



برنامه‌نویسی تحت وب

با C# و ASP.Net

مؤلف:

دکتر محمد علی ترکمانی

سرشناسه	: ترکمانی، محمدعلی، ۱۳۵۴ -
عنوان و نام پدیدآور	: برنامه نویسی تحت وب با C# و ASP.net [کتاب] / مولف محمدعلی ترکمانی.
مشخصات نشر	: مشهد: ارسطو، ۱۳۹۵.
مشخصات ظاهری	: ۲۰۶ ص.: مصور (رنگی).
شابک	: 978-600-432-035-1
وضعیت فهرست نویسی	: فیا
موضوع	: صفحه‌های سرور فعال
موضوع	: Active server pages
موضوع	: سی شارپ (زبان برنامه‌نویسی کامپیوتر)
موضوع	: C # (Computer program language)
موضوع	: مایکروسافت دات‌نت فریم‌ورک
موضوع	: Microsoft .NET Framework
موضوع	: وبگاه‌ها -- توسعه
موضوع	: Web site development
رده بندی کنگره	: ۱۳۹۵ ت۴ص/ ۸۸۸۵/ ۵/ ۱۰۵ TK۵
رده بندی دیویی	: ۰۰۵/۳۷۶
شماره کتابشناسی ملی	: ۴۳۴۴۴۵۸

نام کتاب: برنامه‌نویسی تحت وب با C# و ASP.NET

موضوع: برنامه نویسی C#

موضوع: ASP.NET

موضوع: جاوا اسکریپت

موضوع: CSS

موضوع: JQuery

موضوع: AJAX

موضوع: LINQ

مؤلف: دکتر محمد علی ترکمانی

ناشر: ارسطو (با همکاری سامانه اطلاع رسانی چاپ و نشر ایران)

صفحه آرای، تنظیم و طرح جلد: علی بیات

تیراژ: ۱۰۰۰ جلد

نوبت چاپ: دوم - ۱۳۹۸

چاپ: مدیران

تعداد صفحات: ۳۱۴ ص

قیمت: ۶۰۰۰۰ تومان

تلفن های مرکز پخش: ۳۵۰۹۶۱۴۵ - ۳۵۰۹۶۱۴۶ - ۰۵۱ - ۰۹۱۷۷۱۶۴۹۴۰

وب سایت: Torkamani/www.chaponashr.ir

این اثر مشمول قانون حمایت از مولفان و مصنفان و هنرمندان است. هر کس تمام یا قسمتی از این اثر را بدون اجازه مولف نشر یا پخش یا عرضه کند، مورد پیگرد قانونی قرار خواهد گرفت.

فهرست مطالب

فصل اول: معرفی محیط توسعه و ساختار برنامه های

۱۵..... ASP.NET

- ۱-۱-ویژوال استودیو NET..... ۱۵
- ۱-۲-ASP.NET چیست؟..... ۱۶
- ۱-۳-برخی از مدل های برنامه نویسی..... ۱۶
- ۱-۳-۱-مدل فرم های وب (web form model) ASP.Net..... ۱۶
- ۱-۳-۲-مدل کامپوننت ASP.Net..... ۱۷
- ۱-۳-۳-Net Framework Class Library..... ۱۸
- ۱-۳-۴-Windows Forms (فرم های ویندوز)..... ۱۸
- ۱-۳-۵-ASP.Net و ASP.Net AJAX..... ۱۸
- ۱-۳-۶-ADO.Net..... ۱۹
- ۱-۳-۷-LINQ..... ۱۹
- ۱-۴-مروری بر ساختار کلی دات نت و سرویس های آن..... ۱۹
- ۱-۵-آشنایی با نقش NET FRAMEWORK..... ۲۰
- ۱-۶-آشنایی با برخی سرویس های ارائه شده توسط CLR..... ۲۱
- ۱-۷-اجرای کدها در دات نت و سرویس JIT و اجرای مدیریت شده کدها..... ۲۴
- ۱-۸-مدیریت حافظه و سرویس GC..... ۲۵
- ۱-۹-ساخت اولین پروژه..... ۲۶
- ۱-۱۰-ساخت یک پروژه از نوع ASP.NET WEB APPLICATION..... ۳۱
- ۱-۱۱- THE VISUAL STUDIO IDE (محیط برنامه نویسی ویژوال استودیو)..... ۳۳

۳۵	۱-۱۲- کار با VIEWS و WINDOWS
۳۵	۱-۱۳- افزودن فولدر و فایل به وب سایت خود
۳۷	۱-۱۴- PROJECTS و SOLUTIONS
۳۷	۱-۱۵- انواع فایلها در ASP.NET
۳۸	۱-۱۶- ساخت و اجرای پروژه ((BUILDING & RUNNING
۳۸	۱-۱۷- اجرا و مشاهده صفحه ایجاد شده در BROWSER
۳۹	۱-۱۸- دستورات USING
۳۹	۱-۱۹- کلاس با پیشوند PARTIAL
۳۹	۱-۲۰- کالبدشکافی ساختار یک صفحه وب ASP
۴۰	۱-۲۱- PAGE DIRECTIVE
۴۱	۱-۲۲- CODE SECTION
۴۱	۱-۲۳- PAGE LAYOUT (طرح کلی صفحه)
۴۳	۱-۲۴- استفاده از محیط برنامه نویسی VISUAL STUDIO

فصل دوم: HTML ۴۵

۴۵	۲-۱- مقدمه
۴۶	۲-۲- قالب کلی فایل های HTML
۵۵	۲-۳- ایجاد فاصله از مارجین یا حاشیه صفحه
۵۶	۲-۴- ساخت لینک (تگ A)
۵۷	۲-۵- پرش از یک بخش به بخش دیگر در همان صفحه:
۵۸	۲-۶- نمایش TOOLTIP با ویژگی TITLE
۵۹	۲-۷- توضیحات
۵۹	۲-۸- تگ <HR/>

- ۶۰ ۲-۹-تگهای SUP و SUB
- ۶۰ ۲-۱۰-رسم جدول برای قالب بندی محتویات صفحه
- ۶۲ ۲-۱۱-تگ OBJECT

۶۳ فصل سوم: CSS

- ۶۳ ۳-۱-مقدمه
- ۶۴ ۳-۲-استفاده از سبک روی HEADING
- ۶۵ ۳-۳-ساخت یک پروژه در ویژوال استودیو و اضافه کردن یک تصویر به وب
- ۷۰ ۳-۴-ویژگی‌های مربوط به تصویر
- ۷۲ ۳-۵-تراز بندی متن در صفحه با CSS
- ۷۲ ۳-۶-ایجاد پس زمینه با استفاده از CSS
- ۷۴ ۳-۷-نمایش یک تصویر به عنوان پس زمینه در تگ BODY
- ۸۰ ۳-۸-تنظیم MARGIN و PADDING
- ۸۳ ۳-۹-رسم BORDER با رنگ
- ۸۳ ۳-۱۰-STYLE دادن به لینک‌ها
- ۸۶ ۳-۱۱-قالب بندی جدول
- ۸۶ ۳-۱۲-تگ DIV
- ۸۹ ۳-۱۳-لایه بندی به کمک DIV
- ۸۹ ۳-۱۴-ساخت جدول با DIV
- ۹۳ ۳-۱۵-SPAN
- ۹۳ ۳-۱۶-تعریف کلاس در CSS
- ۹۵ ۳-۱۷-ID در سبک نویسی
- ۹۷ ۳-۱۸-انواع سبک‌های CSS

۹۸ ۳-۱۹- تثبیت موقعیت لایه ها در CSS

۱۰۳ ۳-۲۰- کار با Z-INDEX

۱۰۴ ۳-۲۱- حالت‌های گروهی

فصل چهارم: جاوا اسکریپت ۱۰۵

۱۰۵ ۴-۱- تگ SCRIPT

۱۰۶ ۴-۲- بررسی کنترل‌های HTML

۱۱۰ ۴-۳- تعریف تابع اسکریپت در فایل خارجی (یعنی در دل خود پروژه)

۱۱۲ ۴-۴- تصاویر متداول مورد استفاده در صفحات وب

۱۱۲ ۴-۵- تگ NOSCRIPT

۱۱۳ ۴-۶- POST BACK چیست؟

۱۱۴ ۴-۷- رد کردن پارامتر به صفحه

۱۱۷ ۴-۸- مدیریت کنترل‌های پویا در سمت CLIENT با استفاده از جاوا اسکریپت:

