

بسم الله الرحمن الرحيم

## آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

کنترل های سرور – ASP.NET

مدرس : مهندس افشین رفوآ

کنترل های سرور – ASP.NET

**Control** ها در واقع یکی از اجزای اصلی تشکیل دهنده رابط گرافیکی کاربری (**graphical user interface**) هستند، که شامل **textbox** ها (کادر متن)، **checkbox** ها (کادر تیک)، **button** ها (دکمه)، **listbox** ها، **label** ها و چندین ابزار مختلف دیگر می شود. با استفاده از این ابزار، کاربر قادر خواهد بود داده های مورد نیاز را وارد کرده و انتخاب کند، همچنین تنظیمات دلخواه خود را اعمال کند.

**Control** ها برای اهداف دیگری نیز بکار گرفته می شوند که از میان آن ها می توان به اعتبارسنجی، دسترسی به داده، امنیت، ایجاد **master page** ها (صفحات الگو) و دستکاری داده اشاره کرد.

**ASP.NET** از پنج نوع مختلف **web control** (کنترل وب) بهره می گیرد که به شرح زیر می باشند :

HTML controls

HTML Server controls

ASP.NET Server controls

ASP.NET Ajax Server controls

User controls and custom controls

کنترل های سرور بیش از دیگر کنترل های ذکر شده در تکنولوژی **ASP.NET** کاربرد داشته و در واقع جزء کنترل های اولیه و اصلی آن می باشند. کنترل های سرور را می توان به ترتیب زیر گروه بندی کرد:

**Validation controls** – این کنترل ها به منظور اعتبار سنجی ورودی کاربر (**user input**) بکار گرفته شده و با اجرای اسکریپت های سمت سروریس گیرنده (**client-side**) کار می کنند.

**Data source controls** – این کنترل ها داده را به **data source** های متفاوتی متصل (**bind**) می کند.

**Data view controls** - کنترل های نام برده در حقیقت **list**ها (فهرست ها) و **table**ها (جداول) هستند که برای اینکه نمایش داده شوند از **data source** ها به داده متصل (**bind**) شوند.

**Personalization controls** - این کنترل جهت شخصی سازی و تنظیم (**personalization**) صفحه مطابق با تنظیمات دلخواه کاربر و بر مبنای اطلاعات کاربر بکار گرفته می شود.

**Login and security controls** - کنترل مذکور در واقع کاربر را احراز هویت می کند (**authentication**).

**Master pages** - صفحات الگو یک **layout** (طرح کلی) و **interface** (رابط) همسان و یکپارچه ارائه می دهند که سرتاسر برنامه ی مورد نظر بکار فته می شود.

خاصیت	شرح
AccessKey	فشردن این کلید به همراه دکمه ی Alt باعث می شود که control مورد نظر انتخاب شود (focus به کنترل مورد نظر انتقال داده شود).
Attributes	مجموعه ای از خصیصه های (attribute) قراردادی که (فقط به منظور render یا اجرا کردن بکار گرفته شده) و با خواص (property) روی کنترل ارتباطی ندارد.
BackColor	رنگ پس زمینه (background color) را تعیین می کند.
BindingContainer	این خاصیت ارجاعی به Control object می دهد که دربردارنده ی اطلاعات مربوط به data binding (مقید سازی داده) کنترل جاری می باشد.
BorderColor	رنگ خط (Border color) حاشیه را تعیین می کند.
BorderStyle	استایل خط حاشیه (border style) را تعیین می کند.
BorderWidth	عرض خط حاشیه (Border width) را تعیین می کند.
CausesValidation	با استفاده از این خاصیت می توان تعیین کرد آیا پس از فشردن کنترل Button اعتبارسنجی (validation) برای کلاینت و سرور انجام گیرد یا خیر.
ChildControlCreated	نشانیگر این است که آیا کنترل های فرزند server control ایجاد شده اند یا خیر.
ClientID	شناسه ی control را برای HTML markup (صفحه ی html یا صفحه ی نشانیگذاری شده با html) برمی گرداند.
Context	این خاصیت امکان دسترسی به شیء HttpContext را برای web request جاری فراهم می کند / HttpContext object مربوط به کنترل سرور مورد نظر را برمی گرداند.
Controls	مجموعه تمامی کنترل های موجود در کنترل مورد نظر را برمی گرداند (get).
ControlStyle	Web server control را سبک دهی می کند. این خاصیت کلیه ی خواص کلاس webControl را که ظاهر کنترل را مشخص می کند را کپسوله سازی می کند.
CssClass	می توان با این خاصیت کلاس های CSS را به تگ های ASP اعمال کرد.
DataItemContainer	در صورتی که کلاس مذکور رابط (interface) IDataItemContainer را پیاده سازی کرده باشد، یک ارجاع به آن برمی گرداند.

