

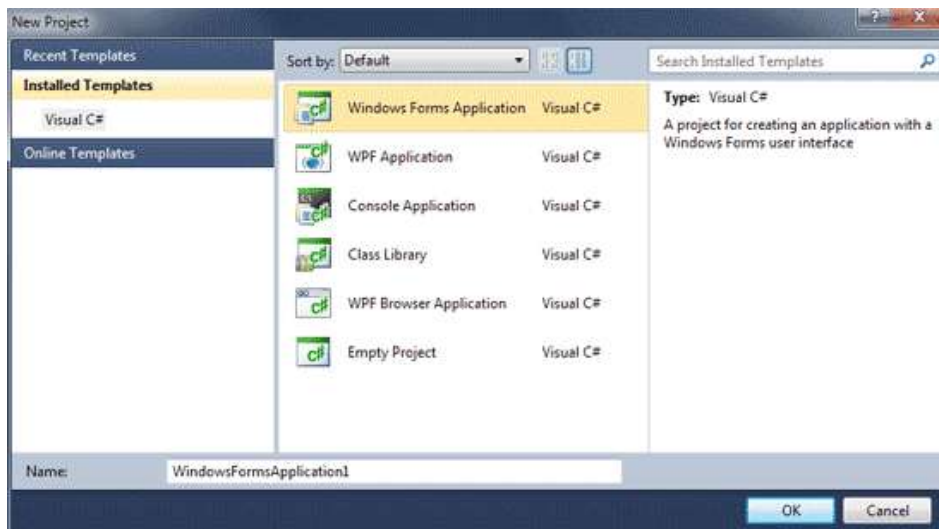
در بخش پیش از آموزش زبان سی شارپ، شما اولین کد خود را در محیط زیبا و حرفه‌ای Visual Studio نوشتید. اگر چه دیدن کامپایل شدن برنامه یک حس خوبی دارد، اما قطعاً هنوز اول راه هست و ما تنها یک رشته (=چند کاراکتر کنار هم) را کامپایل کردیم.

در این بخش از آموزش مطابق با قولی که قبلاً به شما داده بودم، از محیط خسته‌کننده Console جدا می‌شویم و به محیط جذاب و کاربردی Windows Form Application روی می‌آوریم. با یک بسم‌الله شروع می‌کنم.

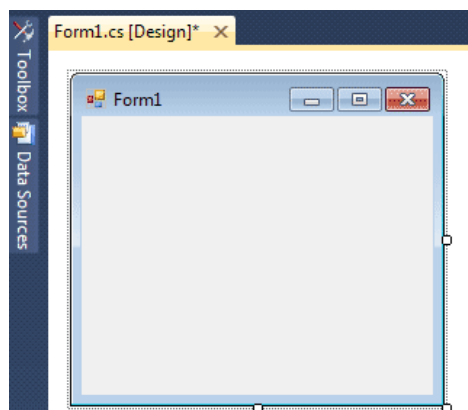
## • اولین Windows Form Application شما!

در Windows Form Application، محیطی به نام Form وجود دارد که تقریباً تمامی کدنویسی‌های شما در پشت پرده‌ی آن انجام خواهد گرفت. در ابتدا این Form برای شما کاملاً خالی می‌باشد و این شما ی برنامه‌نویس هستید که می‌بایست توسط ابزارها و کدها، برحسب نیازهای خودتان، از آن یک برنامه‌ی کاربردی بسازید. می‌دانید که اساساً در Formها، ابزارهایی همانند جعبه متن (Text Box)، لیبل (Label)، منوها (Menus) و ... قرار می‌گیرند. سی شارپ تمامی این ابزارهای اساسی را برای شما مهیا می‌سازد و شما کفایت آن‌ها را کد نویسی کنید. البته که در مراحل حرفه‌ای کار، شما می‌توانید ابزارهای مورد نیاز خودتان را بسازید و یا با استفاده از componentهای آماده، نیازهایتان را برطرف کنید.

