

بنام خداوند بخشنده مهربان

مقاله ای را که هم اکنون مشاهده و مطالعه می نمائید ، آموزش نگاهی سریع بر

.net دارد.

با تشکر ----- میلاد اسلامیزد
کپی برداری بدون ذکر نام منبع مجاز نیست

پارسی e-book

ایجاد یک برنامه وب نمونه در دات نت (بخش اول)

- مجموعه مقالات مبانی برنامه نویسی وب در دات نت
- ایجاد یک برنامه وب در دات نت
- رویدادها در برنامه های وب دات نت
- مجموعه مقالات فرم های وب
- مجموعه مقالات دستیابی به داده ها در دات نت
- دستیابی به داده ها در فرم های وب
- مجموعه مقالات ارتباط به منابع داده در **ADO.NET**
- مجموعه مقالات ذخیره و بازیابی داده توسط **ADO.NET**
- مجموعه مقالات امنیت برنامه های وب
- بررسی اعتبار داده ها توسط **VB.NET** در فرم های وب
- راهکارهای مدیریت وضعیت صفحات **ASP.NET**

در ابتدا لازم است یک مسئله را تعریف و در ادامه مراحل لازم برای حل آن را دنبال نمائیم :

- مسئله : ایجاد یک برنامه وب در دات نت (سیستم مقالات)
- توضیحات : در این برنامه کاربران در ابتدای طریق یک فرم Login تائید و پس از تائید آنان توسط برنامه ، امکان استفاده از پتانسیل های برنامه (مشاهده لیست مقالات و انجام ویرایش های لازم در رابطه با یک مقاله) در اختیار آنان قرار خواهد گرفت
- مراحل حل مسئله : جدول زیر فازهای متفاوت حل مسئله به همراه مراحل می بایست در هر فاز دنبال شود را نشان می دهد :

فاز	مراحل لازم برای تکمیل فاز
فاز اول : آشنائی با برنامه های وب دات نت	<ul style="list-style-type: none"> • مرحله اول : ایجاد و ذخیره سازی یک برنامه وب ASP.NET • مرحله دوم : ایجاد یک فرم وب
فاز دوم : استفاده از کنترل ها	<ul style="list-style-type: none"> • مرحله اول : افزودن کنترل های سرویس دهنده به فرم وب • مرحله دوم : آنالیز کدهای HTML تولید شده • مرحله سوم : بررسی صحت و اعتبار داده ها و ارائه پیام خطا • مرحله چهارم : افزودن اسکریپت • مرحله پنجم : نوشتن کد لازم در فایل Code Behind

فاز سوم : اشکال زدائی برنامه	<ul style="list-style-type: none"> • مرحله اول :فعال نمودن امکان دیباگ در برنامه های وب • مرحله دوم : تنظیم Break point • مرحله سوم : اشکال زدائی هوشمند • مرحله چهارم : استفاده از Debug.write
فاز چهارم : مدیریت وضعیت برنامه	<ul style="list-style-type: none"> • مرحله اول : مدیریت وضعیت برنامه سمت سرور گیرنده • مرحله دوم : مدیریت وضعیت برنامه سمت سرور دهنده
فاز پنجم : دستیابی به داده در فرم های وب	<ul style="list-style-type: none"> • مرحله اول : استفاده از کلاس SqlConnection • مرحله دوم : نمایش داده در DataGrid • مرحله سوم : Paging و مرتب سازی DataGrid • مرحله چهارم : دستورات ویرایش ، بهنگام سازی و ...
فاز ششم : بکارگیری برنامه وب	<ul style="list-style-type: none"> • مرحله اول : بکارگیری از طریق امکانات ویزوال استودیو دات نت
خلاصه و نتایج	

فاز اول : آشنائی با برنامه های وب دات نت

در این فاز با نحوه ایجاد یک برنامه وب و اضافه نمودن یک فرم وب (صفحه Login) ،

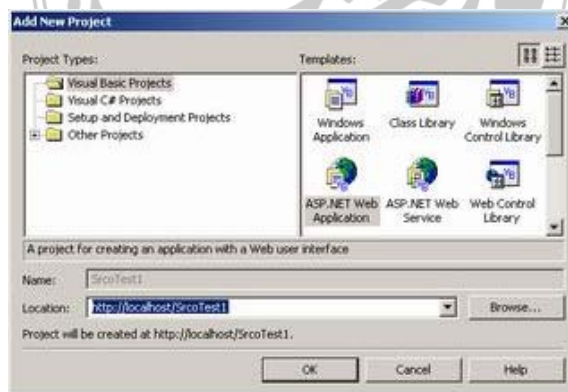
آشنا خواهیم شد
 مرحله اول : ایجاد و ذخیره سازی یک برنامه وب **ASP.NET** : در این مرحله ، یک

برنامه وب خالی (ساختار پیش فرض) ، ایجاد می گردد . برای ایجاد یک پروژه جدید
وب با نام SrcoTest1 ، مراحل زیر را دنبال می نمائیم :

- اجرای برنامه ویژوال استودیو
 - انتخاب **File|New|Project** و یا فشردن کلیدهای **CTRL+SHIFT+N**
 - انتخاب نوع پروژه **Projects Visual Basic** از طریق جعبه محاوره ای
- New Project**
- انتخاب **ASP.NET Web Application**، از طریق تمپلیت های ارائه

شده

- درج آدرس **http://localhost/SrcoTest1** در فیلد **Location** (نام در نظر گرفته شده برای پروژه ، اختیاری است) .
- تأیید عملیات با فشردن دکمه **OK**



توضیحات : در این مرحله ، ویژوال استودیو دات نت ، یک ریشه مجازی با نام

SrcoTest1 ، فایل **Global.asax** ، فایل **Global.asax.vb** ، فایل **Web.Config** ، فایل **WebForm1.aspx.vb** ، فایل **Style.css** و یک فرم وب

پیش فرض با نام WebForm1.aspx را ایجاد می نماید . ویژوال استودیو ، همچنین

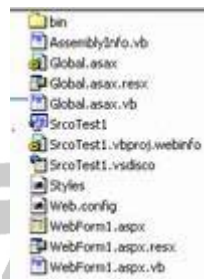
فایل های SrcTest1.disco ، فایل SrcTest1.vbproj و فایل

SrcTest1.vbproj.webinfo را نیز ایجاد می نماید .



ویژوال استودیو دات نت ، یک فولدر با نام SrcTest\1 در مسیر

\InetPub\wwwroot را ایجاد می نماید . محتوی این فولدر به صورت زیر است :



مرحله دوم : ایجاد یک فرم وب : در این مرحله ، فرم وب پیش فرض ایجاد شده در

مرحله قبل را تغییر نام داده و با نام جدید ذخیره می نمائیم .