DataKeysContainer	چنانچه کلاس مزبور interface (رابط) IDataKeysControl را پیاده سازی کرده باشد، یک ارجاع (reference) به آن را بازمی گرداند.
DesignMode	خاصیت designmode زمانی که برنامه نویس در حال اصلاح و ویرایش صفحه ی ASP.NET در محیط ویژوال استودیو است روی مقدار true تنظیم می شود.
DisabledCssClass	به کمک این خاصیت می توان اسمی که برای خاصیت DisabledCssClass تک تک وب کنترل ها render شده را تغییر داد. مقدار پیش فرض "aspNetDisabled" می باشد. کلاس CSS را هنگامی که control غیر فعال شده است، به المان render شده ی HTML اعمال می کند.
Enabled	فعال یا غیر فعال بودن کنترل را مشخص می کند.
EnableTheming	این خاصیت نشان می دهد که آیا تم ها به کنترل مربوطه اعمال می شوند یا خیر.
EnableViewState	مقداری برگردانده یا تنظیم می کند که این مقدار مشخص می کند آیا کنترل سمت سرور view state خود و نیز کنترل های فرزندی که دربر می گیرد را حفظ (ماندگار) کند یا خیر.
Events	لیست یا آرایه ای از delegate های event handler برای کنترل مورد نظر برمی گرداند. این خاصیت از نوع EventHandlerList هست که با استفاده از آلوگوریتیم جستجو خطی (linear search algorithm) ورودی (entry) های مورد نظر را در فهرست delegate ها پیدا می کند.
Font	Font. فونت یا قلم نوشته ی متنی که توسط کنترل نمایش داده می شود را تنظیم می کند.
ForeColor	Foreground color. با استفاده از این خاصیت می توان رنگ پس زمینه ی Web server control را

	تعیین کرد.
HasAttributes	مقداری برمی گرداند که نشان می دهد کنترل مورد نظر دارای مجموعه خصیصه ها (attribute set) هست یا خیر. این خاصیت در صورتی که نمونه ی کلاس WebControl دارای key-value pair باشد، true برمی گرداند.
HasChildViewState	این خاصیت مقداری را برمی گرداند که این مقدار مشخص می کند آیا کنترل های فرزند server control جاری اطلاعات view state را ذخیره کرده اند یا خیر.
Height	ارتفاع (طول) کنترل را بر حسب پیکسل یا با درصد، گرفته و تنظیم می کند.
ID	شناسه ی برنامه ای (programmatic ID) تخصیص داده شده به کنترل سرور را تنظیم (set یا get) می کند.
IsChildControlStateCleared	مقداری بر می گرداند که این مقدار نشان می دهد آیا کنترل های موجود در کنترل مورد نظر دارای Control state هستند یا خیر
IsEnabled	مقداری می گیرد که این مقدار نشان می دهد کنترل در رابط کاربری فعال شده است یا خیر.
IsTrackingViewState	این خاصیت مقداری را بازمی گرداند و این مقدار نشان می دهد آیا کنترل سرور تغییرات را در view state خود ذخیره می کند یا خیر
IsViewStateEnabled	مقداری باز می گرداند که این مقدار تعیین می کند آیا view state برای این کنترل فعال سازی شده است یا خیر.
LoadViewStateById	این خاصیت مقداری را برمی گرداند که به وسیله ی آن تعیین می کند آیا کنترل مورد نظر view state خود را بجای اندیس (index) با شناسه (ID) بارگذاری کند یا خیر.
Page	Page containing the control. ارجاعی (reference) به نمونه ی Page که میزبان (دربردارنده ی) کنترل سرور مورد نظر می باشد باز می گرداند.
Parent	Parent control. ارجاعی به control پدر server control در سلسله مراتب page control برمی