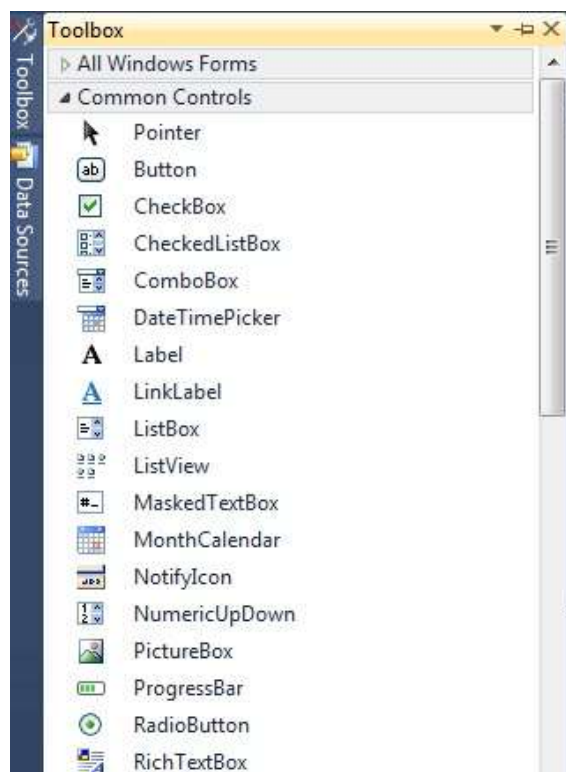
خوب، اکنون به سراغ ساختن یک پروژه‌ی جدید با استفاده از Windows Form Application (یا همان Windows Application) می‌پردازیم. برای شروع، پس از اجرای نرم‌افزار Visual Studio، از منوی بالا، گزینه‌ی Project New را از File انتخاب کنید. پنجره‌ای همانند زیر را مشاهده خواهید کرد. (در نسخه‌های مختلف Visual Studio، این پنجره دارای ظاهرهای متفاوتی می‌باشد، اما ماهیت همه‌ی آن‌ها یکی است.)



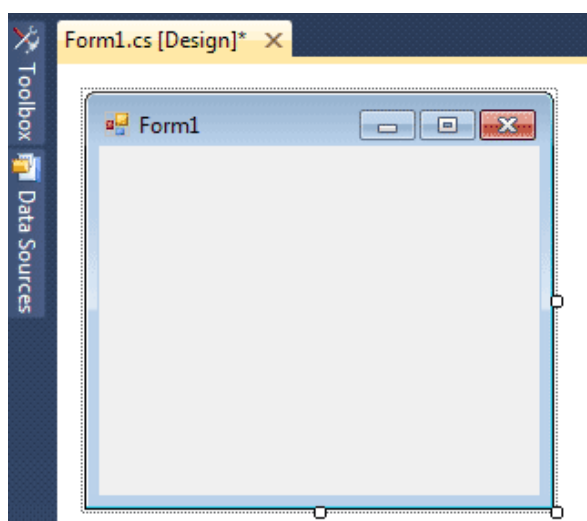
از پنجره‌ی باز شده، بر روی Windows Forms Application کلیک کرده تا انتخاب شود. در بخش نام پروژه، همان نام پیشفرض را، یعنی WindowsFormsApplication1، قبول کنید و با کلیک بر روی OK، به ادامه فرآیند پردازش بپردازید. محیط زیر را خواهید دید:



تفاوت کاملاً مشخصی که میان پروژه‌های Console Application و Windows Form Application وجود دارد، وجود یک Form خالی می‌باشد. در سمت چپ، به جعبه ابزارها (Toolbox) دقت کنید. ما توسط این بخش، کنترل‌های لازم را در Form ایجاد می‌کنیم و در نهایت، آن‌ها را کدنویسی خواهیم کرد.



اگر که احياناً شما نمی‌توانید Toolbox را ببینید، احتمالاً شما یک پنجره‌ی خالی همانند زیر خواهید دید.



اگر چنین هست، موس خود را بر روی سربرگ Toolbox که در سمت چپ هست، نگاه دارید. مشاهده خواهید کرد که Toolbox مجدداً ظاهر می‌شود. برای این‌که حالت نمایش آن را دائمی کنید، می‌توانید بر روی آیکون مشخص شده (Pin) کلیک کنید تا نمایش آن به‌صورت ثابت بماند.