۱۱۸ ۴-۹- رویداد ONTEXTCHANGE مربوط به TEXTBOX

۱۲۴ ۴-۱۰- دستوره‌های جاوا اسکریپت

۱۲۴ ۴-۱۰-۱- کامنت

۱۲۵ ۴-۱۰-۲- تعریف متغیرها

۱۲۵ ۴-۱۰-۳- چاپ محتویات متغیر در MessageBox

۱۲۵ ۴-۱۰-۴- نامگذاری متغیرها

۱۲۶ ۴-۱۰-۵- عملگرهای Object و typeof

۱۲۹ ۴-۱۰-۶- تبدیل به عدد: parseInt و parseFloat

۱۳۱ ۴-۱۰-۷- تابع Boolean

۱۳۱ ۴-۱۰-۸- تابع number

۱۳۲ String تابع ۴-۱۰-۹

فصل پنجم: برنامه نویسی سمت سرویس دهنده ۱۳۵

۱۳۵ ۵-۱-مقدمه

۱۳۵ ۵-۲-ساخت یک پروژه

۱۳۷ ۵-۳-کنترل HYPERLINK

۱۳۸ ۵-۴-اضافه کردن HYPER LINK به پروژه

۱۳۸ ۵-۵-کنترل های BUTTON

۱۴۰ ۵-۶-کنترل LABEL و TEXTBOX

۱۴۲ ۵-۷-ساخت یک فرم ساده ورود اطلاعات

۱۴۹ ۵-۸-آشنایی با POST BACK

۱۵۱ ۵-۹-ویژگی AUTO POST BACK

۱۵۲ ۵-۱۰-کنترل IMAGE

۱۵۲ ۵-۱۱-اضافه کردن عکس به پروژه

۱۵۶ ۵-۱۲-آشنایی با کنترل کاربر (USER CONTROL)

۱۶۰ ۵-۱۳-آشنایی با فایل WEB.CONFIG

۱۶۲ ۵-۱۴-چرخه مدیریت رویدادها

۱۶۳ ۵-۱۵-آرگومان های EVENT

۱۶۳ ۵-۱۶-رخدادهای مربوط به APPLICATION - رخدادهای SESSION

۱۶۴ ۵-۱۷-رخدادهای مربوط به PAGE و CONTROL

۱۶۵ ۵-۱۸-رویداد PRELINIT صفحه

۱۶۵ ۵-۱۹-مدیریت رخدادها با استفاده از CONTROL ها

۱۶۶ ۵-۲۰-رخدادهای پیش فرض (DEFAULT EVENTS)

۲۱-۵- رویداد COMMAND وخواص COMMAND NAME و COMMAND ARGUMENTS مربوط به دکمه	۱۶۸
۲۲-۵- نکته مهم در مورد کار با رویدادها	۱۷۱
۲۳-۵- کنترل های رادیویی (RADIO BUTTON)	۱۷۴
۲۴-۵- کنترل RADIO BUTTON LIST	۱۷۶
۲۵-۵- کنترل CHECKBOX	۱۸۰
۲۶-۵- کنترل CHECKBOXLIST	۱۸۰
۲۷-۵- کنترل TABLE	۱۸۲
۲۸-۵- کنترل DROPDOWNLIST	۱۸۳
۲۹-۵- درک تفاوت برنامه های ویندوز و برنامه های وب .	۱۸۶
۳۰-۵- کنترل LIST BOX	۱۸۹
۳۱-۵- کنترل BULLETEDLIST	۱۹۱
۳۲-۵- MASTER PAGE	۱۹۲
۳۳-۵- کنترل HYPER LINK	۱۹۶
۳۴-۵- آپلود / بارگذاری فایل در سرویس دهنده با استفاده از تکنولوژی ASP.NET	۱۹۶

فصل ششم: دسترسی به پایگاه داده با ASP.NET..... ۲۰۱

۶-۱- مقدمه	۲۰۱
۶-۲- بازبایی و نمایش داده	۲۰۱
۶-۳- ADO.NET و کاربرد آن در ASP	۲۰۷
۶-۳-۱- کلاس DataSet	۲۰۸
۶-۳-۲- کلاس DataTable	۲۱۲
۶-۳-۳- کلاس DataRow	۲۱۳

۲۱۴ DataAdapter شی ۶-۳-۴

۲۱۴ DataReader شی ۶-۳-۵

۲۱۵ DbConnection و DbCommand های شی ۶-۳-۶

۲۱۹ فصل هفتم: مدیریت حالت

۲۱۹ ۷-۱-مقدمه

۲۲۰ ۷-۲-مدیریت حالت

۲۲۰ ۷-۳-مدیریت وضعیت در ASP.NET

۲۲۱ ۷-۴-روشهای مدیریت حالت در ASP.NET

۲۲۱ ۷-۴-۱-استفاده از کنترل hidden field

۲۲۱ ۷-۴-۲-استفاده از viewstate

۲۲۲ ۷-۴-۳-کوکی

۲۲۵ ۷-۴-۴-استفاده از session

۲۲۶ ۷-۴-۵-حافظه Application

۲۲۸ ۷-۴-۶-session ID

۲۳۱ فصل هشتم: اعتبارسنجی در ASP.NET

۲۳۱ ۸-۱-کنترل های ASP.NET-VALIDATION

۲۳۱ ۸-۲-کلاس BASEVALIDATOR

۲۳۲ ۸-۳-کنترل REQUIREDFIELDVALIDATOR

۲۳۲ ۸-۴-کنترل RANGEVALIDATOR

۲۳۳ ۸-۵-کنترل COMPAREVALIDATOR

۲۳۴ ۸-۶-REGULAREXPRESSIONVALIDATOR

۲۳۶..... CUSTOMVALIDATOR-۸-۷

۲۳۶..... VALIDATIONSUMMARY کنترل-۸-۸

فصل نهم: AJAX و JQUERY ۲۴۳

۲۴۳ ۹-۱- JQUERY چیست؟

۲۴۳ ۹-۲- افزودن JQUERY به صفحات وب خود

۲۴۴ ۹-۳- روش اول دانلود JQUERY

۲۴۵ ۹-۴- روش جایگزین استفاده از یک CDN

۲۴۵ ۹-۵- GOOGLE CDN

۲۴۵ ۹-۶- MICROSOFT CDN

۲۴۶ ۹-۷- یکی از مزایای بزرگ استفاده از CDN های GOOGLE یا MICROSOFT

۲۴۶ ۹-۸- ساختار نگارشی (JQUERY SYNTAX)

۲۴۷ ۹-۹- رخدادهای READY

۲۴۸ ۹-۱۰- JQUERY SELECTOR انتخاب گره های JQUERY

۲۴۸ ۹-۱۱- انتخاب المان بر مبنای نام آن (ELEMENT SELECTOR)

۲۴۸ ۹-۱۲- انتخاب گر ID#

۲۴۹ ۹-۱۳- انتخاب بر اساس اسم کلاس (.CLASS SELECTOR)

۲۵۰ ۹-۱۴- جایگذاری توابع مورد نیاز داخل یک فایل جدا

۲۵۱ ۹-۱۵- EVENT ها در JQUERY

۲۵۱ ۹-۱۶- اساسا رخداد (EVENT) به چه گفته می شود؟

۲۵۲ ۹-۱۷- چگونگی استفاده از رخدادها در JQUERY

۲۵۲ ۹-۱۸- توابع رخداد پر کاربرد در JQUERY

۲۵۲ ۹-۱۸-۱- تابع \$(document).ready()

۲۵۲	click	تابع	۹-۱۸-۲
۲۵۳	dblclick	تابع	۹-۱۸-۳
۲۵۳	mouseenter	تابع	۹-۱۸-۴
۲۵۳	mouseleave	تابع	۹-۱۸-۵
۲۵۴	mousedown	تابع	۹-۱۸-۶
۲۵۴	mouseup	تابع	۹-۱۸-۷
۲۵۴	hover	تابع	۹-۱۸-۸
۲۵۵	focus	تابع	۹-۱۸-۹
۲۵۵	blur	تابع	۹-۱۸-۱۰
۲۵۵	on	تابع	۹-۱۸-۱۱
۲۵۶	AJAX چیست ؟		۹-۱۹
۲۵۷	چرا JQuery و AJAX را با هم بکار می‌بریم ؟		۹-۱۹-۱
۲۵۷	متد load() در Ajax - jquery load()		۹-۱۹-۲
۲۵۹	توابع get() و post()		۹-۱۹-۳
۲۵۹	درخواست داده از سرویس دهنده get: در مقایسه با post		۹-۱۹-۴
۲۶۰	متد \$.get()		۹-۱۹-۵
۲۶۱	متد \$.post()		۹-۱۹-۶

فصل دهم: کنترل‌های AJAX در ASP.NET ۲۶۳

۲۶۳	مقدمه		۱۰-۱
۲۶۴	کنترل SCRIPTMANAGER		۱۰-۲
۲۶۵	کنترل UPDATEPANEL		۱۰-۳
۲۶۸	خاصیت‌های کنترل UPDATEPANEL		۱۰-۴