• انتخاب Solution Explorer | Explorer View و یا فشردن دکمه های

CTRL+ALT+L

• کلیک سمت راست بر روی فایل WebForm1.aspx و انتخاب دستور

Rename (از طریق پنجره Solution Explorer)

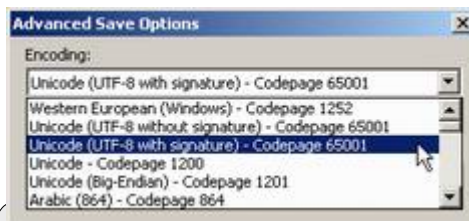
- تغییر نام فایل WebForm1.aspx به Login.aspx
- مشاهده محتویات صفحه با کلیک بر روی HTML Tab (در قسمت پائین صفحه)
- دایرکتیو @ Page در ابتدای صفحه پارامترهای متفاوتی را برای صفحه مشخص می نماید (مثلا) زبان استفاده شده برای نوشتن کد مرتبط با صفحه (

```

Login.aspx*
[Client Objects & Events] (No Events)
Page Language="vb" AutoEventWireup="false" Codebehind="Login.aspx.vb"
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<?xml:namespace prefix="o" ns="urn:schemas-microsoft-com:office:office" type="text/xml" />
<html>
<head>
<title>WebForm1</title>
<meta name="GENERATOR" content="Microsoft Visual Studio.NET 7.0">
<meta name="CODE_LANGUAGE" content="Visual Basic 7.0">
<meta name="vs_defaultClientScript" content="JavaScript">
<meta name="vs_targetSchema" content="http://schemas.microsoft.com/>
</head>
<body MS_POSITIONING="GridLayout">
<form id="Form1" method="post" runat="server">
</form>
</body>

```

- برای ذخیره فایل Login.aspx ، دستور Login.aspx File|Save و یا Login.aspx As File|Save را انتخاب و یا کلید CTRL+S را فعال می نمائیم . با توجه به این که قصد ایجاد فرم وب را با محتویات فارسی (رابط کاربر فارسی) داریم ، گزینه File|Save Login.aspx As را انتخاب و در ادامه با انتخاب گزینه Save with Encoding ، نوع Page Code را مشخص می نمائیم .



نتایج فاز اول :

- ایجاد یک پروژه جدید با نام **SrcoTest1**
- تغییر نام و ذخیره فرم وب پیش فرض (**WebForm1.aspx**) به

Login.aspx

در بخش دوم این مقاله به بررسی عملیات مرتبط با فاز دوم خواهیم پرداخت. در این رابطه با نحوه استفاده از کنترل های سرویس دهنده بر روی فرم وب آشنا خواهیم شد (بخش رابط کاربر یک برنامه وب) .

ایجاد یک برنامه وب نمونه در دات نت (بخش دوم)

در **بخش اول** این مقاله به بررسی یک برنامه نمونه دات نت پرداخته و با نحوه ایجاد یک برنامه وب آشنا شدیم . در این بخش به بررسی عملیات مرتبط با فاز دوم پرداخته و با نحوه استفاده از کنترل های سرویس دهنده بر روی فرم وب آشنا خواهیم شد (بخش رابط کاربر یک برنامه وب) .

parsi e-book
WWW.PARSIBOOK.4T.COM

فاز دوم : استفاده از کنترل ها

- مرحله اول : افزودن کنترل های سرویس دهنده به فرم وب
- مرحله دوم : آنالیز کدهای HTML ، تولید شده
- مرحله سوم : بررسی صحت و اعتبار داده ها و ارائه پیام خطا
- مرحله چهارم : افزودن اسکریپت
- مرحله پنجم : نوشتن کد لازم در فایل **Code Behind**

فاز دوم : استفاده از کنترل ها

در این فاز، از کنترل های سرویس دهنده و Validation (به منظور بررسی صحت داده ورودی توسط کاربر) ، استفاده کرده و کدهای مورد نیاز را هم در سمت سرویس گیرنده (اسکریپت های همراه فرم وب) و هم در سمت سرویس دهنده (کدهای نوشته شده در صفحه Code behind مربوط به صفحه Login) خواهیم نوشت و در نهایت تگ های HTML تولید شده را بررسی می نمائیم .

مرحله اول : افزودن کنترل های سرویس دهنده به فرم وب : در این مرحله ، با نحوه استفاده از کنترل های سرویس دهنده ASP.NET آشنا خواهیم شد (چهار کنترل Lable ، دو کنترل TextBox ، دو کنترل Button)

- سوئیچ به حالت **Design** (کلیک بر روی **Design Tab**)

- فعال نمودن پنجره **Toolbox** (انتخاب گزینه **Toolbox** از طریق منوی **View** و یا فشردن کلیدهای **CTRL+ALT+X**)
- انتخاب کنترل **Label** از طریق **Web Forms tab** موجود در **Toolbox** و استقرار آن بر روی فرم وب **Login.aspx** (در وضعیت **Design view**).



- تنظیم خصالت های زیر برای کنترل **Label** استفاده شده بر روی فرم وب .
برای مشاهده پنجره **Properties** مربوط به یک کنترل ، پس از انتخاب کنترل دستور **View|Properties** را فعال و یا از کلید **F4** ، استفاده می نمائیم .

تنظیم خصالت های کنترل Label اول	
مقدار	خصالت
lblHeading	ID
ورود به بخش مقالات سایت سختا روش	Text

- فرم وب Login.aspx پس از استقرار اولین کنترل سرویس دهنده Label بر روی آن :



- افزودن یک کنترل سرویس دهنده Label دیگر بر روی فرم وب Login و تنظیم خصت های آن

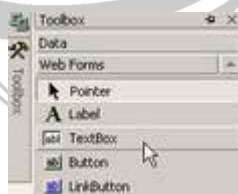
تنظیم خصت های کنترل Label دوم	
مقدار	خصت
lblUserID	ID
	Text نام

- افزودن یک کنترل سرویس دهنده Label دیگر بر روی فرم وب Login و تنظیم خصت های آن

تنظیم خصت های کنترل Label سوم	
مقدار	خصت
lblPassword	ID
رمز عبور	Text



- انتخاب کنترل TextBox از طریق Web Forms tab موجود در Toolbox و استقرار آن بر روی فرم وب Login.aspx (در مجاورت کنترل label مربوط به "نام")



تنظیم خصت های کنترل TextBox اول	
مقدار	خصت

txtUserID	ID
1	TabIndex
182px	Width



- انتخاب یک کنترل TextBox دیگر و استقرار آن بر روی فرم وب Login.aspx (در مجاورت کنترل label مربوط به "رمز عبور")

تنظیم خصت های کنترل TextBox	
دوم	
مقدار	خصت
txtPassword	ID
2	TabIndex
182px	Width
Password	TextMode



- انتخاب کنترل Button از طریق Web Forms tab موجود در Toolbox و استقرار آن بر روی فرم وب Login.aspx و تنظیم خصالت های آن



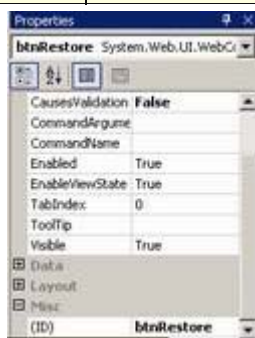
تنظیم خصالت های کنترل Button اول

مقدار	خصالت
btnSignin	ID
ورود به سایت	Text
3	TabIndex



- انتخاب کنترل Button دیگر و استقرار آن بر روی فرم وب Login.aspx و تنظیم خصت های آن

تنظیم خصت های کنترل Button دوم	
مقدار	خصت
btnRestore	ID
انصراف	Text
4	TabIndex
False	CausesValidation



- افزودن یک کنترل سرویس دهنده Label دیگر بر روی فرم وب Login و تنظیم خصلت های آن (از کنترل فوق برای نمایش پیام مورد نظر در زمان بررسی صحت داده ورودی توسط کاربر، استفاده می گردد) .