اکنون به بخش Solution Explorer یک نیم‌نگاهی بیاندازید! اگر آن را با Solution Explorer پروژه قبلی که بصورت Console Application بود مقایسه کنید، متوجه خواهید شد که تمامی آیتم‌ها همانند هم بوده؛ تنها در Windows Form Application یک آیتمی به نام Form مشاهده خواهید کرد.



اکنون بر روی Program.cs دابل-کلیک کنید. کدهای آشنایی را خواهید دید.

```

using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace WindowsApplication1
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}

```

عکس زیر، از فایل Program.cs پروژه Console Application است

```

using System;
using System.Collections.Generic;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
        }
    }
}

```

هر دوی آن‌ها همان خط‌های شروع شده با Using را دارند، یک namespace مشابه وجود دارد، کلاسی با نام Program نیز همچنان هست و متد (تابعی) با نام Main.

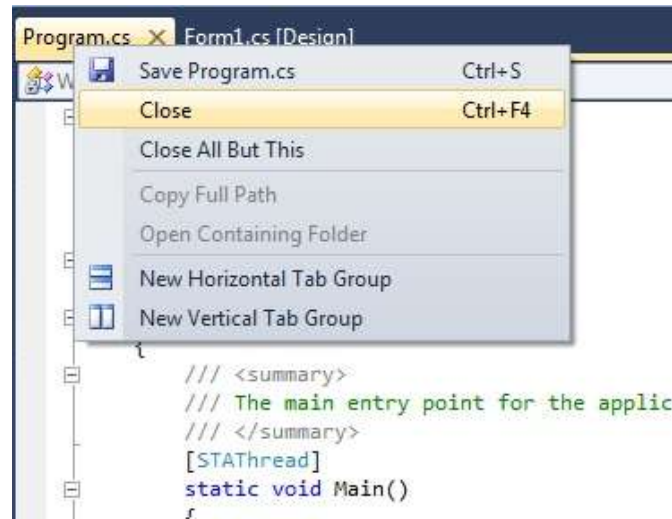
متد Main، نقطه شروع برنامه شماست. کدهایی که در میان براکت‌های متد Main وجود دارند، اولین خط‌هایی هستند که اجرا خواهند شد. آخرین خط از آن‌ها، همانطور که در عکس‌های بالا مشخص شده است، Form1 را برای شما اجرا می‌کند.

شما کارهای دیگری نیز در این متد Main می‌توانید انجام دهید. برای مثال، فرض کنید که شما یک برنامه‌ای دارید که در هنگام اجرا می‌خواهد اتصال به اینترنت را بررسی کند. اگر یک اتصال به اینترنت را بیابد، اطلاعاتی را برای شما از طریق بانک اطلاعاتی

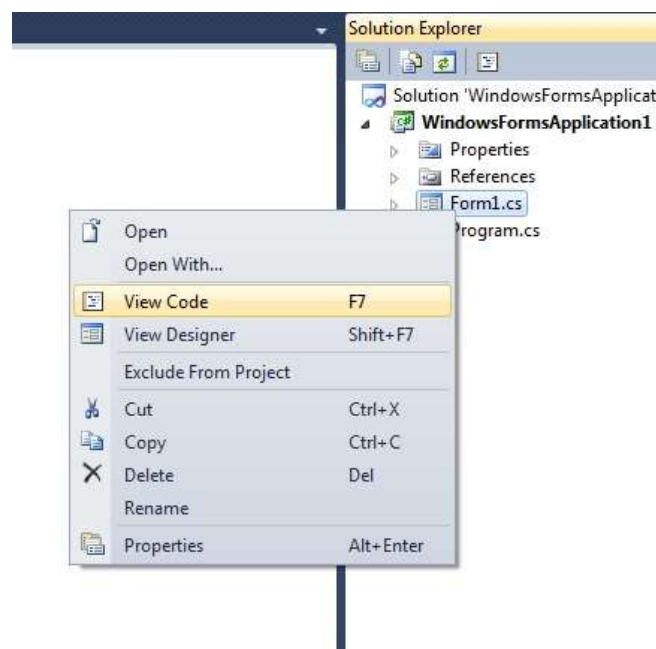
فراخوانی می‌کند و به نمایش می‌گذارد. این کار را دقیقاً می‌تواند در متد Main انجام داد. یعنی کدی بنویسید که آیا سیستم به اینترنت وصل است یا خیر؛ اگر پاسخ خیر باشد، پنجره‌ی دومی را نشان دهد. (مثلاً خطای عدم ارتباط با اینترنت)؛ اگر هم پاسخ بله است، Form را نمایش دهد. (همین اولین فرم منظور است).