۲۶۹ UPDATEPANEL کنترل توابع	۱۰-۵
۲۷۱ UPDATEPROGRESS کنترل	۱۰-۶
۲۷۲ UPDATEPROGRESS کنترل های خاصیت	۱۰-۷
۲۷۳ UPDATEPROGRESS: توابع کنترل	۱۰-۸
۲۷۳ TIMER کنترل	۱۰-۹

فصل یازدهم: WEB SERVICE ۲۷۵

۲۷۵ WEB SERVICE چیست ؟	۱۱-۱
۲۷۵ WEB SERVICE ایجاد	۱۱-۲
۲۸۰ WEB SERVICE استفاده از	۱۱-۳
۲۸۲ PROXY ایجاد	۱۱-۴

فصل دوازدهم: امنیت در ASP.NET ۲۸۷

۲۸۷ مقدمه	۱۲-۱
۲۸۹ پیاده سازی امنیت و احراز هویت از طریق فرم	۱۲-۲
۲۹۷ IIS AUTHENTICATION (اعتبار سنجی و سرویس های اطلاعات اینترنتی): SSL	۱۲-۳

فصل سیزدهم: استفاده از LINQ در ASP.NET ۲۹۹

۲۹۹ مقدمه	۱۳-۱
۳۰۳ عملگرهای LINQ	۱۳-۲
۳۰۴ Join clause (دستور join)	۱۳-۲-۱
۳۰۵ دستور Where	۱۳-۲-۲
۳۰۶ دستورهای Orderby و Orderbydescending	۱۳-۲-۳
۳۰۷ دستور Let	۱۳-۲-۴

فصل چهاردهم: PUBLISH کردن وب سایت و تنظیمات

۳۰۹IIS

۳۰۹۱۴-۱ PUBLISH کردن وب سایت.....

۳۱۰۱۴-۲ نصب IIS در ویندوز ۸,۱.....

۳۱۱۱۴-۳ تنظیمات IIS در ویندوز 8.1.....

۳۱۴.....منابع:

مقدمه:

امروزه فناوری اطلاعات در تمامی ابعاد زندگی بشر نفوذ کرده و زندگی بشر را به شدت تحت تاثیر قرار داده است. برهمگان واضح است که فناوری اطلاعات با استفاده از سرویس‌های مختلفی که ارائه می‌دهد، ارزش افزوده زیادی ایجاد می‌کند. با توجه به اهمیت این موضوع، درس خدمات ارزش افزوده فناوری اطلاعات در برنامه درسی رشته‌های فناوری اطلاعات و ICT قرار گرفته است. گرچه هدف اصلی این درس مطالعه پیرامون فناوری‌های مرتبط با IT و ارزش افزوده آنها است، اما یک هدف مهم تر نیز آن است که دانشجو بتواند در خصوص فناوری‌های جدیدی که در حوزه IT ارائه می‌شود، تحقیق نماید و خدمات ارزش افزوده قابل ارائه در این حوزه را شناسایی نماید. علیرغم اهمیت این درس منبع فارسی مناسبی برای آن وجود ندارد. این کتاب مطابق با سرفصل وزارت علوم تحقیقات و فناوری برای این درس تدوین شده است. در پایان هر بخش سئوالات تشریحی و چهار گزینه‌ای ارائه شده است تا دانشجویان بهتر بتوانند خود را برای آزمون‌های پیش رو آماده نمایند. امید است این اثر مورد توجه همکاران و دانشجویان گرامی قرار گرفته و به افزایش کیفیت آموزشی این درس کمک نماید. از اساتید و دانشجویان گرامی تقاضا دارم دیدگاه‌های خود را از طریق ایمیل m.a.torkamani@gmail.com و با مولف در میان بگذارند تا انشاءالله در ویرایش‌های بعدی اشکالات یا کاستی‌های احتمالی کتاب مورد تجدید نظر قرار گیرد. در پایان وظیفه خود می‌دانیم از زحمات آقای مهندس بیات به خاطر طراحی جلد کتاب و همچنین از مدیریت انتشارات ارسطو و سامانه اطلاع‌رسانی چاپ و نشر ایران، جناب آقای حسین قنبری، تشکر و قدردانی نمایم.

محمد علی ترکمانی

پاییز ۹۵

فصل اول

معرفی محیط توسعه و ساختار برنامه های

ASP.NET

۱-۱- ویژوال استودیو Net.

برای نوشتن ، اشکال زدایی یا خطایابی و اجرای یک برنامه، مدیریت پروژه‌ها و فایل‌ها از یک محیط توسعه Integrated Development Environment یا به اختصار IDE استفاده می‌شود. یکی از معروفترین و بهترین ابزارها برای نوشتن برنامه های تحت وب، Visual Studio هست که در نسخه های مختلفی ارائه شده است. با نصب Visual Studio، Visual NetFramework. و زبان سی شارپ بر روی سیستم شما نصب خواهند شد. Visual Studio در نسخه های مختلفی ارائه شده است. ما از نسخه Enterprise استفاده میکنیم. ویژوال استودیو Net. یک محیط توسعه نرم افزار است که با استفاده از فناوری ها و زبانهای برنامه نویسی زیر می‌توان یک برنامه تحت وب را توسعه داد:

- HTML
- ASP.Net
- C#
- Jscript
- JQuery
- Ajax
- ADO.NET
- Linq
- .Net Framework
- MVC

۲-۱- ASP.Net چیست؟

ASP.Net بخشی از سکوی نرم افزار (Microsoft.Net platform) می باشد که یک مدل برنامه نویسی ، یک زیرساخت نرم افزار جامع و سرویس های متنوعی برای ساخت نرم افزار های تحت وب کارآمد و قدرتمند برای PC همانند موبایل ها ارائه می دهد.

ASP.Net علاوه بر پروتکل HTTP از دستورات و سیاست های HTTP برای تنظیم یک ارتباط و همکاری دوطرفه از مرورگر به سرور استفاده می کند. اپلیکیشن های تحت ASP.Net کدهای ترجمه شده ای هستند که از مولفه های (component) توسعه پذیر (extensible) و قابل استفاده ی مجدد موجود در .Net Framework استفاده می کند. این کدها می توانند از کلیه ی سلسله مراتب (hierarchy) کلاس ها در .Net Framework استفاده کنند.

ASP.Net برای تولید اپلیکیشن های تحت وب تعاملی (interactive) و مبتنی بر داده (داده محور یا data-driven) از طریق اینترنت استفاده می شود، که شامل تعداد زیادی ابزار کنترلی (control) از قبیل textboxها، دکمه ها و lable هایی برای پیکربندی و دستکاری کد برای ایجاد صفحات HTML می باشد.

ASP.NET یک فناوری برای ایجاد صفحات وب پویا است. صفحه وب پویا صفحه ای است که شکل ظاهری آن ثابت اما محتوای آن باتوجه درخواست کاربر تغییر می کند.

ASP.NET یک مدل برنامه نویسی کاملاً شی گرا است.

ASP.NET اجازه می دهد که از زبانهایی که از .NET پشتیبانی می کنند ، نظیر C# و

VB.NET و غیره برنامه نویسی کنید.

ASP.NET چند مرورگری است و به راحتی می توان کدهای نوشته شده به این زبان را توسط هر مرورگری اجرا نمود.

ASP.NET یک زبان برنامه نویسی سمت سرویس دهنده است. یعنی برنامه های نوشته شده در

سمت سرویس دهنده اجرا می شوند.

۳-۱- برخی از مدل های برنامه نویسی

۱-۳-۱- مدل فرم های وب (ASP.Net (web form model)

فرم‌های وب ASP.Net قابلیت رویداد محور بودن (event-driven) را به اپلیکیشن‌های تحت وب، اضافه می‌کند. مرورگر، فرمی از وب را به سرور وب ارسال می‌کند و سرور در پاسخ یک صفحه‌ی نشانه‌گذاری شده (markup page) یا صفحه HTML را برمی‌گرداند. تمام فعالیت‌های سمت سرور، سرور، خروجی (output) عملیات کاربر را پردازش می‌کند و عکس‌العمل فرستاده می‌شوند. سرور، خروجی (output) عملیات کاربر را پردازش می‌کند و عکس‌العمل‌هایی نشان می‌دهد. اکنون HTTP یک پروتکل stateless (ناپایدار) می‌باشد. چارچوب کاری ASP.Net (framework) به ذخیره‌سازی اطلاعات مربوط به وضعیت (state) برنامه‌ی تحت وب (web application) کمک می‌کند که شامل موارد زیر می‌باشد.