	<p>گرداند.</p> <p>توضیح: هر زمان که صفحه ای درخواست می شود، در واقع سلسله مراتبی از کنترل های سرور بر روی آن صفحه ساخته می شود. این خاصیت به شما امکان می دهد کنترل پدر server control جاری را در آن سلسله مراتب پیدا کرده و مشخص کنید.</p>
RenderingCompatibility	<p>این خاصیت ویرایش یا نسخه ی ASP.NET که HTML اجرا شده با آن سازگار هست را مشخص می کند.</p>
Site	<p>اطلاعاتی را درباره ی ظرفی که میزبان کنترل کنونی به هنگام render شدن آن روی design surface می باشد، برمی گرداند.</p> <p>خاصیت site یک شیء Component را به یک شیء container متصل کرده و امکان ارتباط بین این دو را فراهم می آورد. همچنین راهی برای مدیریت component ها توسط ظرف (container) مورد نظر ایجاد می کند.</p>
SkinID	<p>این خاصیت مشخص می کند کدام یک از Skin های موجود (skin های موجود در skin file ها که در پوشه ی theme قرار دارند)، به control مورد نظر اعمال شود.</p>
Style	<p>مجموعه ای از خصیصه های متن (text attribute) را که به عنوان خصیصه ی style بر روی تگ خارجی web server control اجرا شده و نمایش داده می شود، برمی گرداند. از این خاصیت به منظور مدیریت خصیصه های style که در تگ خارجی کنترل سرور وب render می شود، استفاده می شود.</p>
TabIndex	<p>ترتیب پیمایش (tab index) کنترل های سرور وب را در یک صفحه گرفته یا تنظیم می کند.</p>
TagKey	<p>مقدار HtmlTextWriterTag که مربوط به این web server control هست را برمی گرداند.</p>
TagName	<p>اسم المان html که در حال اجرا و نمایش (render) شدن است را گرفته یا تنظیم می کند.</p>
TemplateControl	<p>قالبی (template) که کنترل مورد نظر را دربردارد را برمی گرداند.</p>
TemplateSourceDirectory	<p>دایرکتوری (پوشه) مجازی صفحه یا کنترل دربردارنده ی control مورد نظر را بازمی</p>

	گرداند. آدرس (مسیر) صفحه یا user control ی که دربردارنده ی کنترل جاری است را مشخص می کند.
ToolTip	متنی که هنگام قرار گرفتن مکان نمای موس روی کنترل سرور وب ( web server control ) نمایش داده می شود را گرفته یا تنظیم می کند.
UniqueID	Unique identifier. شناسه ی منحصر بفرد که به صورت سلسله مراتبی برای server control تعریف شده اند را برمی گرداند.
ViewState	یک دیکشنری از اطلاعات state های مختلف که ViewState یک server side control را در بین request های یک صفحه یکسان ذخیره و بازیابی می کنند می گیرد. یک دیکشنری از اطلاعات مربوط به وضعیت (state) که وضعیت view های یک کنترلگر سرور را سرتاسر چندین request برای یک صفحه ذخیره و بازیابی می کند را برمی گرداند.
ViewStateIgnoreCase	با استفاده از این خاصیت می توان مشخص کرد آیا شیء StateBag به کوچک و بزرگی حروف غیر حساس باشد یا خیر.
ViewStateMode	با بکارگیری این خاصیت می توان view state را برای کنترل ها به صورت تکی فعال کرد حتی اگرهم view state برای این صفحه غیر فعال شده باشد.
Visible	بیان گر اینکه آیا server control روی صفحه قابل رویت می باشد یا خیر.
Width	عرض server control را مقداردهی (set یا get) می کند.

**Navigation controls** - این کنترل ها در پیمایش کمک شایانی ارائه می دهند. از میان این کنترل ها می توان به منوها و نمایش درختی (**tree view**) اشاره کرد.