اما نگران این کدها که شاید کمی شما را گیج کرده‌است، نباشید! چیزی که فعلاً باید حواستان به آن جمع باشد این است که متد Main در هنگام شروع برنامه، اجرا می‌شود و Program.cs فایل‌ی است که متد Main در آن ذخیره می‌شود.

اما فعلاً ما در Program.cs کدی را نخواهیم نوشت. از طریق سربرگ فوق، بر روی Program.cs راست کلیک کرده و بر روی Close کلیک کنید. Form مجدداً نمایش داده می‌شود.



اگر که می‌خواهید بدانید که ما بیشترین کدهایمان را در کجا خواهیم نوشت، با مراجعه مجدد به Solution Explorer، بر روی Form1.cs راست کلیک کنید.



این منو گزینه‌هایی با نام‌های View Designer و View Designer را دارا می‌باشد. همان فرمی است که هم‌اکنون می‌بینید. Code View، محلی است که کدهای ما آنجا ذخیره خواهند شد. بر روی آن کلیک کنید.

```

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

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {

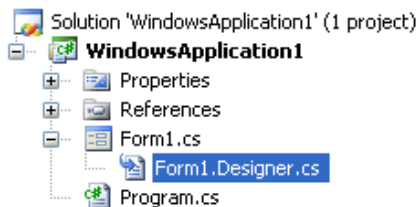
        }
    }
}

```

این کدهای خود Form می‌باشد و نباید به آن‌ها دست برد.

این‌بار، این کدها دارای تعداد بیشتری از کدهایی که با Using شروع شده است، می‌باشد. هیچ نگران این‌ها نباشید. استفاده از کلمه Using تنها به این معنی است که از کدهایی استفاده شود که قبلاً به‌صورت آماده نوشته و تهیه شده‌اند.

کدها همچنان یک بخشی با نام partial class Form1 دارد. آوردن نام Partial (=بخشی، قسمتی از) به این دلیل است که شما تنها بخشی از کدها را می‌بینید و قسمتی دیگر از دید شما پنهان هستند. برای دیدن قسمت‌های پنهان آن، دوباره به Solution Explorer مراجعه کنید و طبق تصویر، Form1.cs را باز کرده و بر روی Form1.Designer.cs دابل کلیک کنید.



```

namespace WindowsApplication1
{
    partial class Form1
    {
        /// <summary> ...
        private System.ComponentModel.IContainer components = null;

        /// <summary> ...
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        Windows Form Designer generated code
    }
}

```

مجدداً، partial class Form1 را خواهید دید که تمامی کدها در اینجا هستند. می‌توانید با کلیک کردن بر روی علامت + در کنار Designer generated code Windows Form، کدها را ببینید که همانند زیر هستند:

```
#region Windows Form Designer generated code

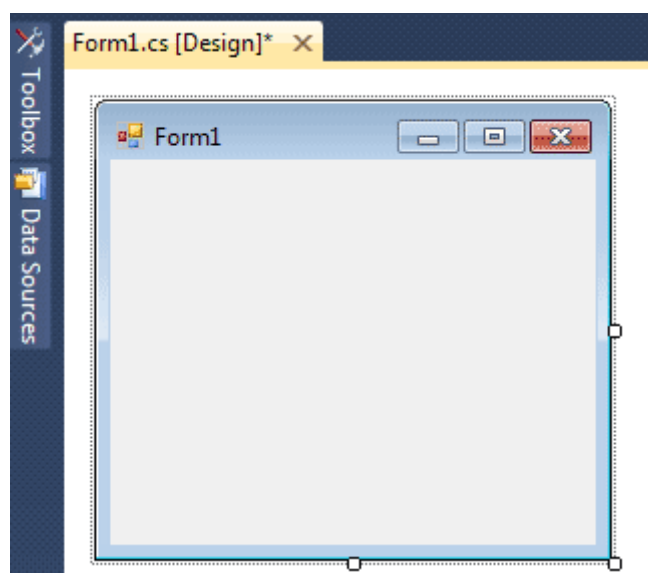
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.components = new System.ComponentModel.Container();
    this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
    this.Text = "Form1";
}

#endregion
```