- Page state

- Session state

وضعیت صفحه و وضعیت مربوط به کاربر می‌باشد، برای مثال محتویات فیلدهای ورودی (input field) مختلف در فرم وب. Session state نیز مجموعه‌ای است حاصل از صفحات مختلفی که کاربر از آن‌ها را بازدید و با آن‌ها کار کرده است، به عنوان مثال می‌توان به کل session state اشاره کرد. برای واضح کردن مفهوم اجازه دهید در ادامه مثالی از سبد خرید ارائه دهیم. کاربر آیتم‌هایی را به سبد خرید اضافه می‌کند. آیتم‌ها از صفحه‌ای انتخاب می‌شوند که صفحه آیتم‌ها نامیده می‌شود، کل آیتم‌های جمع‌آوری شده و قیمت‌ها در صفحه‌ای متفاوت نشان داده می‌شوند که صفحه سبد خرید نامیده می‌شود. فقط HTTP نمی‌تواند مسیر تمام اطلاعات ورودی از تمام صفحات مختلف را حفظ کند. Session state (وضعیت session) و زیرساخت سمت سرور (ASP.Net server side infrastructure) تمام اطلاعاتی که به صورت سراسری در طی یک session جمع‌آوری شده را دارد.

ASP.NET runtime، page state (وضعیت صفحه) را از سمت / و به سمت سرور در سراسر صفحه تقاضاها حمل می‌کند و در زمان اجرا، کدهای asp.net را تولید می‌کند و کامپوننت‌های مربوط به بخش سرور را در فیلدهای پنهان ترکیب (می‌گنجانند) می‌کند. از این طریق سرور از وضعیت کلی برنامه‌ی کاربردی آگاه شده و به یک روش اتصال دو لایه (two-tiered connected way) عمل می‌کند.

۲-۳-۱- مدل کامپوننت ASP.Net

این مدل از ASP.Net ساختارهای متفاوتی از صفحات ASP.Net را ارائه می دهد. اساساً این مدل، مدلی است که موارد زیر را توصیف می کند:

مولفه های بخش سرور شامل همه ی عناصر HTML یا تگ هایی مانند <form> و <input>

کنترل های سرور که به توسعه رابط پیچیده ی کاربر کمک می کند، برای مثال کنترل calendar یا کنترل Gridview.

ASP.Net فن آوری است که تحت Net Framework کار می کند که شامل همه ویژگی های مربوط به وب می شود. Net Framework. از سلسه مراتب شی گزایی ساخته شده است. نرم افزار وب ASP.Net از صفحات ساخته شده است. وقتی که یک کاربر تقاضای صفحه ی ASP.Net دارد، IIS فرایند صفحه را به سیستم ASP.Net runtime محول می کند.

۳-۳-۱- Net Framework Class Library

شامل کتابخانه ای غنی و وسیع از typeها است که امکان و قابلیت بارها استفاده از آن ها (library of reusable types) وجود دارد. کلاس ها، ساختارها، رابط ها (interfaces) و مقادیر شمارشی (enumerated values) که جمعاً نوع (type) نامیده می شوند.

۳-۳-۴- Windows Forms (فرم های ویندوز)

فرم های ویندوز دربردارنده ی نمایش گرافیکی (graphical representation) کلیه ی پنجره (window) هایی است که در برنامه ی کاربردی (app) نشان داده می شوند.

۳-۳-۵- ASP.Net AJAX و ASP.Net

ASP.Net مدل توسعه و برنامه نویسی تحت وب (web development model) و AJAX شاخه ی از (یا افزونه ای برای) ASP.Net برای توسعه و اجرای قابلیت ها و امکانات AJAX می باشد. ASP.Net AJAX شامل کامپوننت هایی است که به برنامه نویس تحت وب اجازه می دهد که داده را در یک وب سایت بدون نیاز به بارگذاری مجدد در صفحه آپدیت و بروز رسانی کند.

۶-۳-۱- ADO.Net

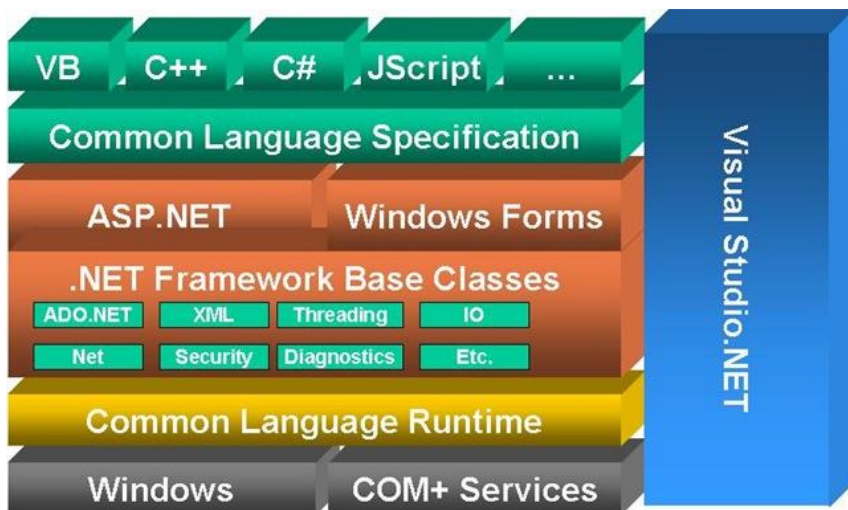
فن آوری می باشد که برای کار با داده و پایگاه داده استفاده می شود. ADO.Net دسترسی به منابع دیتا مانند سرور SQL, XML, OLE DB و غیره را فراهم می کند. این فن آوری امکان اتصال با منابع دیتا را برای بازیابی، دستکاری و آپدیت کردن دیتا فراهم می کند.

۷-۳-۱- LINQ

قابلیت های data querying (پرسمان از داده) را با استفاده از سینتکسی (دستور نحوی) که شبیه به سینتکس SQL query language (زبان پرسمان از داده ی SQL) قدیمی است به زبان های (تحت).Net انتقال می دهد (اضافه می کند).

۴-۱- مروری بر ساختار کلی دات نت و سرویس های آن

NET Framework از دو بخش اصلی تشکیل شده، کتابخانه دات نت یا NET. Framework Class Library و Common Language Runtime یا CLR. کتابخانه دات نت مجموعه ای از کلاس های آماده از پیش نوشته شده است که در تمامی زبان های مبتنی بر NET قابل استفاده هستند. اما CLR وظیفه اجرا و مدیریت برنامه های تحت دات نت را دارد. تبدیل کدهای IL به کد ماشین، کنترل و مدیریت اجرای برنامه ها، مدیریت حافظه و کنترل و مدیریت امنیت در برنامه های دات نت از وظایف CLR هست. در حقیقت می توان گفت CLR قلب تپنده دات نت هست. برای درک بهتر ساختار دات نت به شکل ۱-۱ توجه کنید.



شکل ۱-۱:

در بالای شکل ۱-۱ زبان هایی قرار دارند که شما می توانید بوسیله آنها برنامه های خود را مبتنی بر دات بنویسید. در مرکز این شکل کتابخانه دات نت قرار دارد. این کتابخانه مجموعه ای از کدهای از پیش نوشته شده است که نوشتن کدها را برای شما راحت تر می کند. بعد از کتابخانه، CLR قرار دارد که وظیفه تبدیل کدهای شما به زبان ماشین و اجرای برنامه ها رو به عهده دارد. در پایین ترین لایه، سرویس های ویندوز قرار دارد که وظیفه ارتباط با این لایه، به عهده CLR و کتابخانه دات نت هست .

۵-۱- آشنایی با نقش .NET Framework

کتابخانه دات نت مجموعه ای از کدهای Reusable (کدهایی که یکبار نوشته شده و قابلیت استفاده مجدد را دارند) می باشند که در اختیار برنامه نویسان دات نت قرار گرفته اند. از این کدها می تواند در کلیه برنامه های دات نت، از برنامه های Console گرفته تا برنامه های ویندوز، وب، ارتباط با بانک های اطلاعاتی و سرویس های وب و ویندوز استفاده کرد . اما یکی از اصطلاحات در دات نت CLS یا Common Language Specification است .

موسسه European Computer Manufacturers Association یا ECMA ، استاندارد را تدوین کرده که CLS مبتنی بر این استاندارد پیاده سازی شده است. در حقیقت CLS استاندارد است که با رعایت این استاندارد، کلیه زبان های برنامه نویسی مبتنی بر دات نت، قابلیت ارتباط با یکدیگر را خواهند داشت. برای مثال زبان های VB.NET و C# دو زبان مبتنی بر دات نت هستند. زمانی که شما کدی را با زبان C# نوشته اید، با CTS این قابلیت به شما داده میشود که بتوانید از کد نوشته شده در زبان C# در زبان VB.NET نیز استفاده کنید. کدهایی که با استاندارد CLS تطابق داشته باشند، بعد از کامپایل به یک کد میانی به نام IL ترجمه می شوند، کد IL شبیه به کد Assembly هست. بعد از ایجاد کد IL توسط کامپایلر، کد IL توسط CLR به کد ماشین ترجمه شده و اجرا می شود. در حقیقت بعد از کامپایل کردن دو برنامه که یکی به زبان C# و دیگری به زبان VB.NET نوشته شده، کدی که ایجاد می شود کد IL هست .