تنظیم خصلت های کنترل Label	
مقدار	خصلت
lblResult	ID
خالی	Text
Smaller	Font
Red	ForeColor

- فرم وب Login.aspx پس از استقرار کنترل های سرویس دهنده مورد نیاز بر روی آن :



- مرحله دوم : آنالیز کدهای HTML، تولید شده : در این مرحله با نحوه تبدیل کنترل های سرویس دهنده به عناصر HTML مختص قابل نمایش در مرورگر توسط ASP.NET runtime، آشنا خواهیم شد.

- برای مشاهده محتوی صفحه ، بر روی HTML Tab کلیک می نمائیم . کد زیر به صورت اتوماتیک برای کنترل های سرویس دهنده استفاده شده در فرم وب Login.aspx ایجاد می گردند. خصلت `runat="server"` ، نشاندهنده سمت سرویس دهنده بودن کنترل است.

HTML Code For Server Controls

```

<asp:Label id="lblUserID"      runat="server"
... >نام</asp:Label>

<asp:Label id="lblPassword"    runat="server"
...>عبور رمز</asp:Label>

<asp:TextBox id="txtUserID"    runat="server"
...></asp:TextBox>

<asp:TextBox id="txtPassword"  runat="server"
...></asp:TextBox>

<asp:Button id="btnSignin"     runat="server"
...></asp:Button>

<asp:Button id="btnRestore"
runat="server"...></asp:Button>

<asp:Label id="lblResult"      runat="server"
...></asp:Label>

```

- برای مشاهده خروجی در مرورگر ، گزینه `Debug | Start` را انتخاب و یا کلید `F5` را فعال می نمائیم .

- برای مشاهده خروجی HTML کنترل های سرویس دهنده ، گزینه Source را از طریق منوی View در مرورگر IE انتخاب می نمائیم :

HTML Output for Server Controls in browser

```
<span id="lblHeading" ... > ورود به بخش مقالات
</span>
<input name="txtUserID" type="text"
id="txtUserID" ... />
<input type="submit" name="btnSignin"
value="/ ... " />
```

- همانگونه که مشاهده می شود ، ASP.NET runtime ، کنترل های سرویس دهنده را به عناصر HTML مختص مرورگر ، تبدیل نموده است . Runtime ممکن است در صورت ضرورت از DHTML به منظور تعیین موقعیت و محل استقرار عناصر متفاوت HTML استفاده نماید .

نتایج مراحل یک و دو فاز دوم :

- افزودن کنترل های سرویس دهنده مورد نیاز بر روی فرم Login.aspx) چهار کنترل Lable ، دو کنترل TextBox ، دو کنترل Button (

- بررسی کد تولید شده در ویژوال استودیو دات نت برای کنترل های سرویس دهنده که به صورت اتوماتیک ، ایجاد می گردند .

- بررسی کد تولید شده توسط **Runtime ASP.NET** برای کنترل های

سرویس دهنده به منظور نمایش توسط مرورگر

کپی برداری بدون مجاز نیست
 نام منبع مجاز نیست
 parsi e-book

ایجاد یک برنامه وب نمونه در دات نت (بخش سوم)

آنچه تاکنون گفته شده است :

- **بخش اول** ایجاد یک برنامه و فرم وب (فاز اول)
- **بخش دوم** : استفاده از کنترل های سرویس دهنده و بررسی کد **HTML** تولید شده (فاز دوم)

فاز دوم : استفاده از کنترل ها

- مرحله اول : افزودن کنترل های سرویس دهنده به فرم وب
- مرحله دوم : آنالیز کدهای **HTML** ، تولید شده
- مرحله سوم : بررسی صحت و اعتبار داده ها و ارائه پیام خطا
- مرحله چهارم : افزودن اسکریپت
- مرحله پنجم : نوشتن کد لازم در فایل **Code Behind**

parsi e-book
 WWW.PARSIBOOK.4T.COM

در بخش دوم این مقاله ، مراحل اول و دوم تشریح گردید . در این بخش به بررسی مرحله سوم ، خواهیم پرداخت .

مرحله سوم : بررسی صحت و اعتبار داده ها و ارائه پیام خطا: در این مرحله با نحوه استفاده از کنترل های Validation به منظور بررسی صحت داده ورودی توسط کاربر ، آشنا خواهیم شد.

• سوئیچ نمودن به View Design (کلیک بر روی Design tab)

• استفاده از کنترل RequiredFieldValidator به منظور اطمینان از صحت

درج داده ورودی در کنترل txtUserID با توجه به سیاست های مورد نظر و مقداردهی مناسب خصلت های آن :

- انتخاب کنترل RequiredFieldValidator از طریق Forms tab Web

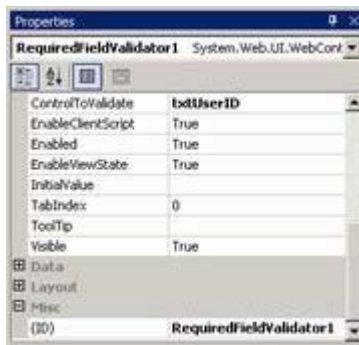
موجود در Toolbox و استقرار آن بر روی فرم وب Login.aspx (پائین کنترل txtUserID) .



• مقداردهی مناسب خصلت های کنترل RequiredFieldValidator .

تنظیم خصلت های کنترل RequiredFieldValidator	
اول	
مقدار	خصلت

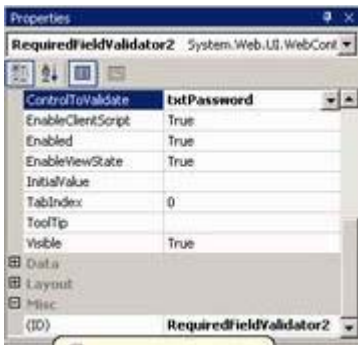
RequiredFieldValidator1	ID
txtUserID	ControlToValidate
نام کاربر می بایست وارد شود	Error Message



- استفاده از کنترل RequiredFieldValidator به منظور اطمینان از صحت درج داده ورودی در کنترل txtPassword با توجه به سیاست های مورد نظر و مقداردهی مناسب خصلت های آن :
- انتخاب کنترل RequiredFieldValidator از طریق Web Forms tab موجود در Toolbox و استقرار آن بر روی فرم وب Login.aspx (پائین کنترل . txtPassword)
- مقداردهی مناسب خصلت های کنترل . RequiredFieldValidator

parsi e-book
WWW.PARSIBOOK.4T.COM

تنظیم خصت های کنترل RequiredFieldValidator	
دوم	
مقدار	خصت
RequiredFieldValidator2	ID
TxtPassword	ControlToValidate
Password is required.	Error Message



- استفاده از کنترل ValidationSummary به منظور نمایش خلاصه

اطلاعات بررسی صحت داده های ورودی :

- انتخاب کنترل ValidationSummary از طریق Web Forms tab

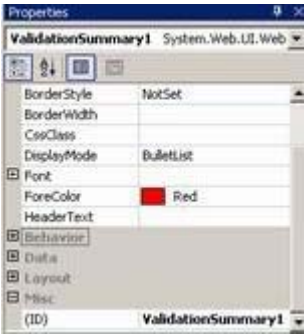
موجود در Toolbox و استقرار آن بر روی فرم وب Login.aspx (قسمت

پائین صفحه)



- مقداردهی مناسب خصلت های ValidationSummary .