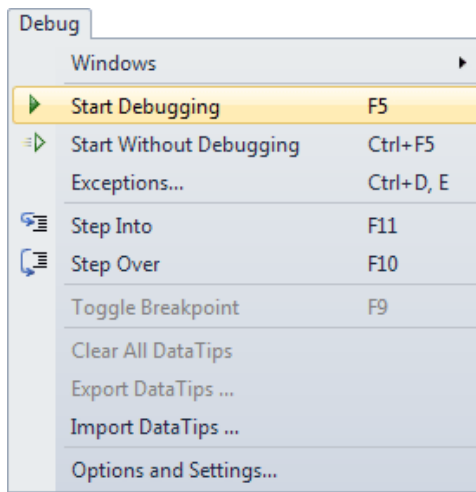
متد IntializeComponent متدی است که به‌صورت اتوماتیک هنگام ساختن یک پروژه Windows Form Application برای شما تولید می‌شود. هنگامی که ابزارهایی را همانند دکمه، جعبه متن، لیبل و ... را قرار می‌دهید، کدهای بیشتری به‌صورت اتوماتیک برای شما تولید می‌شوند.

اما خیالتان را راحت کنیم که فعلاً کاری با اینجا نداریم؛ پس با خیال راحت، پنجره مربوط به Form1.Designer.cs را ببندید.

Form1 را مجدداً باز کنید. اگر آن را بسته بودید، با مراجعه به Solution Explorer و با راست کلیک کردن بر روی آن، بر روی View Designer کلیک کنید. پنجره زیر نمایان می‌شود:



فعلاً ما بدون اضافه نمودن هیچ ابزاری، آن را اجرا می‌کنیم. برای اجرا، از منوی بالای برنامه Visual Studio، Debug را باز کنید و بر روی Start Debugging کلیک کنید.



هنگامی که بر روی Start Debugging کلیک می‌کنید، Studio Visual شروع به ساخت اولین برنامه شما می‌کند. اگر در این میان، مشکلی وجود داشته باشد، با یک پیغام خطایی شما را مطلع می‌سازد.

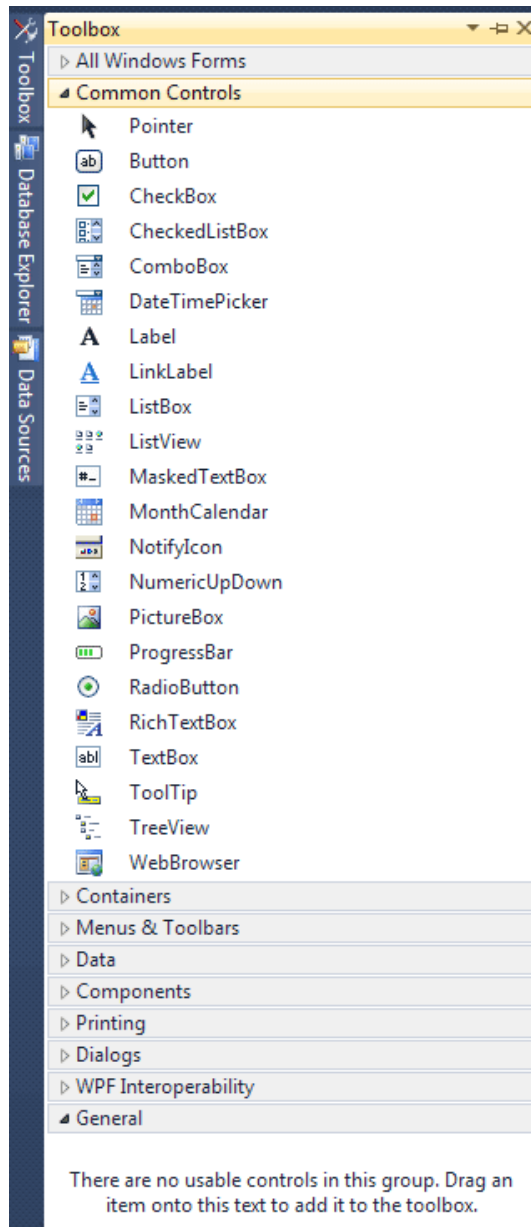
الان که برنامه‌ی شما اجرا شده است، می‌توانید خیلی راحت ببینید که شما نیز توانستید پنجره‌ای همانند دیگر پنجره‌های نرم‌افزارها بسازید ... دکمه قرمز برای خروج، دکمه‌های Maximize و Minimize نیز وجود دارند! حال برنامه را ببینید.