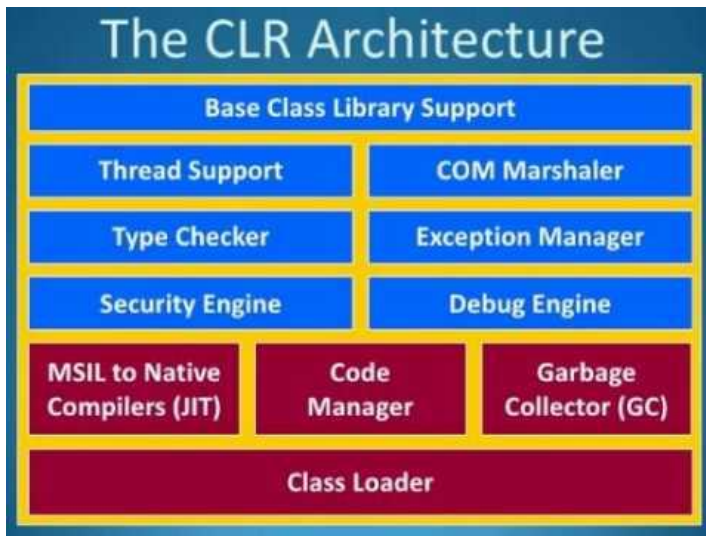
کدی میانی که توسط کامپایلرهای مبتنی بر دات نت ایجاد می شوند Microsoft Intermediate Language یا MSIL نام دارند که به برنامه ها قابلیت برقراری ارتباط با یکدیگر را می دهد . استاندارد دیگری که توسط ECMA تدوین شده، استاندارد CLI یا Common Language Infrastructure هست. این استاندارد یک سری خصوصیات مشترک در بین کدهای میانی ایجاد شده توسط کامپایلرها رو تعریف میکند که برای اجرای کد IL به آنها نیاز است. برای مثال سیستم نوع داده مشترک (Common Type System) یا سرویس های از قبیل Type Safety و اجرای مدیریت شده کدها از خصوصیات تدوین شده هستند. لیست زیر خلاصه ای از استاندارد های ECMA تحت دات نت می باشد :

- CLS یا Common Language Specification
- CTS یا Common Type System
- TS یا Type Safety
- IL یا Intermediate Language

۶-۱- آشنایی با برخی سرویس های ارائه شده توسط

CLR

همانطور که بخش قبلی گفتیم، CLR قلب تپنده دات نت بوده و اکثر وظایف زمان اجرای برنامه های دات نت به عهده CLR می باشد. در حقیقت، برای اجرای برنامه های مبتنی بر دات نت، حتماً باید نسخه مربوطه NET Framework. بر روی سیستم مقصد نصب شده باشد. در این بخش قصد داریم تا برخی از سرویس های CLR را معرفی کرده و در مورد دو سرویس، یعنی JIT و GC به تفصیل صحبت کنیم. در شکل ۱-۲ نمای کلی از سرویس های CLR را مشاهده می کنید.



شکل ۱-۲:

اجزای ذکر شده در این شکل به شرح ذیل هستند:

۱- Class Loader: این سرویس وظیفه بارگذاری کلاس ها در زمان اجرا در محیط CLR را به عهده دارد

۲- IL to Native: در قسمت قبل گفتیم که کدهای نوشته شده در زبان های مبتنی بر دات نت، بعد از کامپایل به یک زبان میانی به نام IL تبدیل می شوند، این سرویس وظیفه تبدیل کدهای IL به کد ماشین جهت اجرای برنامه را به عهده دارد. Code Manager این سرویس وظیفه مدیریت کدها در زمان اجرا را به عهده دارد.

۳- Garbage Collector : با کمک این سرویس، CLR کار مدیریت حافظه را انجام می‌دهد، پاک‌سازی حافظه از اشیاء بلااستفاده از وظایف GC می‌باشد.

۴- Debug Engine: یکی از امکانات مورد نیاز برنامه نویسان، قابلیت خطایابی برنامه‌ها می‌باشد که این سرویس، قابلیت خطایابی یا همان Debugging را به برنامه نویسان می‌دهد.

۵- Type Checker : یکی از ویژگی‌های دات نت، Type Safety می‌باشد، بدین معنی که تمام Data Type ها یا نوع‌های داده باید مشخص باشند. برای مثال شما در دات نت نمی‌توانید در یک متغیر از نوع عددی، رشته‌ای را ذخیره کنید یا بالعکس، این سرویس وظیفه کنترل نوع‌های داده را به عهده دارد.

۶- Security Engine :

یکی دیگر از سرویس‌های CLR می‌باشد که وظیفه امنیت اجرای کدها در محیط دات نت را به عهده دارد.

۷- Exception Manager: هنگامی که در برنامه‌های مبتنی بر دات نت، خطایی اتفاق می‌افتد، CLR یک استثنا یا Exception برای آن خطا ایجاد می‌کند، وظیفه کنترل و مدیریت استثناها به عهده این سرویس می‌باشد.

۸- COM Marshaler : یکی از مزایای دات نت، قابلیت ارتباط با اجزای COM که در بخش قبلی در مورد آن صحبت کردیم می‌باشد، این سرویس وظیفه انتقال اطلاعات بین برنامه‌های دات نت و اجزای COM را به عهده دارد.

۹- Thread Support: یکی از ویژگی‌های برنامه‌های امروزی، قابلیت Multi-Threading جهت همزمانی انجام وظایف و عملیات‌ها می‌باشد که باعث بالارفتن کارایی برنامه می‌شود، این سرویس وظیفه مدیریت اجرای کدها به صورت Multi-Thread را به عهده دارد.

۱۰- Base Class Library : امکان دسترسی به کتابخانه پایه دات نت را به برنامه‌ها می‌دهد.

۷-۱- اجرای کدها در دات نت و سرویس JIT و اجرای مدیریت شده کدها

روند اجرای کدها در دات نت، کمی پیچیده ولی در عین حال، بسیار کارآمد است. زیرا طراحی این قسمت توسط شرکت مایکروسافت طوری انجام شده که اجرای کدها به بهینه ترین شکل ممکن انجام شود. همانطور که در قسمت قبلی گفتیم، سرویسی که وظیفه اجرای کدها در دات نت را دارد، کامپایلر JIT یا Just-In-Time می باشد. در زیر به صورت قدم به قدم به تشریح روند کامپایل و اجرای کدها، از زمان کامپایل توسط کامپایلر زبان برنامه نویسی تا زمان اجرای کد توسط JIT می پردازیم:

۱- در قدم اول شما کد مورد نظر خود را با زبان برنامه نویسی مورد نظرتان مانند سی شارپ می نویسید.

۲- در ادامه، با کمک کامپایلر زبان مورد نظر، کدی که شما نوشتید به زبان IL ترجمه شده و داخل یک فایل به نام Assembly قرار داده می شوند. Assembly ها فایل هایی هستند که بعد از کامپایل کدها توسط کامپایلر ایجاد می شوند Assembly. ها شامل کد IL برنامه، Metadata ها که اطلاعاتی در مورد کد نوشته شده توسط شما در اختیار CLR قرار می دهد و Assembly Manifest که اطلاعات اولیه در مورد اسمبلی مانند نسخه و ... می باشند.

۳- بعد از ایجاد Assembly توسط کامپایلر، با اجرای برنامه، CLR وارد صحنه می شود. بدین صورت که ابتدا توسط سرویس Class Loader که در بخش قبلی در مورد آن توضیح دادیم، اطلاعات اولیه مورد نیاز جهت اجرای کد مانند کد IL و Metadata را داخل حافظه بارگذاری شده، و سپس کد بارگذاری شده توسط Class Loader توسط JIT به کد ماشین تبدیل شده و اجرا می شود.

نکته ۱: هر اسمبلی دات نت برای اجرا نیاز به یک نقطه شروع یا Entry Point دارد که Class Loader در ابتدا این بخش، یعنی Entry Point را در حافظه بارگذاری کرده و JIT آن را اجرا می کند.

نکته ۲: برنامه‌های نوشته شده توسط دات نت، می‌توانند شامل چندین هزار خط و بخش مجزا باشند، از مزیت‌های JIT این است که برای اجرای برنامه‌ها، کل کدها را یکجا به کد ماشین تبدیل نمی‌کند، بلکه تنها بخش مورد نیاز برای اجرا توسط Class Loader در حافظه بارگذاری شده و توسط JIT اجرا می‌شود. با این کار سرعت بارگذاری اولیه برنامه‌های دات نت بسیار بالا می‌رود.