**Rich controls** - ویژگی ها و امکانات فراوانی را ارائه می دهند، از جمله ی آن ها می توان از **AdRotator**، **FileUpload** و **Calendar control** سخن بمیان آورد.

ساختار نحوی (**syntax**) استفاده از کنترل های سرور به ترتیب زیر می باشد:

```
<asp:controlType ID="ControllID" runat="server" Property1=value1 [Property2=value2] />
```

Visual studio با ارائه ی ویژگی ها و امکانات زیر به کدنویسی هرچه صحیح تر و عاری از خطا کمک می کند:

کشیدن و جای گذاری (drag & drop) کنترل ها در **design view**

قابلیت **IntelliSense** که خاصیت ها (**property**) را نشان داده و به صورت خودکار تکمیل می کند

پنجره ی **properties** که امکان تنظیم مقادیر **property** را به صورت مستقیم فراهم می کند

**server control (property)** خواص

کنترل های سرور **ASP.NET** از کلاس **WebControl** مشتق شده و همچنین تمامی خاصیت ها، رخدادها و متدهای این کلاس را به ارث می برد.

خود کلاس **WebControl** و برخی دیگر از **server control** هایی که به صورت بصری اجرا نشده و نمایش داده نمی شوند (**render**) در واقع از کلاس **System.Web.UI.Control** مشتق می شوند، به عنوان نمونه می توان **XML control** و **Placeholder control** را نام برد.

**Server control** های **ASP.NET** کلیه ی خواص (**property**)، رخدادها (**events**) و متدهای کلاس **WebControl** و **System.Web.UI.Control** را به ارث می برد.

جدولی که زیر مشاهده می کنید این خواص به ارث گرفته شده که بین کنترل های سرور مشترک می باشد را به نمایش می گذارد :

Method	Description
AddAttributesToRender	خصیصه ها و سبک های (styles و attributes) html که باید اجرا شده و نمایش داده شود (render) را به HtmlTextWriterTag اضافه می کند.
AddedControl	این تابع پس از اینکه کنترل فرزند به مجموعه کنترل های (control collection) شیء control اضافه شد، فراخوانی می شود.
AddParsedSubObject	این تابع به کنترل سرور اطلاع می دهد که یک المان HTML / XML parse شده و سپس المان مزبور را به مجموعه کنترل های (control)



	collection) server control اضافه می کند.
ApplyStyleSheetSkin	خواص (property های) style تعریف شده در page style sheet (صفحه ی سبک دهی) را به کنترل مورد نظر اعمال می کند.
ClearCachedClientID	این API (رابط برنامه سازی کاربردی) از زیر ساخت NET Framework پشتیبانی می کند. توجه داشته باشید که نباید آن را مستقیماً از کد خود اجرا کنید. این تابع مقدار ClientID به صورت موقت ذخیره (cache) شده را روی NULL تنظیم می کند (مقدار null را به آن تخصیص می دهد).
ClearChildControlState	اطلاعات control-state کنترل های فرزند server control را پاک می کند. این متد زمانی بکار می رود که اطلاعات control-state فرزند که در control state کنترل پدر نوشته شده با ایجاد کنترل های فرزند جدید بازنویسی (override) می شوند.
ClearChildState	اطلاعات view-state و control-state تمامی کنترل های فرزند server control را حذف می کند.
ClearChildViewState	اطلاعات view-state کلیه ی کنترل های فرزند server control را پاک می کند.
CreateChildControls	از این تابع به منظور ایجاد کنترل های فرزند استفاده می شود.
CreateControlCollection	یک شیء ControlCollection جدید ایجاد می کند که کنترل های فرزند را در خود نگه می دارد (دربرمی گیرد).
CreateControlStyle	این متد یک شیء style ایجاد می کند که به منظور پیاده سازی تمامی خواص مربوط به style بکار گرفته می شود.
DataBind	یک منبع داده (data source) را به کنترل سرور و تمامی کنترل های فرزند آن متصل می کند.
DataBind( Boolean)	یک منبع داده (data source) را به کنترل سرور و کلیه ی کنترل های فرزند آن البته به شرط اینکه که DataBinding event اجرا شود