از این به بعد، هر زمانی که از لفظ «اجرای برنامه» استفاده کردیم، منظور آن خواهد بود که شما به منوی Debug مراجعه کنید و بر روی Start Debugging کلیک کنید.

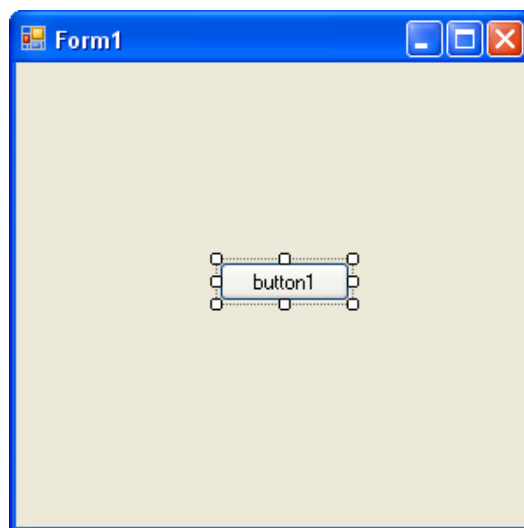
## • اضافه کردن کنترل‌ها به Form خالی خود

اولین کاری که در این بخش قرار است انجام دهیم، اضافه کردن کنترل‌ها به فرم خود است. در ادامه، پس از اضافه کردن آن‌ها، کدهای موردنظر خود را در آن‌ها نوشته و در نهایت با اجرا نمودن برنامه، نحوه اجرای آن را خواهیم دید.

برای اضافه نمودن یک کنترل (مانند Text Box، Button و ...) می‌توانید از Toolbox در کنار Form خود در سمت چپ استفاده کنید. Toolbox را باز کرده و پس از آن بر روی فلش Common Controls کلیک کنید تا موارد مربوط به آن باز شود. شما می‌بایست که لیستی از کنترل‌های مختلف همانند تصویر زیر ببینید.



بر روی Button یکبار کلیک کرده؛ این کار باعث می‌شود که Button انتخاب شود. برای اضافه کردن آن، هرکجای Form خود، یکبار کلیک کنید تا Button درج شود. اتفاقی همانند زیر برای فرم شما خواهد افتاد:

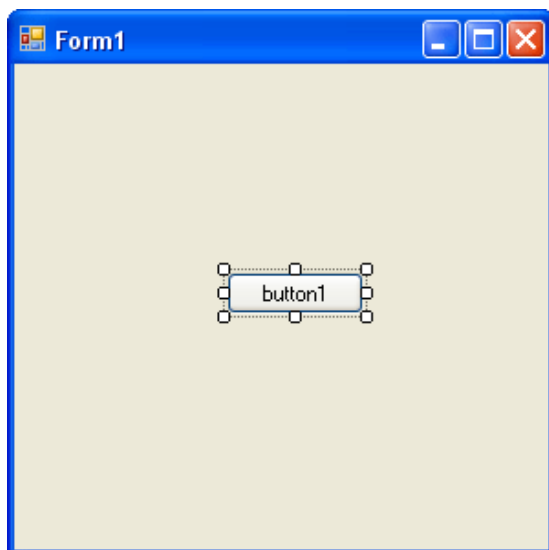


Button در واقع یک شیء‌ای است که توسط آن از دیگران می‌خواهید که بر روی آن کلیک کنند تا یک عمل خاصی انجام شود. وقتی که آن بر روی Button کلیک می‌کنند، کدهایی که در درون این Button نوشته‌اید اجرا خواهند شد. متنی که بر روی Button نوشته شده است (که به‌صورت پیش‌فرض با نام button1 معرفی شده) قابل تغییر می‌باشد.

