.NET Framework 4 - ASP.NET

**ASP.NET Overview**

ASP.NET is a unified Web development model that includes the services necessary for you to build enterprise-class Web applications with a minimum of coding. ASP.NET is part of the .NET Framework, and when coding ASP.NET applications you have access to classes in the .NET Framework. You can code your applications in any language compatible with the common language runtime (CLR), including Microsoft Visual Basic and C#. These languages enable you to develop ASP.NET applications that benefit from the common language runtime, type safety, inheritance, and so on.

|  |
| --- |
| **NoteNote** |
| If you want to try ASP.NET, you can install Visual Web Developer Express using the [Microsoft Web Platform Installer](http://go.microsoft.com/fwlink/?LinkId=153787), which is a free tool that makes it simple to download, install, and service components of the Microsoft Web Platform. These components include Visual Web Developer Express, Internet Information Services (IIS), SQL Server Express, and the .NET Framework. All of these are tools that you use to create ASP.NET Web applications. You can also use the Microsoft Web Platform Installer to install open-source ASP.NET and PHP Web applications. |

This topic describes the following features of ASP.NET and of Visual Web Developer, the development environment for creating ASP.NET applications.

* [Visual Web Developer](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#VisualWebDeveloper)
* [ASP.NET Web Sites and ASP.NET Web Application Projects](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#ASPNETWebSitesAndASPNETWebApplicationProjects)
* [ASP.NET API Reference](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#aspnet_api_reference)
* [Page and Controls Framework](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#PageAndControlsFramework)
* [ASP.NET Compiler](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#ASPNETCompiler)
* [Security Infrastructure](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#SecurityInfrastructure)
* [State-Management Facilities](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#StateManagementFacilities)
* [ASP.NET Configuration](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#ASPNETConfiguration)
* [Health Monitoring and Performance Features](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#HealthMonitoringAndPerformanceFeatures)
* [Debugging Support](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#DebuggingSupport)
* [Web Services Framework](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#XMLWebServicesFramework)
* [Extensible Hosting Environment and Application Life-Cycle Management](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#ExtensibleHostingEnvironment)
* [Extensible Designer Environment](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#ExtensibleDesignerEnvironment)
* [Web Applications Based on the MVC Pattern](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#MVC)
* [ASP.NET Dynamic Data](http://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#ASPNETDynamicData)

http://i.msdn.microsoft.com/Global/Images/clear.gif Visual Web Developer

Visual Web Developer is a full-featured development environment for creating ASP.NET Web applications. Visual Web Developer offers you the following features:

* **Web page design**   A powerful Web page editor that includes WYSIWYG editing and an HTML editing mode with IntelliSense and validation.
* **Page design features**   Consistent site layout with master pages and consistent page appearance with themes and skins.
* **Code editing**   A code editor that enables you to write code for your dynamic Web pages in Visual Basic or C#. The code editor includes syntax coloration and IntelliSense.
* **Testing and Debugging**   A local Web server for testing and a debugger that helps you find errors in your programs.
* **Deployment**   Tools to automate typical tasks for deploying a Web application to a hosting server or a hosting provider.

For more information, see [Visual Web Developer Content Map](http://msdn.microsoft.com/en-us/library/y9d53c8d.aspx).

**Testing and Debugging**

Visual Web Developer provides an ideal environment in which to build Web sites and then publish them to a hosting site. Using the development tools in Visual Web Developer, you can develop ASP.NET Web pages on your own computer. Visual Web Developer includes a local Web server that provides all the features you need to test and debug ASP.NET Web pages, without requiring Internet Information Services (IIS) to be installed.

When your site is ready, you can publish it to the host computer using the built-in Copy Web tool, which transfers your files when you are ready to share them with others. Alternatively, you can precompile and deploy a Web site by using the **Build Web Site** command. The **Build Web Site** command runs the compiler over the entire Web site (not just the code files) and produces a Web site layout that you can deploy to a production server.

|  |
| --- |
| **NoteNote** |
| The Build Web Site feature is not available in Visual Web Developer Express Edition. |

Finally, you can take advantage of the built-in support for File Transfer Protocol (FTP). Using the FTP capabilities of Visual Web Developer, you can connect directly to the host computer and then create and edit files on the server.

http://i.msdn.microsoft.com/Global/Images/clear.gif ASP.NET Web Sites and ASP.NET Web Application Projects

Using Visual Studio 2010, you can create different types of ASP.NET projects, which includes Web sites, Web applications, Web services, and AJAX server controls.