	متصل می کند.
DataBindChildren	منبع داده (data source) را به کنترل های فرزند server control متصل می کند.
Dispose	این متد را زمانی باید فراخوانی کنید که دیگر به control نیازی ندارید. این متد در واقع control را در وضعیت غیر قابل استفاده ( unusable state ) رها می کند. پس از صدا زدن این متد باید تمامی ارجاعات به کنترل مربوط را آزاد کنید تا از این طریق حافظه ی اشغال شده توسط زباله روب (garbage collection) بازگرفته و احیاء شود.
EnsureChildControls	مشخص می کند آیا سرور کنترل، کنترل فرزند دربرگیرد یا خیر. این متد اول مقدار خاصیت ChildControlsCreated را چک می کند، چنانچه این مقدار false بود در آن صورت متد CreateChildControls صدا زده می شود.
EnsureID	این متد شناسه ای برای کنترل هایی که دارای identifier نیستند ایجاد می کند.
Equals(Object)	بررسی می کند آیا شیء مشخص شده با شیء جاری یکسان و برابر است یا خیر.
Finalize	به یک شیء امکان می دهد منابع (resource) را آزاد ساخته و به دیگر عملیات پاک سازی (cleanup) پیش از اینکه شیء مذکور توسط garbage collection بازگرفته (احیاء) شود، بپردازد.
FindControl(String)	این تابع به منظور دسترسی به کنترلی که شناسه ی آن در زمان design مشخص نیست استفاده می شود.
FindControl(String, Int32)	بسیاری از اوقات id کنترل ها از قبل مشخص نیست و در زمان اجرا id ها مشخص می شوند مثال زمانی که دکمه ها در grid قرار می گیرند. با استفاده از این متد می توان به این کنترل ها دسترسی داشت.
Focus	Focus را برای یک کنترل انجام می دهد. Focus فرایندی است که در آن یک کنترل انتخاب می شود (keyboard cursor) روی آن قرار می

	گیرد.
GetDesignModeState	اطلاعات زمان طراحی یک کنترل را بازیابی می کند.
GetType	نوع نمونه (instance) جاری را برمی گرداند.
HasControls	مشخص می کند آیا کنترل سرور دربردارنده ی کنترل های فرزند ( child control ) هست یا خیر.
HasEvents	مشخص می کند آیا رخدادهایی ویژه ی کنترل مورد نظر یا کنترل های فرزند (child control) ثبت (register) می شوند یا خیر.
IsLiteralContent	مشخص می کند آیا کنترل سرور فقط دربردارنده ی literal content ( محتویات متنی) هست یا خیر.
LoadControlState	اطلاعات control-state را بازیابی می کند.
LoadViewState	اطلاعات view-state را بازیابی می کند.
MapPathSecure	مسیر فیزیکی (physical path) که یک مسیر مجازی (virtual) به آن نگاشت می شود را بازیابی می کند.
MemberwiseClone	یک shallow copy از شیء جاری ایجاد می کند. : Shallow copy در فرایند shallow copying، B تمامی مقادیر فیلد (field value) A را کپی می کند. اگر مقدار فیلد یک آدرس حافظه باشد، تنها آدرس حافظه را دانلود می کند و اگر مقدار فیلد یک نوع اولیه ( primitive type) باشد در آن صورت مقدار نوع اولیه را کپی می کند.
OnBubbleEvent	تعیین می کند آیا رخداد کنترل سرور به سمت parent control (کنترل والد یا دربرگیرنده) صفحه فرستاده شود یا خیر.
OnDataBinding	رخداد DataBinding را فعال می کند.
OnInit	رخداد Init را فعال می کند.