تنظیم خصلت های کنترل ValidationSummary	
مقدار	خصلت
ValidationSummary1	ID
BulletList	DisplayMode



- فرم وب Login.aspx پس از استقرار کنترل های سرویس دهنده و کنترل های RequiredFieldValidator و ValidationSummary :



ok
WWW.PARSIBOOK.COM

- برای تست صحت داده ها ، گزینه Start را از طریق منوی Debug انتخاب و یا

کلید F5 را فعال می نمایم
 تست اول : کلیک بر روی دکمه ورود به سایت " ، بدون درج نام و رمز عبور



- تست دوم : کلیک بر روی دکمه " ورود به سایت " ، درج نام و بدون درج رمز

عبور



- تست سوم : کلیک بر روی دکمه " ورود به سایت " ، درج رمز عبور و بدون درج

نام



نتایج مراحل اول ، دوم و سوم فاز دوم :

- افزودن کنترل های سرویس دهنده مورد نیاز بر روی فرم **Login.aspx**)
چهار کنترل **Lable** ، دو کنترل **TextBox** ، دو کنترل **Button**)
- بررسی کد تولید شده در ویژوال استودیو دات نت برای کنترل های سرویس دهنده که به صورت اتوماتیک ، ایجاد می گردند .

- بررسی کد تولید شده توسط **Runtime ASP.NET** برای کنترل های

سرویس دهنده به منظور نمایش توسط مرورگر

- بررسی صحت داده ها

در بخش چهارم این مقاله به بررسی سایر مراحل تعریف شده در فاز دوم (افزودن اسکریپت و نوشتن کد لازم در فایل **Code Behind**) ، خواهیم پرداخت

ایجاد یک برنامه وب نمونه در دات نت (بخش ششم)

آنچه تاکنون گفته شده است :

- **بخش اول** ایجاد یک برنامه وب و فرم وب (فاز اول)
- **بخش دوم** : استفاده از کنترل های سرویس دهنده و بررسی کد HTML تولید شده (فاز دوم)
- **بخش سوم** : بررسی صحت و اعتبار داده ها (فاز سوم)
- **بخش چهارم** : افزودن اسکریپت سمت سرویس گیرنده و درج

کد در فایل Code Behind

- **بخش پنجم** : اشکال زدائی برنامه

در این بخش به بررسی عملیات مرتبط با فاز چهارم ، خواهیم پرداخت .

فاز چهارم : مدیریت وضعیت برنامه

برنامه های وب از معماری سرویس گیرنده و سرویس دهنده تبعیت می نمایند. این بدان معنی است که بخشی از برنامه بر روی کامپیوتر سرویس گیرنده و بخشی دیگر بر روی سرویس دهنده اجراء و با همیاری و همکاری یکدیگر ، اهداف تعریف شده در برنامه را دنبال می نمایند. سرویس گیرنده و سرویس دهنده برای ارتباط با یکدیگر می بایست از مجموعه ای قوانین مدون (پروتکل) استفاده نمایند. در این راستا از پروتکل HTTP ، استفاده می گردد . ماهیت پروتکل فوق بصورت Stateless می باشد، این بدان معنی

است که ارتباط ایجاد شده صرفاً برای یک تراکنش ، نگهداری می گردد .
 قبل از این که یک سرویس گیرنده و سرویس دهنده قادر به مبادله اطلاعات باشند ، می
 بایست بین آنان یک ارتباط (Connection) ایجاد گردد . سرویس گیرنده و سرویس
 دهنده با استفاده از پروتکل TCP/IP یک ارتباط را برقرار می نمایند . سرویس
 دهندگان و سرویس دهنندگان از پروتکل HTTP برای ارسال درخواست ها و پاسخ به
 درخواست ها ، استفاده می نمایند . سرویس دهنده و سرویس گیرنده ارتباط ایجاد شده
 را صرفاً برای یک تراکنش نگهداری و در این راستا ، چهار مرحله عملیات دنبال خواهد
 شد .

- **مرحله اول : ایجاد ارتباط .** در این مرحله با استفاده از پروتکل TCP/IP یک
 ارتباط بین سرویس گیرنده و سرویس دهنده برقرار می گردد . برای تشخیص
 نوع پروتکل ها ، برنامه ها از یک عدد صحیح منحصر بفرد که " شماره پورت " ،
 نامیده می شود ، استفاده می نمایند . جدول زیر برخی از پروتکل های رایج و
 شماره پورت نسبت داده شده به آنان را نشان می دهد.

پروتکل	شماره پورت
FTP	۲۱
Telnet	۲۳
SMTP	۲۵
HTTP	۸۰
...	...

- مرحله دوم: ایجاد یک درخواست توسط سرویس گیرنده . در دومین مرحله ، سرویس گیرنده اقدام به ایجاد یک درخواست و ارسال آن برای سرویس دهنده وب می نماید.
- مرحله سوم: پاسخ سرویس دهنده وب . پس از دریافت و تفسیر درخواست ارسالی، سرویس دهنده پاسخ لازم را برای سرویس گیرنده ارسال می نماید.
- مرحله چهارم: قطع ارتباط بین سرویس دهنده و سرویس گیرنده . سرویس دهنده مسئولیت قطع ارتباط با سرویس گیرنده پس از پردازش و پاسخگوئی به درخواست را برعهده دارد . مسئولیت فوق در انحصار سرویس دهنده نبوده و سرویس گیرنده نیز می بایست توانائی غیرفعال نمودن ارتباط ایجاد شده را داشته باشد . مثلاً در صورتی که شما بر روی دکمه Stop مرورگر خود کلیک نمائید ، مرورگر می بایست توانائی غیرفعال نمودن ارتباط ایجاد شده را دارا باشد .

برنامه های وب با توجه به واقعیت های موجود در رابطه با ماهیت پروتکل HTTP خصوصاً ویژگی Stateless آن می بایست از راهکاری متعددی به منظور نگهداری داده در بین تراکنش های متعدد بین سرویس گیرنده و سرویس دهنده استفاده نمایند (Round trips). اهمیت موضوع فوق تا به آن اندازه است که هر یک از تکنولوژی های موجود که از آنان به منظور پیاده سازی برنامه های وب استفاده می شود ، به مقوله " مدیریت وضعیت برنامه " (State Management) با دقت پرداخته و در این راستا گزینه های متعددی را در اختیار طراحان و پیاده کنندگان برنامه های وب قرار می دهند . با توجه به ماهیت برنامه های وب (جایگاه سرویس گیرنده و سرویس دهنده) ،

راهکارهای ارائه شده دارای گرایش سمت سرویس دهنده و یا سمت سرویس گیرنده می باشند .

با این مقدمه طولانی و خسته کننده ! به بررسی مدیریت وضعیت برنامه با تاکید بر امکانات ارائه شده سمت سرویس گیرنده و سرویس دهنده در دات نت ، خواهیم پرداخت . در این راستا هدف ، تشریح تمامی امکانات ارائه شده نبوده و صرفاً " قصد داریم با ذکر نمونه هایی به بررسی دو رویکرد فوق (سمت سرویس گیرنده ، سمت سرویس دهنده) ، بپردازیم .

