000 % 60) + " minutes, ";

    strResult += Convert.ToString(Environment.TickCount / 1000 % 60) + " seconds.";

    return strResult;
}

```

ارسال و اجرای مستقیم کوئری در LINQ To SQL

شما میتوانید صرفنظر از تواناییهای LINQ To SQL کوئریهای دلخواه خود را مستقیم برای SQL Server ارسال نمایید :

```

Northwnd db = new Northwnd(@"c:\northwnd.mdf");

IEnumerable<Customer> results = db.ExecuteQuery<Customer>

(@"SELECT c1.custid as CustomerID, c2.custName as ContactName

FROM customer1 as c1, customer2 as c2

WHERE c1.custid = c2.custid"

);

```

تشخیص روشن یا خاموش بودن کلیدهای CapsLock, NumLock, ScrollLock به روشی بسیار ساده

```

Microsoft.VisualBasic.Devices.Keyboard keyboard = new Microsoft.VisualBasic.Devices.Keyboard();

string COnOff = keyboard.CapsLock ? "On" : "Off";

```

```
string NOnOff = keyboard.NumLock ? "On" : "Off";

string SOnOff = keyboard.ScrollLock ? "On" : "Off";

MessageBox.Show("Caps Lock is " + COnOff + "\nNum Lock is " + NOnOff + "\nScroll Lock is " + SOnOff);
```

فقط یک نمونه از یک فرم اجرا شود:

گاهی می‌خواهیم فقط یک نمونه از یک فرم اجرا شود.

برای اینکار تابعی نوشتیم که در هنگام باز کردن فرم نام فرممان را به تابع ارسال می‌کنیم، اگر فرم بسته باشد که آنرا باز میکند اما اگر باز باشد آنرا به اولین لایه ممکن انتقال می‌دهد.

```
Boolean IsRunning(string formName)

{

    foreach (Form frm in Application.OpenForms)

    {

        if ((frm.Name == formName))

        {

            frm.BringToFront();

            return true;

        }

    }

    return false;

}
```

مثال از نحوه استفاده:

```
private void OpenForm2_Click(object sender, EventArgs e)

{

    Form2 af = new Form2();

    if (!IsRunning("Form2"))

        af.Show();

}
```

تبدیل مبنای استفاده از Convert.ToString

پارامتر اول عدد شما است و پارامتر دوم مبنای مورد نظر که می‌تواند ۲، ۸، ۱۰ و ۱۶ باشد.

```
long a = 9999999999999999999;

string b = Convert.ToString(a, 2);

Console.WriteLine(b);
```

```
Console.WriteLine(Convert.ToString(a, 8));  
  
Console.WriteLine(Convert.ToString(a, 16));  
  
Console.ReadKey();
```

اجرای یک متود تنها با دانستن نام آن (بصورت رشت)

برای مثال در تیکه کد زیر دو متود Method1 و Method2 اجرا میشوند :

```
public partial class Form1 : Form  
{  
    public Form1()  
    {  
        InitializeComponent();  
    }  
  
    public void Method1()  
    {  
        MessageBox.Show("Method1 invoked");  
    }  
  
    public void Method2()  
    {  
        MessageBox.Show("Method2 invoked");  
    }  
  
    private void Form1_Load(object sender, EventArgs e)  
    {  
        MethodInvoke();  
    }  
  
    private void MethodInvoke()  
    {  
        Type t = typeof(Form1);
```

```

int i = 0;

i++;

string MethodName = "Method" + i.ToString();

t.InvokeMember(MethodName, BindingFlags.Default | BindingFlags.InvokeMethod, null, this, null);

i++;

MethodName = "Method" + i.ToString();

t.InvokeMember(MethodName, BindingFlags.Default | BindingFlags.InvokeMethod, null, this, null);