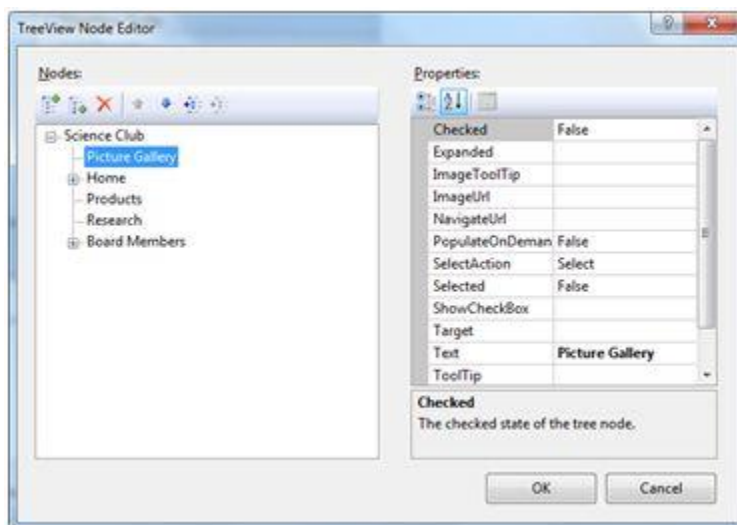
OnLoad	رخداد Load را فعال می کند.
OnPreRender	رخداد PreRender را فعال می کند.
OnUnload	رخداد Unload را اجرا می کند.
OpenFile	این متد یک شیء stream برمی گرداند که می توان از آن ها برای خواندن محتوا (content) فایل مشخص شده در پارامتر <i>path</i> استفاده کرد.
RemovedControl	این تابع پس از اینکه کنترل فرزند از مجموعه کنترل های ( control collection) حذف گردد صدا زده می شود.
Render	محتویات کنترل را به یک writer اعمال می نماید.
RenderBeginTag	تگ آغازین حاصله از render شدن کنترل را به writer ارسال می نماید.
RenderChildren	محتویات کنترل های فرزند (child control) از یک کنترل را به writer انتقال می دهد.
RenderContents	محتویات یک کنترل را به یک writer اعمال می نماید.
RenderControl(HtmlTextWriter)	خروجی (نتیجه ی render شدن) یک کنترل را به html text writer ارسال می کند.
RenderEndTag	در کنترل writer (به عنوان مثال html text writer) تگ انتهایی یک تگ را درج می نماید.
SaveControlState	کلید ی تغییرات اعمال شده در control state را در زمان post back بازیابی می نماید.
SaveViewState	تغییرات view state را که بعد از آخرین فراخوانی متد TrackViewState اعمال شده است را ذخیره می نماید.
SetDesignModeState	تنظیم کننده ی اطلاعات زمان طراحی برای control می باشد.

ToString	شئ جاری را به صورت رشته برمی گرداند.
TrackViewState	با true نمودن این خصیصه تغییرات property های کنترل در view state اعمال می شوند.

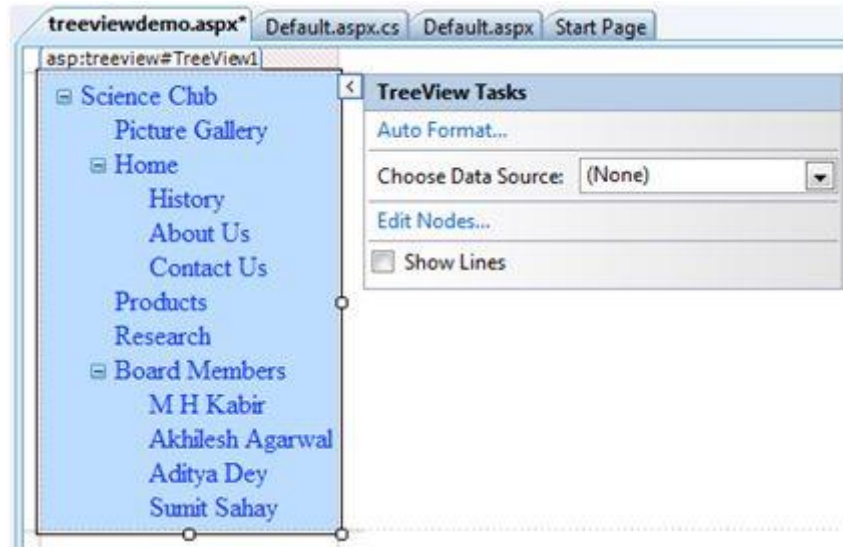
مثال :

اکنون نگاهی به تنها یکی از کنترل **server-side** فوق می اندازیم که یک کنترل نمایش درختی (**control tree view**) می باشد. یک کنترل **tree view** در واقع زیر کنترل های پیمایش (**navigation control**) نمایش داده می شود. کنترل های پیمایش عبارتند از: کنترل **Menu** و کنترل **SiteMapPath**.