- **مرحله اول : مدیریت وضعیت برنامه سمت سرویس گیرنده**

- **مرحله دوم : مدیریت وضعیت برنامه سمت سرویس دهنده**

مرحله اول : مدیریت وضعیت برنامه سمت سرویس گیرنده : با توجه به ماهیت Stateless بودن پروتکل استفاده شده برای ارتباط بین سرویس گیرنده و سرویس دهنده ، می بایست از امکانات و روش های خاصی برای نگهداری مقادیر متغیرهای یک صفحه و کنترل ها بر روی سرویس گیرنده و یا سرویس دهنده ، استفاده نمود. برای نگهداری داده بر روی سرویس گیرنده از ViewState استفاده می شود . در این مرحله با نحوه استفاده از شی ViewState برای نگهداری مقدار " رمز عبور " بر روی سرویس گیرنده استفاده می شود . در این رابطه مراحل زیر را دنبال می نمایم :

- استفاده از شی ViewState برای نگهداری داده درج شده در فیلد " رمز عبور "

. بدین منظور کد زیر را در انتهای متد btnSignIn_Click ، اضافه می نمایم

بدین ترتیب پس از فعال شدن متد فوق ، مقدار درج شده در فیلد "رمز عبور" ،
بر روی سرویس گیرنده ، نگهداری می گردد .

Client-Side State Management

```
Private Sub btnSignin_Click (ByVal sender As  
System.Object, ByVal e As System.EventArgs) Handles  
btnSignin.Click  
.....  
ViewState("Password") = txtPassword.Text  
End Sub
```

- سوئیچ به Design view (انتخاب View | Designer و یا فعال نمودن کلیدهای SHIFT + F7)
- کلیک (Double) بر روی دکمه " انصراف " و افزودن کد زیر در تابع مسئول به پاسخگودی به رویداد btnRestore_Click

Client side State management using ViewState

```
Private Sub btnRestore_Click(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles btnRestore.Click  
lblResult.Text = " " & CStr  
(ViewState("Password"))  
End Sub
```

- اجرای برنامه (| Start Debug و یا فعال نمودن کلید F5)
- درج داده ورودی در فیلدهای " نام " و " رمز عبور " (Src,Srcopassword)

- کلیک بر روی دکمه "ورود به سایت"
- پس از فعال نمودن دکمه "ورود به سایت"، یک round trip به سرور می‌دهنده محقق شده و مقدار فیلد "رمز عبور" در شی ViewState نگهداری می‌گردد (فیلد txtPassword خالی است).
- کلیک بر روی دکمه "انصراف"
- پس از فعال نمودن دکمه فوق، رمز عبور ورودی در lblResult، نمایش داده می‌شود.



- مرحله دوم: مدیریت وضعیت برنامه سمت سرور دهنده: در این مرحله از شی Session برای نگهداری نام درج شده در فیلد "نام"، بر روی سرور دهنده استفاده می‌نمائیم. بدین منظور مراحل زیر را دنبال می‌نمائیم:

- سوئیچ به صفحه behind Code مربوط به صفحه Login.aspx (انتخاب View | Code و یا فعال نمودن کلید F7)

- استفاده از شی Session برای نگهداری مقدار درج شده در فیلد "نام" (افزودن کد زیر به متد (btnSignin_Click)

کپی برداری بدون ذکر نام منبع مجاز نیست
narsi e-book

Server side State management using Session

```
Private Sub btnSignin_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnSignin.Click
    If (txtUserID.Text.Length >= 4 And
txtPassword.Text.Length >= 4 And
txtUserID.Text.StartsWith("S")) Then
        lblResult.Text = "کاربر مجاز"

        Session("UserID") = txtUserID.Text
        Response.Redirect("LoginDemo.aspx")
    Else
        lblResult.Text = "مجاز کاربر غیر"
    End If
    ViewState("Password") = txtPassword.Text
End Sub
```

- برای دستیابی به مقدار session در صفحه ای دیگر، یک فرم جدید را ایجاد می نمائیم :

مرحله اول : فعال نمودن پنجره Solution Explorer (انتخاب گزینه

Explorer Solution از طریق منوی View)

مرحله دوم : انتخاب Project | Add Web Form برای افزودن یک فرم وب

جدید (نمایش جعبه محاوره ای (Add New Item

مرحله سوم : انتخاب نام مورد نظر برای فرم وب (LoginDemo.aspx) و

درج آن در فیلد Name و فعال نمودن دکمه Open



مرحله چهارم : پس از انجام مراحل فوق ، فرم وب LoginDemo.aspx

ایجاد و به پروژه اضافه می گردد .

• برای نمایش نام " ، یک کنترل Label را بر روی فرم LoginDemo.aspx

ایجاد می نمایم .

مرحله اول : سوئیچ به حالت Design (کلیک بر روی Design Tab)

مرحله دوم : فعال نمودن پنجره Toolbox (انتخاب گزینه از

طریق منوی View و یا فشردن کلیدهای CTRL+ALT+X

مرحله سوم: انتخاب کنترل Label از طریق Web Forms tab موجود در

Toolbox و استقرار آن بر روی فرم وب

مرحله چهارم: تنظیم خصالت های زیر برای کنترل Label استفاده شده بر روی

فرم وب. (برای مشاهده پنجره Properties مربوط به یک کنترل ، پس از

انتخاب کنترل دستور View Properties را فعال و یا از کلید F4 ، استفاده

می نمائیم) .

تنظیم خصالت های کنترل Label	
مقدار	خصالت
lblUserID	ID
شما با این نام وارد سایت سخا روش شده اید	Text

• سوئیچ به صفحه behind Code مربوط به صفحه LoginDemo.aspx)

انتخاب View | Code و یا فعال نمودن کلید F7

• افزودن کد زیر در مدت Page_Load (بازیابی مقدار "نام" از طریق Session

ایجاد شده در صفحه Login.aspx)

Page_Load method

```

Private Sub Page_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
    If Not Page.IsPostBack Then
        lblUserID.Text = lblUserID.Text & " " &
Session("UserID")
    End If
End Sub

```

- اجرای برنامه (| Start Debug و یا فعال نمودن کلید F5)
- درج داده ورودی در فیلدهای "نام" و "رمز عبور" (Src,Srcpassword)
- کلیک بر روی دکمه "ورود به سایت" (نام درج شده در فیلد "نام" ، در Session ذخیره می گردد)
- در ادامه کاربر به صفحه دیگر (LoginDemo.aspx) هدایت و از طریق صفحه فوق پیام : " " ، نمایش داده می شود .



parsi e-book
WWW.PARSIBOOK.4T.COM

نتایج فاز چهارم :

- آشنائی با مفهوم **stateless** بودن پروتکل **HTTP**
 - استفاده از **ViewState** برای نگهداری داده (سمت سرویس گیرنده)
 - استفاده از شی **Session** برای نگهداری داده (سمت سرویس دهنده)
- در بخش هفتم این مقاله به بررسی سایر فازهای تعریف شده برای حل مسئله نمونه ، خواهیم پرداخت .