اما MCE یا Managed Code Execution دقیقاً "به چه معناست؟ همانطور که گفتیم، CLR

پروژه‌ای برای اجرای کدها دارد که به شرح زیر می‌باشد:

۱- بارگذاری کد IL و Meta Data های مربوطه

۲- تبدیل کد IL به کد Native و اجرای آن

۳- کنترل و مدیریت حافظه در زمان اجرای برنامه

۴- اعمال Type Safety

۵- کنترل امنیت در اجرای برنامه‌ها

۶- مدیریت Exception ها

پروژه‌ای فوق برای اجرای کدها در دات نت ذکر شد به MCE یا همان Managed Code Execution شناخته می‌شود.

۸-۱- مدیریت حافظه و سرویس GC

یکی از کابوس‌های برنامه‌نویسان، مدیریت و استفاده بهینه از حافظه کامپیوتر می‌باشد. اگر شما با زبان‌هایی که مدیریت حافظه در آنها به صورت خودکار نباشد، کار کرده باشید، حتماً واژه کابوس را تأیید می‌کنید. برنامه‌ای که از حافظه استفاده بهینه داشته باشد، سرعت اجرا و عملکرد بهتری دارد.

اما در دات نت، مسئله مدیریت و استفاده بهینه از حافظه بوسیله سرویس GC یا Garbage Collector به صورت کامل حل شده است. در حقیقت سرویس GC وظیفه مانیتور کردن حافظه در حین اجرای برنامه و پاک‌سازی آن را به عهده دارد. به همین دلیل شما به عنوان برنامه‌نویس، هیچ‌گونه نگرانی در مورد پاک‌سازی حافظه را نخواهید داشت. پروژه آزاد سازی حافظه در دات نت، کمی پیچیده است،

اما به طور کلی، در مدیریت حافظه توسط CLR ، دو وظیفه کلی وجود دارد. تخصیص حافظه و آزاد سازی حافظه که این وظایف به صورت خودکار توسط CLR انجام می شوند .

۹-۱- ساخت اولین پروژه

در ادامه میخواهیم نحوه ساخت یک پروژه وب سایت را شرح دهیم.

۱-ویژوال استودیو را اجرا کنید

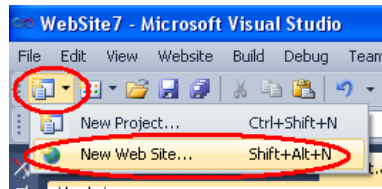
۲-منوی file/new/web site را انتخاب کنید و یا از گزینه موجود در جعبه ابزار (شکل ۱-۳) استفاده کنید تا شکل ۱-۴ روی صفحه ظاهر می شود. سپس گزینه asp.net empty web site را انتخاب کنید.

۳-یک پروژه جدید ایجاد شده و در Solution Explorer نام پروژه نمایش داده می شود(شکل ۱-۵).

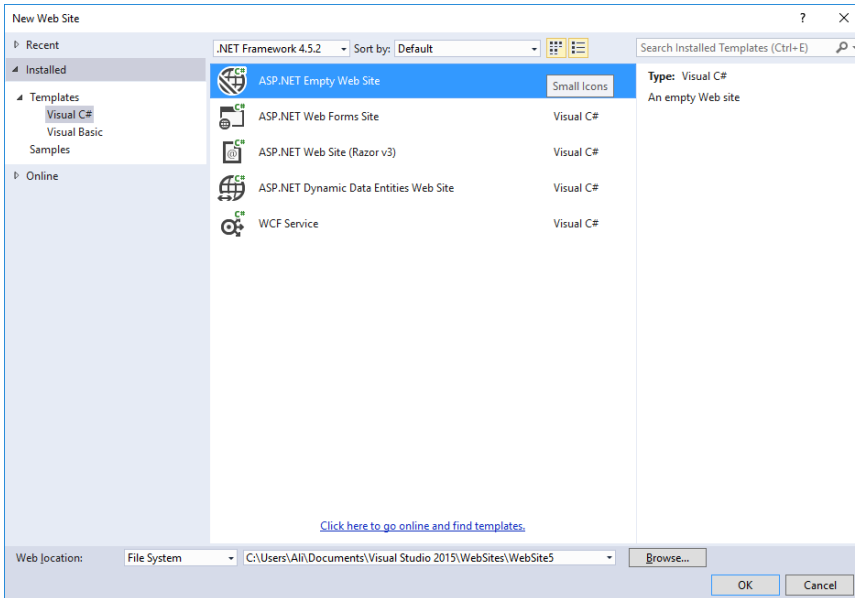
۴-روی نام پروژه (در اینجا web site7) کلیک راست کرده و گزینه add new item را انتخاب کنید تا شکل ۱-۶ روی صفحه ظاهر شود. روش دیگر استفاده از منوی add new web site/ item است.

۵-گزینه web form را انتخاب کرده و بعد از وارد کردن نام پروژه، روی دکمه add کلیک کنید.

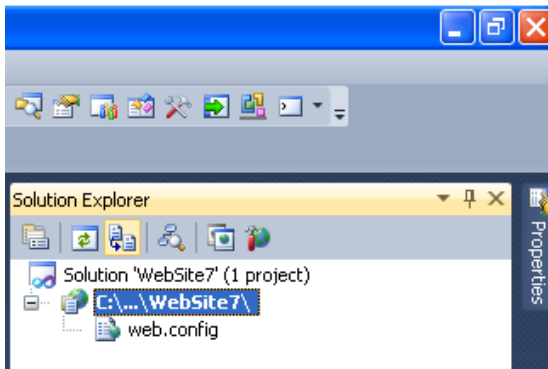
۶-مطابق با شکل ۱-۷ یک فایل aspx به پروژه اضافه شده است.



شکل ۱-۳:



شکل ۴-۱:

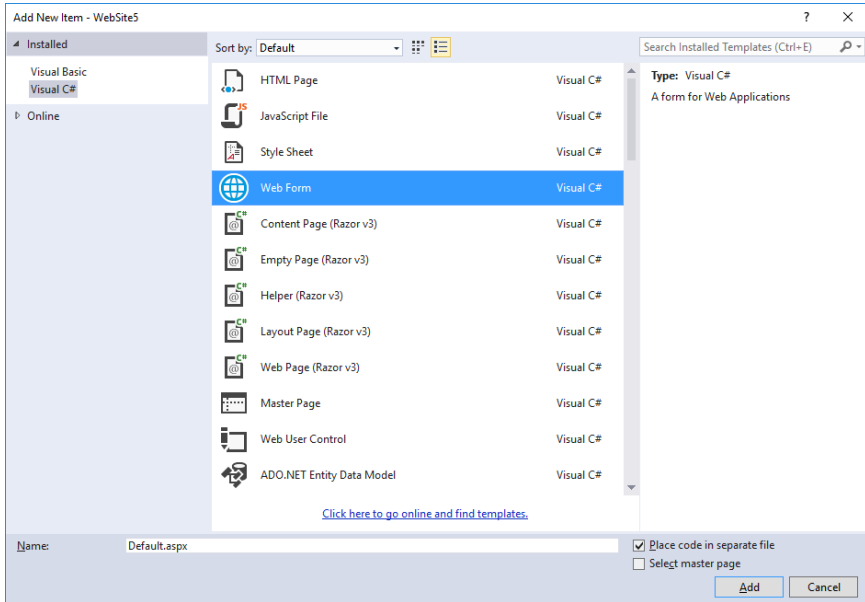


شکل ۵-۱:

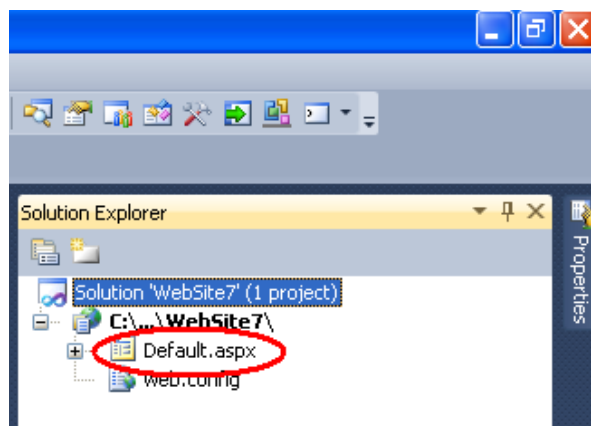
برای مشاهده محیط طراحی و کدهای منبع html، در گوشه سمت چپ و پایین محیط ویژوال استودیو، دو دکمه source و design وجود دارد (شکل ۱-۸) که با کمک آنها می‌توانید محیط طراحی و کدهای منبع html پروژه را مشاهده کنید.

شکل ۱-۹ محیط طراحی و شکل ۱۰ کد منبع ASP.NET را نمایش می‌دهد.

نکته: از طریق منوی view/code می توانید کد #c از طریق منوی view/designer می توانید به یک محیط طراحی بروید(شکل ۱-۱۰).