}

}

```

آموزش کار کردن با BackgroundWorker بصورت آسان و قدم به قدم

این آموزش همراه با یک مثال انجام میشود.

(توجه : در بهینه نبودن این مثال شک نداشته باشید چون فی البداهه و صرف آموزش نوشته شده است)

مثال - میخواهیم تمامی فایل‌های موجود در پوشه ی ویندوز را توسط یک thread پشت زمینه (مستقل از UI Thread) در یک لیست باکس نمایش دهیم.

بلافاصله و بدون مقدمه؛ ابتدا یک آبجکت از نوع BackgroundWorker بسازید و در زمان کلیک شدن یک دکمه مقادیر آن را به شکل زیر تغییر دهید :

```
BackgroundWorker worker = new BackgroundWorker();
```

```
private void button1_Click(object sender, EventArgs e)
```

```

{

    worker.WorkerReportsProgress = true;

    worker.DoWork += new DoWorkEventHandler(worker_DoWork);

    worker.ProgressChanged += new ProgressChangedEventHandler(worker_ProgressChanged);

    worker.RunWorkerAsync(new DirectoryInfo(@"C:\Windows"));

}

```

خط آبی شده حاکی از آن است که worker میتواند در زمان اجرا میزان پیشرفت خود را گزارش کند(که باعث توانایی کار کردن با عناصر UI بصورت مستقیم و بدون استفاده از delegate میشود)

سپس ایونت هندلرها رو تخصیص داده

و در خط آخر نیز دستور اجرای آسنکرون اون رو با یک پارامتر ورودی را صادر کردیم.

کار زمانبری که میخواهیم در پشت زمینه انجام شود را در رویداد DoWork مینویسیم و این همان رویدادی ست که با دستور worker.RunWorkerAsync در تیکه کد قبلی تحریک میشود :

```

void worker_DoWork(object sender, DoWorkEventArgs e)
{
    DirectoryInfo primaryDir = e.Argument as DirectoryInfo;

    int i = 0;

    foreach (DirectoryInfo dir in primaryDir.GetDirectories())
    {
        foreach (FileInfo file in dir.GetFiles())
        {
            worker.ReportProgress(i++, file.FullName);
        }
    }
}

```

همانطور که مشاهده کردید در خط اول متود بالا پارامتر ورودی را در قالب یک DirectoryInfo در آوردیم که بتوانیم از آن استفاده کنیم.

و به ازای هر فایل شناخته شده (البته در عمل : بهتر است اینکار مثلاً به ازای هر ۱۰۰ فایل پیدا شده انجام شود) یک گزارش از میزان پیشرفت همراه با پارامترهای شماره فایل و نام آن میفرستیم.

و در نهایت برای منعکس کردن پیشرفتهای لحظه ای و اختصاص آنها به لیست باکس مینویسیم:

```

void worker_ProgressChanged(object sender, ProgressChangedEventArgs e)
{
    listBox1.Items.Add(e.ProgressPercentage.ToString() + " : " + e.UserState.ToString());
}

```

توجه کنید که در کد بالا مقدار e.UserState همان آبجکتی است که در زمان DoWork ارسال کردید.

ترسیم و نوشتن مستقیم بردسکتاپ ویندوز

```

using System.Drawing;

using System.Drawing.Drawing2D;

using System.Runtime.InteropServices;

[DllImport("user32")]

internal static extern IntPtr GetDC(IntPtr hwnd);

[DllImport("User32.dll")]

internal static extern void ReleaseDC(IntPtr dc);

public void PaintRectangleToScreen()
{

```



```

IntPtr deskDC = GetDC(IntPtr.Zero);

Graphics g = Graphics.FromHdc(deskDC);

Font font = new Font("Arial Black", 36);

DrawStringOnCenter(g, "Sale 1387 mobarak!", font, new Point(0, -25));

font = new Font("Arial Black", 18);

DrawStringOnCenter(g, "by: Sinpin", font, new Point(0, 25));

Rectangle rect = new Rectangle(200, 300,

    Screen.PrimaryScreen.Bounds.Width - 400,

    Screen.PrimaryScreen.Bounds.Height - 600);

g.FillRectangle(new SolidBrush(Color.FromArgb(100, Color.DodgerBlue))

    , rect);

g.DrawRectangle(new Pen(Color.DodgerBlue, 3), rect);

g.Dispose();