parsi e-book
WWW.PARSIBOOK.4T.COM

ایجاد یک برنامه وب نمونه در دات نت (بخش نهم)

آنچه تاکنون گفته شده است :

- **بخش اول** ایجاد یک برنامه وب و فرم وب (فاز اول)
- **بخش دوم** : استفاده از کنترل های سرویس دهنده و بررسی کد HTML تولید شده (فاز دوم)
- **بخش سوم** : بررسی صحت و اعتبار داده ها (فاز سوم)
- **بخش چهارم** : افزودن اسکریپت سمت سرویس گیرنده و درج

کد در فایل Code Behind

- **بخش پنجم** : اشکال زدائی برنامه
- **بخش ششم** : مدیریت وضعیت برنامه
- **بخش هفتم** : دستیابی به داده در فرم های وب
- **بخش هشتم** : دستیابی به داده در فرم های وب

در بخش هفتم این مقاله به مراحل اول و دوم (استفاده از کلاس SqlConnection و نمایش داده در DataGrid) و در بخش هشتم به مرحله سوم (Paging و مرتب سازی DataGrid) مربوط به فاز پنجم ، اشاره گردید . در این بخش به بررسی مرحله چهارم ، خواهیم پرداخت (ویرایش ، بهنگام سازی) .

فاز پنجم : دستیابی به داده در فرم های وب

- **مرحله اول** : استفاده از کلاس SqlConnection

- مرحله دوم : نمایش داده در DataGrid

- مرحله سوم : Paging و مرتب سازی DataGrid

- مرحله چهارم : دستورات ویرایش و بهنگام سازی

مرحله چهارم : دستورات ویرایش و بهنگام سازی مندر این مرحله با نحوه اضافه نمودن

یک ستون به کنترل DataGrid به منظور ویرایش و بهنگام سازی داده ، آشنا می

شویم.

افزودن یک ستون به DataGrid

- سوئیچ به View Design مربوط به صفحه LoginDemo.aspx (انتخاب

Designer View | و یا فعال نمودن کلید (SHIFT+F7

- کلیک بر روی HTML tab برای مشاهده محتوی صفحه . افزودن کد زیر بین

<asp:Datagrid/> و <asp:Datagrid> . (کد فوق یک ستون و با نام

"ویرایش" و از نوع LinkButton ایجاد می نماید)

Creates a custom column

```
<asp:Datagrid id="dgArticles" ... >  
<Columns>  
  <asp:EditCommandColumn ButtonType="LinkButton"  
UpdateText="ویرایش" CancelText="انصراف" EditText="بهنگام"  
</asp:EditCommandColumn>  
</Columns>
```

</asp:Datagrid>

- سوئیچ به Design View (کلیک بر روی Design Tab)
- شکل DataGrid در فرم وب :

شماره مقاله	عنوان	لینک	تاریخ
Databound	Databound	Databound	Databound
Databound	Databound	Databound	Databound
Databound	Databound	Databound	Databound
Databound	Databound	Databound	Databound
Databound	Databound	Databound	Databound

- اجرای برنامه (| Start Debug و یا فعال نمودن کلید F5)
- درج داده ورودی در فیلدهای "نام" و "رمز عبور" (Src,Srcopassword)
- کلیک بر روی دکمه "ورود به سایت"
- در صورت تائید کاربر ، وی به صفحه LoginDemo.aspx ، هدایت و از طریق صفحه فوق ، داده موجود در جدول SrcArticles در DataGrid (به همراه یک ستون جدید با نام "ویرایش") نمایش داده می شود.



7 تمهیدات لازم برای نمایش پیام به کاربر پس از بهنگام سازی داده

- انتخاب View | Toolbox و یا فعال نمودن کلید های CTRL+ALT+X
- انتخاب کنترل Label از طریق Web Forms tab موجود در Toolbox و استقرار آن بر روی فرم وب LoginDemo.aspx (در وضعیت Design view) و در قسمت پائین DataGrid
- تنظیم خصلت های زیر برای کنترل Label استفاده شده بر روی فرم وب . (برای مشاهده پنجره Properties مربوط به یک کنترل ، پس از انتخاب کنترل دستور View|Properties را فعال و یا از کلید F4 ، استفاده می نمائیم) .

تنظیم خصلت های کنترل Label	
مقدار	خصلت
lblMessage	ID
خالی	Text

افزودن امکان ویرایش ، بهنگام سازی ، انصراف در DataGrid

WWW.PARSIBOOK.4T.COM

- سوئیچ به صفحه behind Code مربوط به صفحه LoginDemo.aspx)

انتخاب View | Code و یا فعال نمودن کلید F7)

- انتخاب dgArticles از طریق Drop Down مربوط به اسامی کلاس ها

- انتخاب متد EditCommand از طریق Drop Down مربوط به اسامی

متدها . در ادامه ویزوال استودیو یک Event handler با نام

dgArticles_EditCommand ، ایجاد می نماید



- انتخاب CancelCommand از طریق Drop Down مربوط به اسامی متدها

. در ادامه ویزوال استودیو یک Event handler با نام

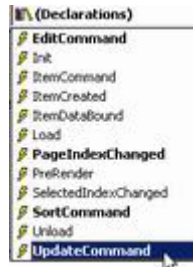
dgArticles_CancelCommand ، ایجاد می نماید .



- انتخاب UpdateCommand از طریق Drop Down مربوط به اسامی متدها

. در ادامه ویزوال استودیو یک Event handler با نام

dgArticles_UpdateCommand ، ایجاد می نماید .



- کپی برداری بدون ذکر نام منبع مجاز نیست
- پارسی e-book
- افزودن کد زیر در متد `dgArticles_EditCommand` (امکان ویرایش `DataGrid` فراهم می گردد) .

EditCommand method

```
Private Sub dgArticles_EditCommand(ByVal source As Object,
ByVal e As_
System.Web.UI.WebControls.DataGridCommandEventArgs)
Handles dgArticles.EditCommand
dgArticles.EditRowIndex = CInt(e.Item.ItemIndex)
BindGrid("dgArticles")
End Sub
```

- افزودن کد زیر در متد `dgArticles_CancelCommand` (امکان انصراف از عملیات ویرایش در `DataGrid` فراهم می گردد) .