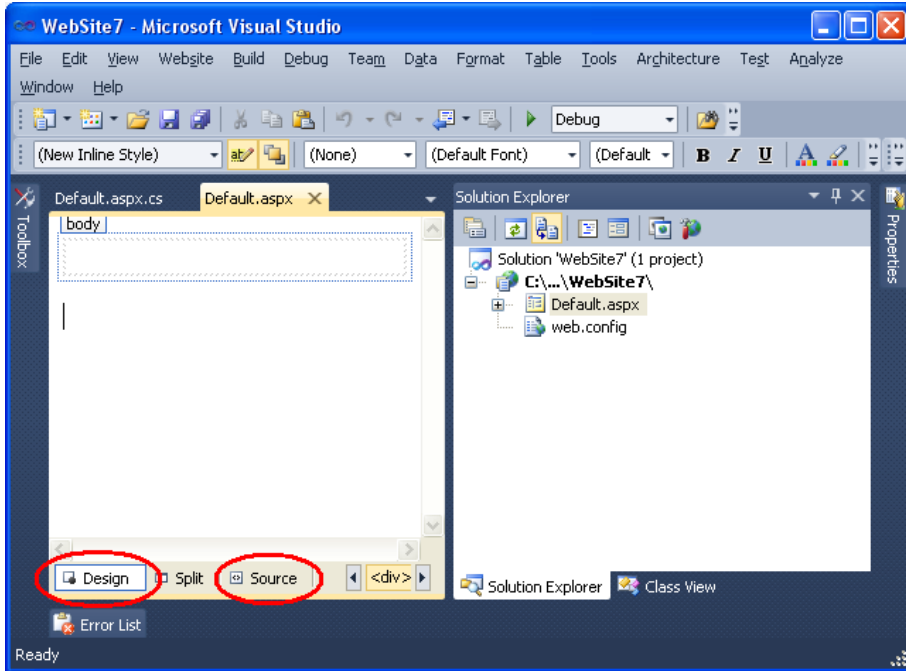
شکل ۱-۶



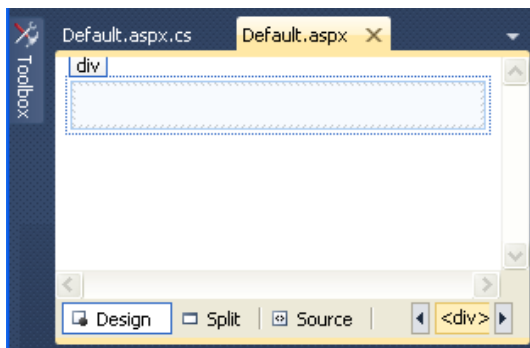
شکل ۱-۷

برای مشاهده کدهای C# پروژه، روی علامت + کنار نام پروژه در پنجره Solution explorer کلیک کنید. (شکل ۱-۱۱)

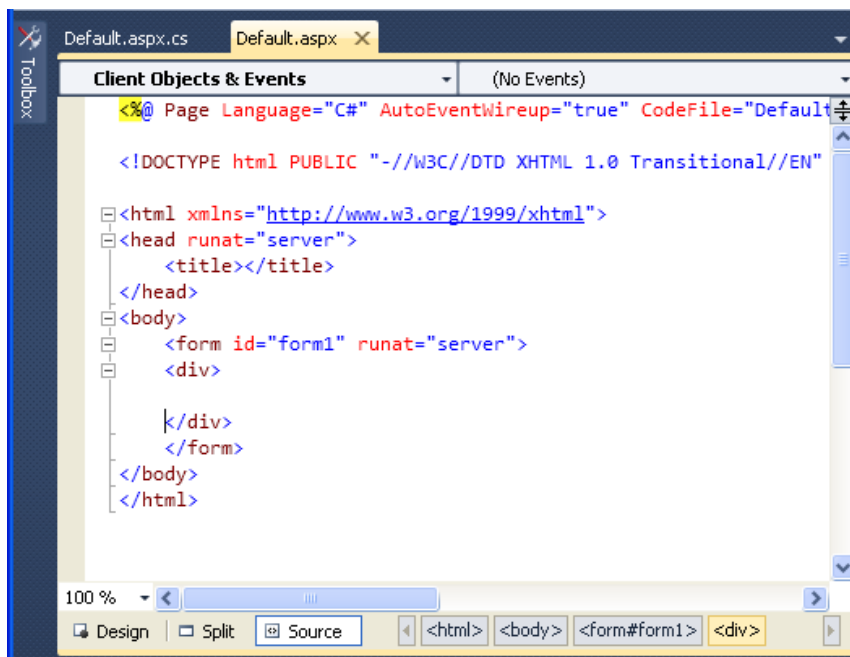
بعد از انجام این کار فایل C# در پنجره Solution explorer نمایش داده می‌شود (شکل ۱-۱۲)



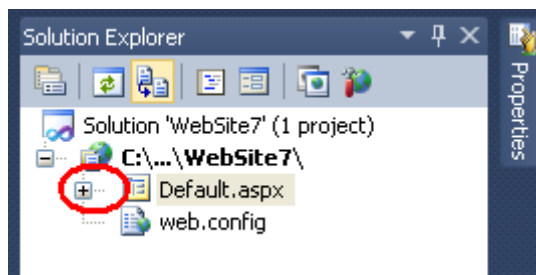
شکل ۱-۸



شکل ۱-۹

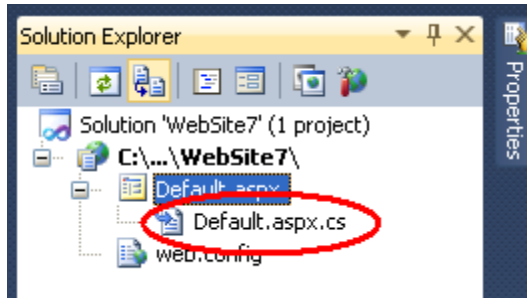


شکل ۱۰-۱

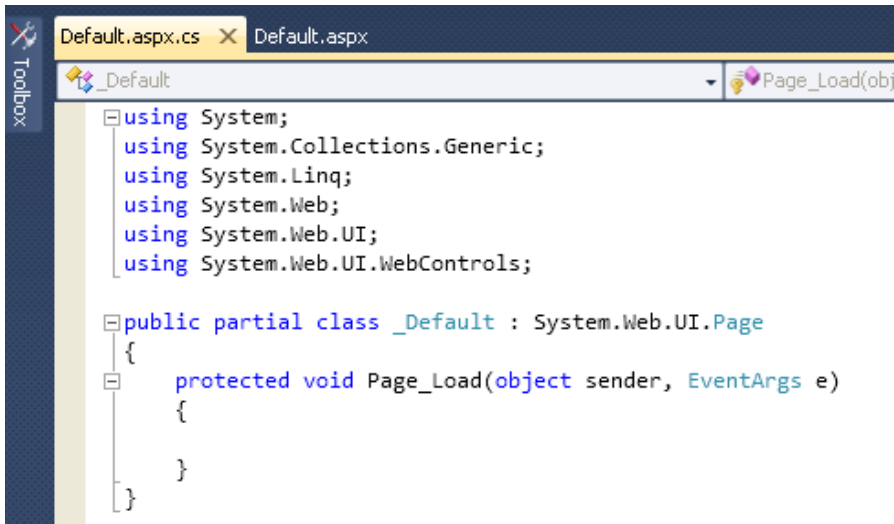


شکل ۱۱-۱

روی این فایل کلیک کنید تا محیط کد نویسی C# نمایش داده شود (شکل ۱۳-۱)



شکل ۱۲-۱



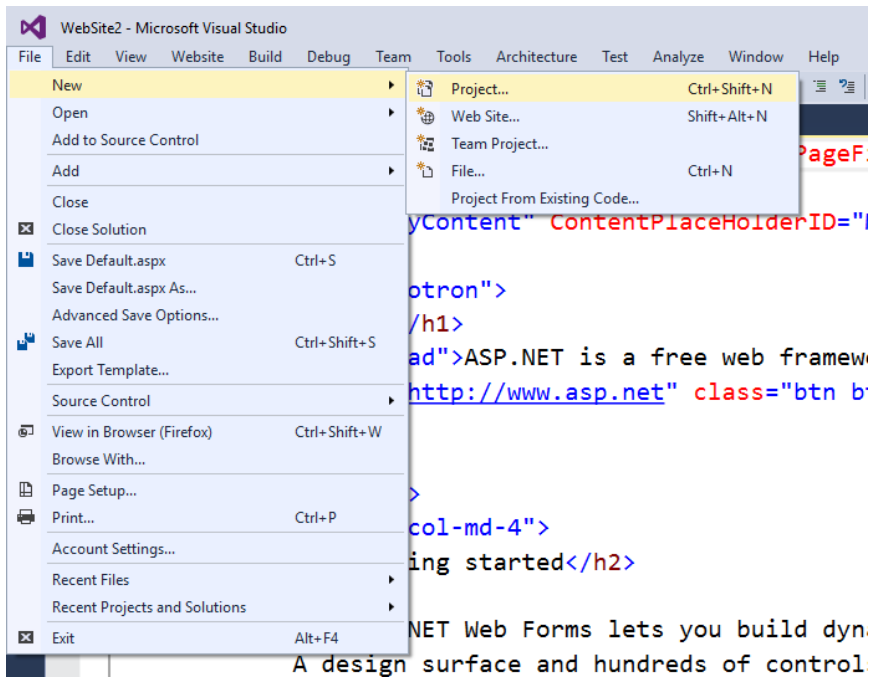
شکل ۱۳-۱

برای بازگشت به محیط طراحی یا کد نویسی html ، روی نام پروژه در Solution explorer کلیک کنید.