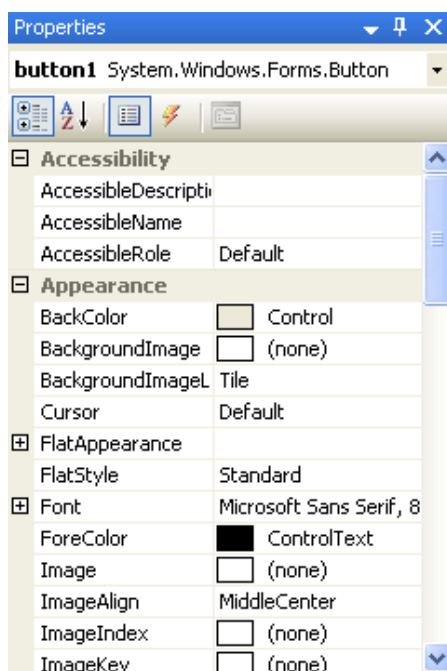


تمامی ابزارهایی که در Form خود استفاده می‌کنید (دقیقاً همانند همین Button-ای که قرار داده‌ایم)، همه دارای بخشی به نام Properties، به معنی خصوصیات هستند. توسط Properties این ابزارها، شما قادر به تغییر بسیار از خصوصیات ابزارها هستید و آن‌ها را به شکلی که تمایل دارید، شخصی‌سازی کنید. مثلاً طول و یا عرض آن را تغییر دهید، رنگ آن را عوض کنید و یا نام آن را به نام دیگری تبدیل کنید.

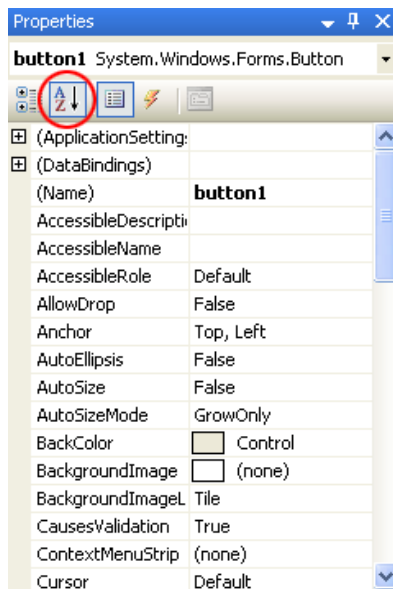
برای دستیابی به خصوصیات، ابتدا اطمینان حاصل فرمایید که کنترل شما انتخاب شده باشد. یعنی با یکبار کلیک کردن بر روی آن، ابزار انتخاب شود. به شکل زیر دقت کنید:



اکنون، سمت راست و پایین برنامه Visual Studio را مشاهده کنید؛ دقیقاً زیر Solution Explorer. شما می‌بایست که پنجره‌ای به نام Properties را مشاهده کنید. اگر این پنجره برای شما وجود ندارد (و یا حتی اگر شاید وجود دارد و آن را نمی‌بینید) با راست‌کلیک کردن بر روی کنترل موردنظر شما، آخرین گزینه، یعنی Properties را انتخاب کنید تا پنجره Properties برای شما به نمایش درآید.

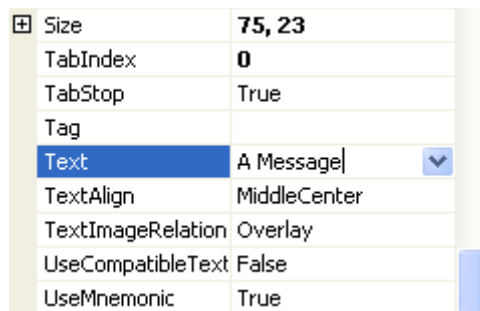


برای راحتی کار هنگام استفاده از Properties، پیشنهاد می‌شود که آن را بر اساس ترتیب حروف الفبای انگلیسی تنظیم کنید. برای این کار، طبق تصویر زیر، بر روی آیکون AZ کلیک کنید.