CancelCommand method

```
Private Sub dgArticles_CancelCommand(ByVal source As
Object, ByVal e As_
System.Web.UI.WebControls.DataGridCommandEventArgs)
```

```
Handles dgArticles.CancelCommand
```

```
dgArticles.EditItemIndex = -1
```

```
BindGrid("dgArticles")
```

```
End Sub
```

- افزودن کد زیر در متد dgArticles_UpdateCommand (امکان بهنگام سازی داده در DataGrid فراهم می گردد)

UpdateCommand method

```
Private Sub dgArticles_UpdateCommand(ByVal source As Object,  
ByVal e As _
```

```
System.Web.UI.WebControls.DataGridCommandEventArgs) Handles  
dgArticles.UpdateCommand
```

```
Dim updateCmd As String
```

```
updateCmd = "UPDATE SrcoArticles SET Title = @Title where  
ArticleID = " & _
```

```
dgArticles.DataKeys(CInt(e.Item.ItemIndex))
```

```
Dim myCommand As SqlCommand
```

```
myCommand = New SqlCommand(updateCmd, myConnection)
```

```
myCommand.Parameters.Add(New SqlParameter("@Title",  
SqlDbType.NVarChar, 80))
```

```
Dim CurrentTextBox As System.Web.UI.WebControls.TextBox
```

```
CurrentTextBox = e.Item.Cells(1).Controls(0)
```

```

myCommand.Parameters("@Title").Value = CurrentTextBox.Text
myCommand.Connection.Open()
Try
    myCommand.ExecuteNonQuery()
    lblMessage.Text = "<b> " عنوان مقاله &
dgArticles.DataKeys(CInt(e.Item.ItemIndex)) & " بهنگام گردید </b><br>"
    dgArticles.EditItemIndex = -1
Catch exp As Exception
    lblMessage.Text = "ERROR: " & exp.Message
End Try

myCommand.Connection.Close()
BindGrid("dgArticles")
End Sub

```

- اجرای برنامه (| Start Debug و یا فعال نمودن کلید F5)
- درج داده ورودی در فیلدهای "نام" و "رمز عبور" (Src,Srcpassword)
- کلیک بر روی دکمه "ورود به سایت"
- در صورت تائید کاربر ، وی به صفحه LoginDemo.aspx ، هدایت و از طریق صفحه فوق ، داده موجود در جدول SrcArticles در DataGrid نمایش داده می شود(به همراه لینک ویرایش)
- کلیک بر روی لینک "ویرایش" در هر یک از سطرهای DataGrid :



- پس از نمایش DataGrid (قابل ویرایش) ، عنوان مقاله را تغییر و در ادامه لینک " بهنگام " را فعال تا تغییرات مورد نظر در بانک اطلاعاتی ذخیره گردد. پس از بهنگام سازی ، پیام " عنوان مقاله < شماره مقاله > بهنگام گردید " از طریق کنترل IblMessage نیز نمایش داده می شود.



parsi e-book
WWW.PARSIBOOK.4T.COM

نتایج مرحله چهارم فاز پنجم :

- آشنائی با نحوه ویرایش و بهنگام سازی داده موجود در یک **DataGrid** در بخش دهم (آخر) این مقاله به بررسی عملیات مربوط به فاز ششم ، خواهیم پرداخت
- ایجاد یک برنامه وب نمونه در دات نت (بخش دهم)

آنچه تاکنون گفته شده است :

- بخش اول ایجاد یک برنامه وب و فرم وب (فاز اول)
- بخش دوم : استفاده از کنترل های سرویس دهنده و بررسی کد **HTML** تولید شده (فاز دوم)
- بخش سوم : بررسی صحت و اعتبار داده ها (فاز سوم)
- بخش چهارم : افزودن اسکریپت سمت سرویس گیرنده و درج کد در فایل **Code Behind**
- بخش پنجم : اشکال زدائی برنامه
- بخش ششم : مدیریت وضعیت برنامه
- بخش هفتم : دستیابی به داده در فرم های وب
- بخش هشتم : دستیابی به داده در فرم های وب
- بخش نهم : دستیابی به داده در فرم های وب

در آخرین بخش از مجموعه مقالات " ایجاد یک برنامه وب نمونه در دات نت " ، به بررسی نحوه بکارگیری و نصب یک برنامه وب ، خواهیم پرداخت .

فاز ششم : بکارگیری برنامه وب

بکارگیری و یا استقرار (Deployment) ، یک برنامه وب بدین معنی است که برنامه بر روی سرویس دهنده ای که می بایست اجراء گردد ، نصب شود. با توجه به این که برنامه های وب بر روی سرویس دهندگان اجراء می گردند، معمولاً دارای یک برنامه "نصب" ، نمی باشند. از این زاویه برنامه های وب دارای تفاوت اساسی با برنامه های Desktop نوشته شده در محیط ویندوز می باشند . این نوع برنامه ها (برنامه های Desktop) از طریق CD-ROM و یا سایر رسانه های ذخیره سازی توزیع و با استفاده از یک برنامه "نصب" بر روی کامپیوتر مورد نظر مستقر شده و در ادامه عناصر مورد نیاز خود را بر روی ماشین کاربر ، رجیستر می نمایند . در زمان استقرار یک برنامه وب بر روی سرویس دهندگان ، می بایست امکانات لازم نرم افزاری نیز بر روی آن نصب و پیکربندی شده باشد تا زمینه اجرای برنامه های وب بر روی سرویس دهنده ، فراهم گردد .

امکانات نرم افزاری مورد نیاز برای اجرای یک برنامه وب دات نت

برنامه های وب دات نت ، تحت IIS اجرا شده و می بایست قبل از این که سرویس دهنده قادر به میزبان نمودن برنامه های وب گردد ، IIS و فریمورک دات نت بر روی آن نصب شده باشد. برنامه های وب دات نت برای اجراء ، نیازمند عناصر نرم افزاری زیر می باشند :

- IIS نسخه پنج و یا بالاتر

- فریمورک دات نت

پس از اطمینان از نصب عناصر فوق بر روی سرویس دهنده می توان یک فولدر مجازی را برای برنامه وب با دنبال نمودن مراحل زیر، ایجاد نمود :

- ایجاد یک فولدر فیزیکی بر روی سرویس دهنده برای ذخیره سازی برنامه

- ایجاد یک دایرکتوری مجازی با استفاده از IIS . در این رابطه وبسازد "Virtual

Directory Creation" را به منظور ایجاد یک فولدر مجازی برای محل

فیزیکی ایجاد شده در مرحله قبل ، اجرا می نمائیم . برای آشنائی با جزئیات

مراحل مربوطه می توانید به مقاله " ایجاد یک برنامه وب در دات نت " ، مراجعه

نمائید .

محدوده عملیاتی یک برنامه وب با استفاده از ساختار دایرکتوری برنامه مشخص می

گردد. این بدان معنی است که شروع یک برنامه از فولدر ریشه بوده و تمامی فولدرهای

زیر مجموعه را نیز حمایت می نماید.

مدل های متفاوت سرویس دهندگان

برنامه های وب که دارای تعداد بسیار زیادی کاربر می باشند و یا حجم بسیار بالائی از

داده را ارائه می نمایند ، می بایست به موازات رشد و افزایش کاربران ، توانائی و قابلیت

های خود را ارتقاء نمایند (Scalability) برنامه های وب از ویژگی فوق حمایت نموده

و می توان امکان اجرای آنان را بر روی سیستمی با چندین پردازنده ، سرویس دهندگانی

با چندین پردازنده و یا بر روی چندین سرویس دهنده ، فراهم نمود. در این رابطه می توان به سه حالت متفاوت اشاره نمود :