There is a difference between Web site projects and Web application projects. Some features work only with Web application projects, such as MVC and certain tools for automating Web deployment. Other features, such as Dynamic Data, work with both Web sites and Web application projects. For more information about the differences between Web application projects and Web site projects, see [Web Application Projects versus Web Site Projects](http://msdn.microsoft.com/en-us/library/dd547590.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif ASP.NET API Reference

Some of the most important namespaces in the .NET Framework class library that pertain to ASP.NET are the following:

[System.Web](http://msdn.microsoft.com/en-us/library/system.web.aspx)

Provides classes and interfaces that enable browser-server communication. This namespace includes the [HttpRequest](http://msdn.microsoft.com/en-us/library/system.web.httprequest.aspx) class, which provides extensive information about the current HTTP request, the [HttpResponse](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.aspx) class, which manages HTTP output to the client, and the [HttpServerUtility](http://msdn.microsoft.com/en-us/library/system.web.httpserverutility.aspx) class, which provides access to server-side utilities and processes. [System.Web](http://msdn.microsoft.com/en-us/library/system.web.aspx) also includes classes for cookie manipulation, file transfer, exception information, and output cache control.

[System.Web.ApplicationServices](http://msdn.microsoft.com/en-us/library/system.web.applicationservices.aspx)

Provides classes that provide access to ASP.NET forms authentication, roles, and profiles application services as Windows Communication Foundation (WCF) services.

[System.Runtime.Caching](http://msdn.microsoft.com/en-us/library/system.runtime.caching.aspx)

Contains types that let you implement caching in .NET Framework applications.

[System.Web.ClientServices](http://msdn.microsoft.com/en-us/library/system.web.clientservices.aspx)

Contains classes that support access to the ASP.NET login, roles, and profiles services from Windows-based applications.

[System.Web.Configuration](http://msdn.microsoft.com/en-us/library/system.web.configuration.aspx)

Contains classes that are used to programmatically manage ASP.NET configuration. (Most configuration settings can be made in XML files.)

[System.Web.DynamicData](http://msdn.microsoft.com/en-us/library/system.web.dynamicdata.aspx)

Contains classes that provide the core functionality for ASP.NET dynamic data and extensibility features that let you customize dynamic data behavior.

[System.Web.Handlers](http://msdn.microsoft.com/en-us/library/system.web.handlers.aspx)

Contains HTTP handler classes that process HTTP requests to a Web server. (An ASP.NET Web Forms page -- .aspx file -- is a special form of an HTTP handler.)

[System.Web.Management](http://msdn.microsoft.com/en-us/library/system.web.management.aspx)

Contains classes and interfaces for managing and monitoring the health of Web applications.

[System.Web.Profile](http://msdn.microsoft.com/en-us/library/system.web.profile.aspx)

Contains classes that are used to implement the ASP.NET user profile in Web server applications.

[System.Web.Query.Dynamic](http://msdn.microsoft.com/en-us/library/system.web.query.dynamic.aspx)

Contains classes that are used to parse expressions from a [LinqDataSource](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.linqdatasource.aspx) control into a Language-Integrated Query (LINQ).

[System.Web.RegularExpressions](http://msdn.microsoft.com/en-us/library/system.web.regularexpressions.aspx)

Provides regular expressions that are used to parse ASP.NET files. All members of the [System.Web.RegularExpressions](http://msdn.microsoft.com/en-us/library/system.web.regularexpressions.aspx) namespace are descendants of the [Regex](http://msdn.microsoft.com/en-us/library/system.text.regularexpressions.regex.aspx) class. (You typically do not have to parse ASP.NET pages yourself.)

[System.Web.Routing](http://msdn.microsoft.com/en-us/library/system.web.routing.aspx)

Provides classes that are used with URL routing, which enables you to use URLs that do not map to a physical file.

[System.Web.Script](http://msdn.microsoft.com/en-us/library/system.web.script.aspx)

Contains classes that provide client-script resource information.

[System.Web.Script.Services](http://msdn.microsoft.com/en-us/library/system.web.script.services.aspx)

Provides attributes to customize Web service support for using Ajax functionality in ASP.NET.

[System.Web.Security](http://msdn.microsoft.com/en-us/library/system.web.security.aspx)

Contains classes that are used to implement ASP.NET security in Web server applications.