۱-۱۰- ساخت یک پروژه از نوع ASP.NET Web Application

۱- ویژوال استودیو دات نت را اجرا کنید.

۲- منوی File/New/Project را اجرا کنید(شکل ۱-۱۴).



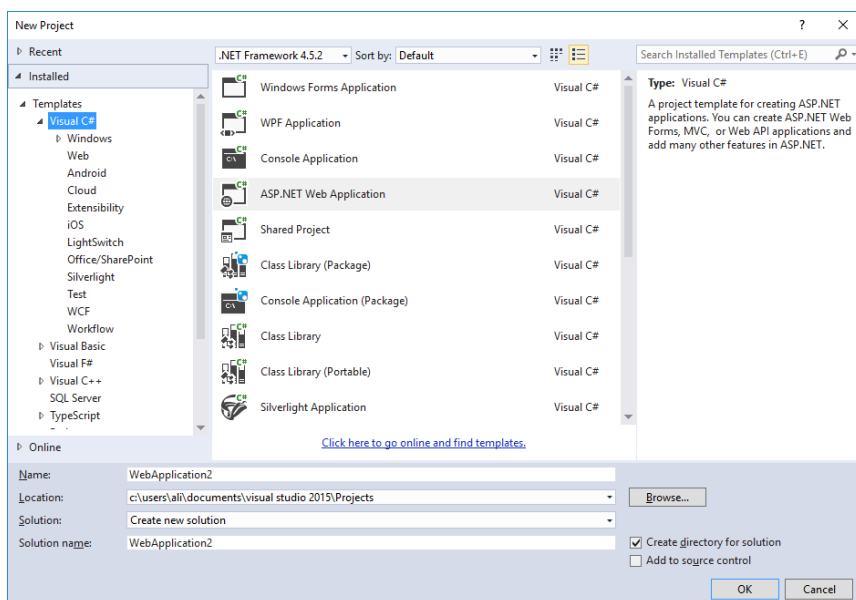
شکل ۱-۱۴

۳- پنجره ای با عنوان New Project روی صفحه تشکیل می‌شود(شکل ۱-۱۵).

۴- در قسمت installed ، Visual C# ، و در کادر مقابل آن ASP.NET Web Application را انتخاب کنید. نام پروژه را در کادر Name و مسیر ذخیره سازی آن را در کادر Location تعیین کنید.

۵- روی دکمه ok کلیک کنید. پروژه شما ایجاد شد.

در ادامه محیط توسعه و فایل‌های ایجاد شده در پروژه های WEB را بررسی می‌کنیم.



شکل ۱-۱۵

۱۱-۱- The Visual Studio IDE (محیط برنامه‌نویسی)

نویسی ویزوال استودیو

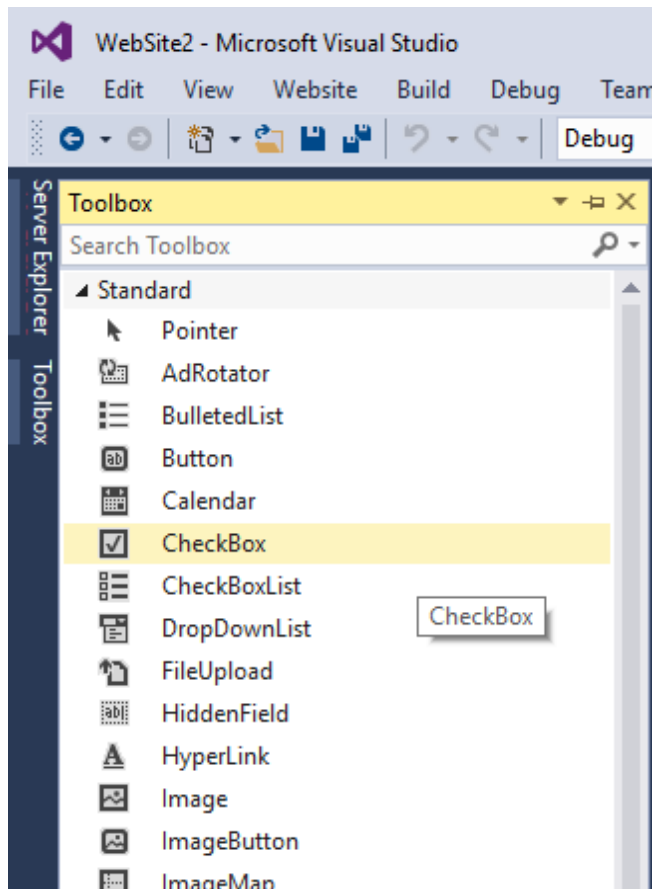
پنجره ی new project به شما امکان انتخاب application template (قالب برنامه ی کاربردی) دلخواه را از میان قالب های موجود می دهد.

وقتی که شما یک وب سایت جدید راه اندازی می کنید، ASP.Net پوشه ها و فایل های اولیه ی برای سایت که شامل دو فایل برای اولین web form از سایت هست را ارائه می دهد. فایل Default.aspx دربردارنده html و کد asp.Net است که فرم را تعریف می کنند.

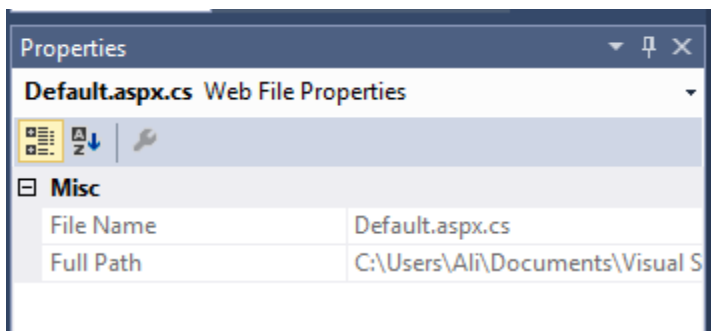
فایل های Default.aspx.cs (ویژه ی کد نویسی با زبان C#) دربردارنده ی کد زبانی هستند که قرار است با آن برنامه نویسی کنید. این کد در واقع مسئول عملیاتی است که روی فرم انجام می شود.

نخستین پنجره در محیط برنامه نویسی (IDE) Visual Studio پنجره Web Form Designer می باشد.

دیگر پنجره های پشتیبان عبارتند از toolbox (شکل ۱۶)، Solution Explorer و Properties window (شکل ۱۷-۱). شما از پنجره Web Form Designer برای طراحی یک وب استفاده می کنید و برای اضافه کردن کد به کنترل روی فرم طوریکه فرم طبق نیازهای شما کار کند، از ویرایش گر کد (code editor) استفاده می کنید.



شکل ۱۶-۱



شکل ۱-۱۷

نکته: اگر یک Component از tool box پاک شود می‌توانیم روی Tool box کلیک راست کرده و گزینه choose toolbox را انتخاب کنیم در برگ Net framework components تمامی کنترل‌ها را load می‌کند و ما می‌توانیم انتخاب کنیم.

۱۲-۱- کار با views و windows

در ادامه روش‌هایی برای کار با پنجره‌ی های مختلف ارائه می‌شود:
 برای تغییر Web Forms Designer از یک view به view دیگر بر روی دکمه Design یا source کلیک کنید.
 برای بستن یک پنجره روی دکمه close در بالا سمت راست پنجره کلیک کنید، و برای نمایش دوباره آن، پنجره را از منوی view انتخاب کنید.
 برای مخفی کردن یک پنجره روی دکمه ی Auto Hide آن کلیک کنید؛ پنجره به یک tab تغییر می‌کند، برای نمایش مجدد آن دوباره روی Auto Hide کلیک کنید.

۱۳-۱- افزودن فولدر و فایل به وب سایت خود

وقتی web form جدیدی ایجاد می‌شود، Visual Studio به صورت خودکار کدهای اولیه ی HTML را برای فرم اضافه می‌کند و آن را در Source view مربوط به فرم‌های وب نمایش