ReleaseDC(deskDC);
}

private void DrawStringOnCenter(Graphics g, string str, Font font, Point offset)
{
    SizeF size = g.MeasureString(str, font);

    g.DrawString(str, font, Brushes.White,

        new PointF(

            (Screen.PrimaryScreen.Bounds.Width - size.Width) / 2 + offset.X,

            (Screen.PrimaryScreen.Bounds.Height - size.Height) / 2 + offset.Y

        ));
}

```

مثالی از استفاده:

```

private void Form1_Load(object sender, EventArgs e)
{
    PaintRectangleToScreen();

    this.Close();
}

```

آشنایی با الگوهای طراحی: Singleton Pattern

هدف : می‌خواهیم تضمین کنیم که از یک کلاس تنها بتوان یک آبجکت (حالت ساده و متعارف سینگلتون) ایجاد کرد و همه ی درخواستها نیز تنها به همان یک آبجکت هدایت شوند و نیز بتوان یک دسترسی عمومی به آن داشت.

اولین قدم : سازنده ی کلاس را private می‌کنیم

چرا private ؟ - پاسخ : تا هیچکسی نتواند از کلاس آبجکت بسازد.

دومین قدم : یک فیلد استاتیک از نوع خود کلاس تعریف می‌کنیم.

چرا استاتیک : چون قرار است در یک متود استاتیک از آن استفاده کنیم.

چرا private : چون قرار است امکان دسترسی به این آبجکت از طریق یک متود یا پراپرتی صورت گیرد و دسترسی مستقیم مطلوب نیست.

سومین قدم : ایجاد یک پراپرتی یا متود public و استاتیک که فیلد را برمیگرداند.

چرا public ؟ چون می‌خواهیم از همه جا قابل دسترسی باشد.

چرا استاتیک ؟ چون آبجکتی از کلاس ندارم و ناچاراً باید دسترسی به این متود یا پراپرتی در سطح کلاس باشد.

پس تا اینجا داریم :

```
public sealed class Singleton
{
    // Private Constructor
    Singleton() { }

    // Private object instantiated with private constructor
    static readonly Singleton instance = new Singleton();

    // Public static property to get the object
    public static Singleton UniqueInstance
    {
        get { return instance; }
    }
}
```

معمولاً بهتر است این پیاده سازی بصورت lazy instantiation باشد (یعنی تا قبل از اولین invoking آبجکت مقداردهی اولیه نشود) :

```
public class Singleton
{
    // Private constructor
    Singleton() { }
```

```
// Nested class for lazy instantiation

class SingletonCreator
{
    static SingletonCreator() { }

    // Private object instantiated with private constructor
    internal static readonly
    Singleton uniqueInstance = new Singleton();
}

// Public static property to get the object
public static Singleton UniqueInstance
{
    get { return SingletonCreator.uniqueInstance; }
}
}
```

این الگو به شدت در طراحیهای مختلف و الگوهای دیگر (مانند: Façade, Abstract Factory, ...) مورد استفاده دارد.

تبدیل اتوماتیک دکمه Enter به Tab جهت انتقال فوکوس دکترها

این کد را در کلاس مربوط به کنترل TextBox خود بنویسید :

```
protected override bool ProcessDialogKey(Keys keyData)
{
    if (keyData == Keys.Return)
        SendKeys.Send("{TAB}");
    return base.ProcessDialogKey(keyData);
}
```

و در مورد استفاده دکترهای استاندارد:

```
private void textBox1_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Return)
        SendKeys.Send("{TAB}");
}
```

تعدادی کد اعتبار سنجی (Validation) برای TextBox

TextBox عددی:

```
private void textBox1_KeyPress(object sender, System.Windows.Forms.KeyPressEventArgs e)
{
    if ( !( char.IsDigit( e.KeyChar ) || char.IsControl( e.KeyChar ) ) )
    {
        e.Handled = true;
    }
}
```

TextBox عددی با اعشار:

```
private void textBox1_KeyPress(object sender, System.Windows.Forms.KeyPressEventArgs e)
{
    if ( !( char.IsDigit( e.KeyChar ) || char.IsControl( e.KeyChar ) || (e.KeyChar== (char)46) ) )
    {
        e.Handled = true;
    }
}
```

TextBox فقط کاراکتری:

```
private void textBox1_KeyPress(object sender, System.Windows.Forms.KeyPressEventArgs e)
{
    if ( !( char.IsLetter( e.KeyChar ) || char.IsControl( e.KeyChar ) ) )
    {
        e.Handled = true;
    }
}
```

TextBox برای فقط حروف بزرگ:

```
private void textBox1_KeyPress(object sender, System.Windows.Forms.KeyPressEventArgs e)
{
    if ( !( char.IsUpper( e.KeyChar ) || char.IsControl( e.KeyChar ) ) )
    {
        e.Handled = true;
    }
}
```

TextBox برای فقط حروف کوچک:

```
private void textBox1_KeyPress(object sender, System.Windows.Forms.KeyPressEventArgs e)
{
    if ( !( char.IsLower( e.KeyChar ) || char.IsControl( e.KeyChar ) ) )
    {
        e.Handled = true;
    }
}
```

چک کردن TextBox های خالی (می توانید کلیه TextBox های روی فرمتان را به آن ارسال کنید)

```
public static bool ChkEmpty(params System.Windows.Forms.TextBox[] tb) {    int i;    for (i = 0; i < tb.Length; i++)
{
    if (tb[i].Text.Trim() == "")
    {
        MessageBox.Show("Don't keep field empty");
        tb[i].Focus();
        return false;
    }
}
return true;
}
```

TextBox اعشاری (با علامت اعشار مربوط به هر منطقه)

```
string DecimalSeparator = Thread.CurrentThread.CurrentCulture.NumberFormat.NumberDecimalSeparator;

private void textBox1_KeyPress(object sender, System.Windows.Forms.KeyPressEventArgs e)
{
    if ( !( char.IsDigit( e.KeyChar ) || char.IsControl( e.KeyChar ) || (DecimalSeparator.IndexOf(e.KeyChar)
    != -1 ) ) )
    {
        e.Handled = true;
    }
}
```

ساختن Connection String به روشی بسیار ساده

برای ساخت یک Connection String بدون نیاز به Visual Studio ، به روش زیر عمل می‌کنیم:

یک فایل جدید ایجاد کرده و پسوند آنرا به "udl" تغییر می‌دهیم. (مثلا یک فایل متنی که می‌شود New Text file.udl)

می‌بینید که آیکون آن تغییر کرد

آنرا باز کنید. پنجره مشخصات Data Link باز می‌شود

برای ایجاد Connection String برای SQL گزینه Microsoft OLE DB Provider For SQL Server را از لیست انتخاب کنید

دکمه Next را زده و یا به لیست Connection بروید

از این قسمت تمامی مشخصه‌های ارتباط را می‌توانید تعیین کنید

بعد از مشخص کردن تمامی پارامترهای مورد نیاز دکمه Test Connection را برای اطمینان از درستی آن بزنید. اگر همه چیز درست است پنجره را ببندید

حال فایلی که ابتدا ایجاد کردید را با Notepad باز کنید.

Connection String درون فایل ایجاد شده است. (البته این ارتباط برای OLE DB است که می‌توانید این قسمت را حذف کنید)

اگر از User و Password استفاده می‌کنید، تیک Allow Saving Password را بزنید تا پسورد در فایل ذخیره شود.

اطلاعاتی در مورد موس

شاید نیاز داشته باشید بفهمید کاربر از موس استفاده می‌کند یا نه؛ و در صورتی که استفاده می‌کند اطلاعاتی از موس (تعداد دکمه، Wheel)

این اطلاعات و یا سایر اطلاعاتی را که از سیستم کاربر می‌خواهید می‌توانید در کلاس Enviroment بیابید.