- نصب بر روی سرویس دهنده ای که دارای یک پردازنده می باشد (یک پردازنده و چندین پردازنده) .
- نصب بر روی سرویس دهنده ای با چندین پردازنده . برنامه وبی که بر روی سرویس دهنده ای با چندین پردازنده ، اجراء می گردد را Web garden می گویند . در صورتی که یک سرویس دهنده دارای یک و یا چندین پردازنده می باشد ، می توان برای یک برنامه وب دات نت مشخص نمود که از تمامی پردازنده ها و یا از برخی از آنان استفاده نماید . در این رابطه می بایست خصلت WebGarden مربوط به عنصر < ProcessModel > در فایل Machine.config ، مقدار True را پیدا نموده و خصلت CpuMask نیز دارای مقدار مناسب باشد . خصلت فوق از یک bit mask برای اجرای برنامه وب بر روی هر پردازنده استفاده می نماید . مثلاً "۱۱۰۱" ، باعث می شود که برنامه وب بر روی پردازنده های شماره صفر ، دوم و سوم یک ماشین که دارای چهار پردازنده می باشد، اجراء گردد . در صورتی مقدار خصلت فوق " Xfffffff" باشد ، برنامه وب با استفاده از توان تمامی پردازنده های موجود بر روی یک سرویس دهنده ، اجراء خواهد شد .

parsi e-book
WWW.PARSIBOOK.4T.COM

Machine.Config file

```
<ProcessModel
  enable="true"
  timeoutt="infinite"
  ...
  WebGarden =
  "true"

  CpuMask="0xffffffff"
  ....
/>
```

- نصب بر روی چندین سرور دهنده . برنامه وبی که بر روی چندین سرور دهنده اجرا می گردد را Web farm می گویند . برای این که چندین سرور دهنده قادر به پاسخگویی درخواست هائی برای یک آدرس HTTP خاص ، باشند ، می بایست سرور Load balancing در شبکه نصب گردد . سیستم های عامل Windows 2000 Advanced Server و Windows 2000 Data Center دارای نرم افزار (Network NLB Load Balancing) برای توزیع درخواست ها بر روی چندین سرور دهنده می باشند . پس از فعال شدن سرور فوق (NLB) در شبکه ، می توان برنامه وب را بر روی چندین سرور دهنده نصب نمود . در چنین وضعیتی

همواره درخواست کاربران به صورت اتوماتیک در اختیار سرویس دهنده ای که دارای مشغله کمتری است ، قرار داده می شود .

جایگاه فایل های **Web.Config** و **Machine.config** در یک برنامه وب

برنامه های وب از فایل های پیکربندی متنی (فایل هایی با انشعاب `config` .) برای مشخص نمودن نحوه اجراء ، استفاده می نمایند . یک فایل پیکربندی برنامه وب (`Web.config`) در دایرکتوری ریشه مربوط به برنامه وب مستقر می گردد . فایل فوق ، متاثر از تمامی تعاریف و تنظیمات انجام شده در فایل های پیکربندی زیر خواهد بود (وراثت) . اولویت تنظیمات و تعاریف انجام شده در صورت وجود تعاریف مشترک و یکسان با فایل های پیکربندی سطوح پائین تر است (فایل `Machine.Config` بالاترین سطح و به دنبال آن فایل `Web.Config` موجود در فهرست ریشه دایرکتوری IIS وجود دارد)

- فایل `Machine.config` موجود در آدرس

`. windows\Microsoft.NET\Framework\version\config`

تنظیمات انجام شده در فایل فوق برای تمامی اسمبلی های دات نت که بر روی سرویس دهنده اجراء می گردند ، در نظر گرفته می شود .

- فایل `Web.Config` . موجود در آدرس دایرکتوری ریشه IIS . تنظیمات انجام

شده ، تنظیمات پایه برای تمامی برنامه های وب بوده و می تواند بر روی تنظیمات انجام شده در فایل `Machine.Config` که به وی به ارث می رسد ، تاثیر و تعاریف و تنظیمات خاص خود را در این رابطه ارائه نماید .

- هر نوع فایل Web.Config موجود در دایرکتوری یک برنامه

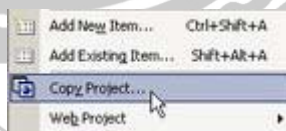
در صورت اعمال تغییرات در یک فایل پیکربندی برنامه وب (فایل Web.config) ، برنامه IIS ، به صورت اتوماتیک برنامه وب را راه اندازی و تغییرات مورد نظر را در رابطه با آن اعمال خواهد کرد.

پارسی برداری بدون ذکر نام منبع مجاز نیست
parsie-book

بکارگیری برنامه وب

برای استقرار یک برنامه وب بر روی سرویس دهنده می توان از XCOPY ، FTP و یا امکانات ارائه شده توسط ویژوال استودیو دات نت ، استفاده نمود . در ادامه با نحوه استفاده از امکانات ویژوال استودیو در این رابطه ، آشنا خواهیم شد .

- انتخاب Project | Copy Project (در ادامه جعبه محاوره ای Copy Project نمایش داده می شود)



- درج مسیر "http://localhost/SrcoFinal" در فیلد Destination
- project folder

- انتخاب گزینه File Share به عنوان Web access method
- درج آدرس "c:\inetpub\wwwroot\SrcoFinal" در فیلد Web access method

پارسی برداری بدون ذکر نام منبع مجاز نیست
parsie-book
WWW.PARSIBOOK.4T.COM

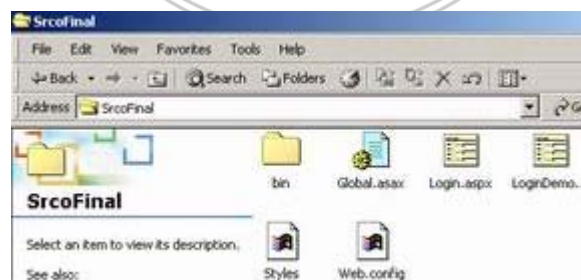
- انتخاب گزینه (پیش فرض) **Only files needed to run this**

Copy در application

- پس از انجام مراحل فوق ، شکل جعبه محاوره ای **Copy Project** به صورت زیر خواهد بود



- پس از تأیید عملیات (فشردن دکمه OK) ، ویژوال استودیو دات نت ، یک فولدر با نام SrcoFinal در مسیر c:\inetpub\wwwroot را ایجاد نموده و صرفاً فایل های مورد نیاز برای اجرای برنامه وب را در آن کپی می نماید .



- اجرای برنامه وب از طریق مرورگر (<http://localhost/SrcoFinal/login.aspx>)



نتایج فاز ششم :

- آشنائی با امکانات لازم به منظور اجرای یک برنامه وب بر روی سرویس دهنده
- آشنائی با مدل های متفاوت سرویس دهندگان
- آشنائی با جایگاه فایل های پیکربندی در یک برنامه وب
- آشنائی با نحوه بکارگیری و استقرار یک برنامه وب بر روی سرویس دهنده

خلاصه و جمع بندی مقاله

در مجموعه مقالاتی که با عنوان " ایجاد یک برنامه وب نمونه در دات نت " ، ارائه گردید با موارد زیر آشنا شدیم :

- آشنائی با برنامه های وب دات نت

- استفاده از کنترل ها

- اشکال زدائی برنامه

- مدیریت وضعیت برنامه

- دستیابی به داده در فرم های وب

- بکارگیری برنامه وب

منابع :

برای تهیه مجموعه مقالات " ایجاد یک برنامه وب نمونه در دات نت " از منابع زیر

استفاده شده است: *کتاب برداری بدون ذکر نام منبع مجاز نیست*

parsi e-book

- سایت مایکروسافت
- MSDN
- Kit Microsoft Visual Basic .NET 2003 Resource
- Microsoft .NET Framework SDK
- ASP.NET Resource Kit



parsi e-book
WWW.PARSIBOOK.4T.COM