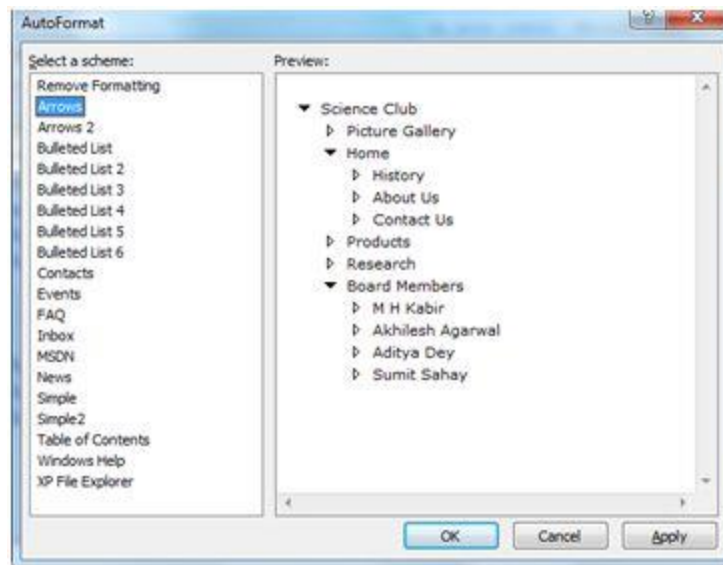
یک کنترل **tree view** در صفحه ایجاد کنید. اکنون... **Edit Nodes** را از میان **task** ها انتخاب کنید. هر یک از گره ها (**node**) را همان گونه که تصویر زیر نمایش می دهد با استفاده از **Tree view node editor** ویرایش کنید.



پس از ایجاد گره ها (**node**)، کنترل نمایش درختی (در **design view**) به صورت زیر در خواهد آمد:



tree view به شما امکان می دهد تصویر زیر نمایش می دهد قالب بندی (format دهی) کنید:



یک **label control** و **textbox control** به صفحه اضافه کرده و آن ها را به ترتیب **lblmessage** و **txtmessage** نام گذاری کنید.

اکنون کدهای لازم را می نویسید تا زمانی که روی یک گره ی (**node**) معین کلیک شد، کنترل **label** متن **node** را نشان دهد و **text box** نیز تمامی گره های فرزند (**child node**) را در خود به نمایش بگذارد.

: **Code behind**

```

using System;
using System.Collections;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Linq;
namespace ServerControl
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            txtmessage.Text = " ";
        }

        protected void TreeView1_SelectedNodeChanged(object sender, EventArgs e)
        {
            txtmessage.Text = " ";
            lblmessage.Text = "Selected node changed to: " + TreeView1.SelectedNode.Text;
            TreeNodeCollection childnodes = TreeView1.SelectedNode.ChildNodes;

            if (childnodes != null)
            {
                txtmessage.Text = " ";

                foreach (TreeNode t in childnodes)
                {
                    txtmessage.Text += t.Value;
                }
            }
        }
    }
}

```

صفحه را اجرا کرده تا نتیجه را مشاهده کنید. حال خواهی دید که باز کردن (بسط دادن) گره ها و بستن (جمع کردن)

آن ها امکان پذیر می باشد.

- ▼ Science Club
  - ▷ Picture Gallery
  - ▼ Home
    - ▷ History
    - ▷ About Us
    - ▷ Contact Us
  - ▷ Products
  - ▷ Research
  - ▼ Board Members
    - ▷ M H Kabir
    - ▷ Akhilesh Agarwal
    - ▷ Aditya Dey
    - ▷ Sumit Sahay

Selected node changed to: Board Members

M H KabirAkhilesh AgarwalAditya