```
SystemInformation.MousePresent.ToString()

SystemInformation.MouseButtons.ToString()

SystemInformation.MouseButtonsSwapped.ToString()
```

```
SystemInformation.MouseWheelPresent.ToString()
```

فراخوانی یک رویداد از رویداد دیگر

```
private void Form1_Load(object sender, EventArgs e)
{
    MessageBox.Show(sender.ToString());
}

private void button1_Click(object sender, EventArgs e)
{
    System.EventArgs arg = new System.EventArgs();
    Form1_Load(button1.Text, arg);
}
```

دستیابی به تاریخ شمسی با استفاده از PersianCalender

```
using System.Globalization;

public string Miladi2Shamsi(DateTime _date)
{
    PersianCalendar pc = new PersianCalendar();
    StringBuilder sb = new StringBuilder();
    sb.Append(pc.GetYear(_date).ToString("0000"));
    sb.Append("/");
    sb.Append(pc.GetMonth(_date).ToString("00"));
    sb.Append("/");
    sb.Append(pc.GetDayOfMonth(_date).ToString("00"));
    return sb.ToString();
}
```

نحوه ی استفاده، برای مثال تاریخ امروز :

```
string today = Miladi2Shamsi(DateTime.Now);
```

خروجی این تابع رشته ای در قالب "YYYY:MM:DD" مانند : "۲۵/۱۲/۱۳۸۶" خواهد بود.

تبدیل رشته حاوی تاریخ شمسی به تناظر میلادی آن

```
using System.Globalization;

public DateTime Shamsi2Miladi(string _date)
{
    int year = int.Parse(_date.Substring(0, 4));
    int month = int.Parse(_date.Substring(5, 2));
    int day = int.Parse(_date.Substring(8, 2));
    PersianCalendar p = new PersianCalendar();
    DateTime date = p.ToDateTime(year, month, day, 0, 0, 0, 0);
    return date;
}
```

نمونه از برنامه تولد اجرا شود (با استفاده از WMI)

ابتدا System.Management رو به References پروژه بیفزایید و سپس :

```
using System.Reflection;
using System.Management;

[STAThread]
static void Main()
{
    string[] parts =
        Assembly.GetExecutingAssembly().Location.Split("\\".ToCharArray());
    string appName = parts[parts.Length - 1];
    string query = "select name from CIM_Process where name = '"
        + appName + "'";
    ManagementObjectSearcher searcher = new ManagementObjectSearcher(query);
    int runcount = 0;
    foreach (ManagementObject item in searcher.Get())
    {
```



```

        runcount++;

        if (runcount > 1) break;
    }

    if (runcount == 1)
    {
        Application.Run(new Form1());
    }
    else
    {
        System.Windows.Forms.MessageBox.Show("قبلا اجرا شده");
        Application.Exit();
    }
}

```

جلوگیری از رخ دادن Unhandled Exceptions

در فایل program.cs مربوط به پروژه بعد از دستورات زیر:

```

Application.EnableVisualStyles();
Application.SetCompatibleTextRenderingDefault(false);

```

دستور زیر رو قرار بدید:

```

Application.ThreadException += new ThreadExceptionHandler(Application_ThreadException);

```

و در همین فایل program.cs متدی با همان نامی که در سازنده بالا دیدیم باید ایجاد کنید مثل متد زیر:

```

static void Application_ThreadException(object sender, ThreadExceptionEventArgs e)
{
    MessageBox.Show("خطا در اجرای دستور کاربر", "خطا");
}

```

از این به بعد در صورتی که خطایی در قطعه کدی از برنامه رخ بده که درون بلوک try قرار نداشته باشه به این متد فرستاده می شه و دیگه پیام پیش فرض دات نت فریم ورک نمایش داده نمی شه.

نکته: آرگومان e در متد فوق حاوی Exception ایجاد شده می باشد که از آن می توان استفاده های مختلفی کرد.

جایگزین کردن یک رشته درون متن یک textbox به روش اندیس دهی

با فشار دادن دکمه، زیررشته ی موجود در تکس باکس که از اندیس ۱۲ شروع می شود با *** جایگزین می شود

```
private void button1_Click(object sender, EventArgs e)
{
    textbox1.SelectionStart = 0;
    textbox1.SelectionLength = textbox1.Text.Length;
    textbox1.Text = textbox1.SelectedText.Insert(12, "***");
}
```

آغاز کار با کامپوننت ErrorProvider

از این کامپوننت برای نمایش پیغامهای خطا و هشدار در UI استفاده میشود.