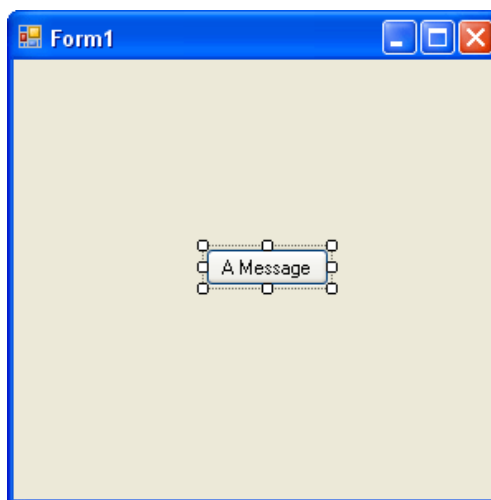


همانطور که می‌بینید، Property های بسیار زیادی را در اینجا خواهید دید. به سراغ حرف T بروید و Property ای به نام Text را پیدا کنید. این خصوصیت، متن کنترل را مشخص می‌کند. مثلاً برای همین Button، قبلاً دیدید نامی که کاربر شما مشاهده می‌کند، button1 می‌باشد. شما با استفاده از این خصوصیت، می‌توانید نام نمایشی را تغییر دهید تا کاربری که برنامه شما را می‌بیند، نامی مطابق با عملکرد ابزار را مشاهده کند.

بر روی مقدار مقابل Text کلیک کنید و آن را به نام مقابل تغییر دهید: **A Message**



بر روی Enter کلیک کنید تا نام جدید ثبت شود. حال بر روی Form خود نگاهی بیاندازید؛ خواهید دید که نام آن به نام A Message تغییر یافت.



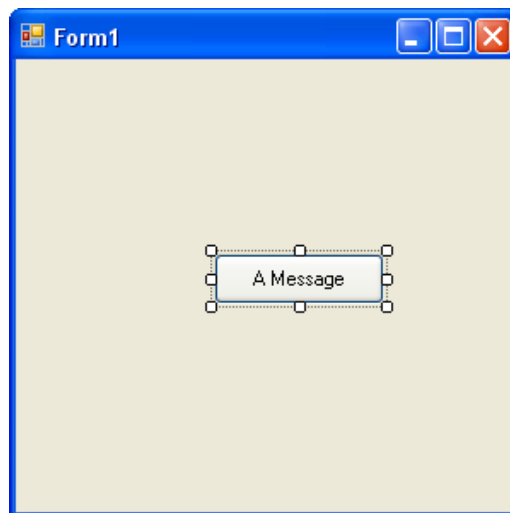
یک خصوصیت دیگری در پنجره‌ی Properties می‌توانید پیدا کنید که نام آن Size می‌باشد. توسط این خصوصیت، شما مقدار طول و عرض ابزار خود را می‌توانید تعیین کنید.

Padding	0, 0, 0, 0
RightToLeft	No
Size	75, 23
TabIndex	0
TabStop	True

اولین رقم از آن که ۷۳ می‌باشد، نمایانگر عرض ابزار شماست. دومین رقم که ۲۳ می‌باشد، نمایانگر طول ابزار شماست. آن‌ها را به ترتیب به ۱۰۰ و ۳۰ طبق تصویر زیر تغییر دهید.

Modifiers	Private
Padding	0, 0, 0, 0
RightToLeft	No
Size	100, 30
TabIndex	0
TabStop	True
Tag	
Text	A Message

Enter را مجدداً فشار دهید تا سایز جدید اعمال شود.



**نکته:** اگر که شما به جای ابزار خود، خود Form را انتخاب کنید و به بخش Properties آن مراجعه کنید، خواهید دید که خود Form نیز دارای خصوصیت‌های مختلفی می‌باشد. اگر مقدار Text را تغییر دهید، عنوان آن تغییر خواهد کرد.