```
private void textBox1_Leave(object sender, EventArgs e)
{
    ErrorProvider ep = new ErrorProvider();
    if (string.IsNullOrEmpty(textBox1.Text))
        ep.SetError(textBox1, "نمیتواند خالی باشد");
    else
        ep.SetError(textBox1, "");
}
```

برعکس کردن ترتیب آیتمهای یک آرایه

```
int[] someArray = new int[5] { 1, 2, 3, 4, 5 };
Array.Reverse(someArray);
```

Hash کردن یک رشته متنی با الگوریتم MD5

```
using System.Security.Cryptography;

private string encryptString(string strToEncrypt)
{
    UTF8Encoding ue = new UTF8Encoding();
    byte[] bytes = ue.GetBytes(strToEncrypt);
```

```

MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();

byte[] hashBytes = md5.ComputeHash(bytes);

// Bytes to string

return System.Text.RegularExpressions.Regex.Replace

    (BitConverter.ToString(hashBytes), "-", "").ToLower();

}

```

استفاده:

```

MessageBox.Show(encryptString("Sinpin"));

```

خروجی:

```

e140088b10c4f19588ed3624e2b3c463

```

به دست آوردن مقدار یک Enum از عدد متناظر آن

```

public enum Days { Sat = 1, Sun, Mon, Tue, Wed, Thu, Fri };

string s= Enum.GetName(typeof(Days),4);

```

کار با اشیاء بوسیده می نام آنها (با کمک از Reflection)

مثال - نمایش یک فرم با دانستن نام آن:

(این مثال کاربردهای زیادی میتواند داشته باشد، فرض کنید که نام فرمهای مورد نیاز برای کاربردی خاص رو در یک جدول دیتابیس ذخیره کرده اید)

```
using System.Reflection;
```

```
private void button1_Click(object sender, EventArgs e)
{
    Assembly myAssembly = Assembly.GetExecutingAssembly();

    Form myForm =
        myAssembly.CreateInstance("WindowsFormsApplication1.Form2") as Form;

    myForm.ShowDialog();
}
```

نکته : حتما نام Namespace را قبل از نام آبجکت قرار دهید؛ در مثال بالا "WindowsFormsApplication1" بود.

ارسال تعداد متغیر پارامتر به یک متود

با استفاده از کلمه کلیدی params مانند نمونه زیر :

```
public int Add(params int[] list)
{
    int sum = 0;

    foreach (int i in list)
        sum += i;

    return sum;
}
```

حال میتوانیم این متود را به این شکل صدا بزنیم :

```
int ans1 = Add(1);
```

```
int ans2 = Add(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);
```

```
int ans3 = Add(new int[] { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 });
```

```
int ans4 = Add();
```

نکته: متود میتواند چندین پارامتر دیگر غیر از پارامتر از نوع params بپذیرد اما پارامتر params حتما باید آخرین پارامتر باشد. مانند:

```
public void AddAndPrint(string str, params int[] list)
{
    int sum = 0;

    foreach (int i in list)
        sum += i;

    Console.WriteLine(str + sum);
}
```

refresh کردن صفحه نمایش (desktop)

ابتدا بالای فرم ، کد زیر را وارد نمایید:

```
[System.Runtime.InteropServices.DllImport("Shell32.dll")]
```

```
private static extern int SHChangeNotify(int eventId, int flags, IntPtr item1, IntPtr item2);
```

سپس در دکمه ای که می خواهد refresh کند بنویسید:

```
SHChangeNotify(0x8000000, 0x1000, IntPtr.Zero, IntPtr.Zero);
```

روش برای تشخیص عدد بودن یک string

```
bool str_is_digit(string str)
{
    bool res = true;

    foreach (char c in str)
    {
        if (!char.IsDigit(c))
        {
            res = false;
            break;
        }
    }

    return res;
}
```

پایان

www.EhsanAvr.Com