[System.Web.Services](http://msdn.microsoft.com/en-us/library/system.web.services.aspx)

Consists of the classes that enable you to create XML Web services using ASP.NET and XML Web service clients. XML Web services are applications that provide the ability to exchange messages in a loosely coupled environment using standard protocols such as HTTP, XML, XSD, SOAP, and WSDL. XML Web services let you build modular applications that are interoperable across a broad variety of implementations, platforms, and devices.

[System.Web.SessionState](http://msdn.microsoft.com/en-us/library/system.web.sessionstate.aspx)

Contains classes and interfaces that enable storage of data specific to a single client during a single browser session on the server. Session state data is used to give the client the appearance of a persistent connection with the application.

[System.Web.UI](http://msdn.microsoft.com/en-us/library/system.web.ui.aspx)

Provides classes and interfaces that enable you to create ASP.NET server controls and ASP.NET Web pages for the user interface of your ASP.NET Web applications. This namespace includes the [Control](http://msdn.microsoft.com/en-us/library/system.web.ui.control.aspx) class, which provides all HTML server controls, Web server controls, and user controls with a common set of functionality. It also includes the [Page](http://msdn.microsoft.com/en-us/library/system.web.ui.page.aspx) control, which is generated automatically whenever a request is made for an .aspx file in an ASP.NET Web application. Also included are classes which provide the server controls with data-binding functionality, the ability to save the view state of a given control or page, and parsing functionality.

[System.Web.UI.DataVisualization.Charting](http://msdn.microsoft.com/en-us/library/system.web.ui.datavisualization.charting.aspx)

Contains types for the **Chart** Web server control.

[System.Web.UI.Design.WebControls](http://msdn.microsoft.com/en-us/library/system.web.ui.design.webcontrols.aspx)

Contains classes that can be used to extend design-time support for Web server controls.

[System.Web.UI.Design.WebControls.WebParts](http://msdn.microsoft.com/en-us/library/system.web.ui.design.webcontrols.webparts.aspx)

Contains classes that provide design-time support for controls derived from classes in the [System.Web.UI.WebControls.WebParts](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.webparts.aspx) namespace.

[System.Web.UI.HtmlControls](http://msdn.microsoft.com/en-us/library/system.web.ui.htmlcontrols.aspx)

Contains a collection of classes that enable you to create HTML server controls on a Web Forms page. HTML server controls run on the server and map directly to standard HTML tags supported by most browsers. This enables you to programmatically control the HTML elements on a Web Forms page.

[System.Web.UI.WebControls](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.aspx)

Contains classes that enable you to create Web server controls on a Web page. Web server controls run on the server and include form controls such as buttons and text boxes. They also include special-purpose controls such as a calendar. Because Web server controls run on the server, you can programmatically control these elements. Web server controls are more abstract than HTML server controls. Their object model does not necessarily reflect HTML syntax.

[System.Web.UI.WebControls.WebParts](http://msdn.microsoft.com/en-us/library/system.web.ui.webcontrols.webparts.aspx)

Contains an integrated set of classes and interfaces for creating Web pages whose appearance and behavior can be modified (personalized) by end users. The user-defined settings for each page are saved for future browser sessions.

[System.Web.Util](http://msdn.microsoft.com/en-us/library/system.web.util.aspx)

Contains classes that enable callback methods to be run under the scope of a transaction and that enable work to be posted to separate threads.

[ASP.NET MVC Reference](http://msdn.microsoft.com/en-us/library/dd566232.aspx)

This topic provides links to four namespaces that are used by the MVC framework.

For a complete list of .NET Framework namespaces, with links to API reference topics for them, see [.NET Framework Class Library](http://msdn.microsoft.com/en-us/library/ms229335.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Page and Controls Framework

The ASP.NET page and controls framework is a programming framework that runs on a Web server to dynamically produce and render ASP.NET Web pages. ASP.NET Web pages can be requested from any browser or client device, and ASP.NET renders markup (such as HTML) to the requesting browser. As a rule, you can use the same page for multiple browsers, because ASP.NET renders the appropriate markup for the browser making the request. However, you can design your ASP.NET Web page to target a specific browser and take advantage of the features of that browser.

ASP.NET Web pages are completely object-oriented. Within ASP.NET Web pages you can work with HTML elements using properties, methods, and events. The ASP.NET page framework removes the implementation details of the separation of client and server inherent in Web-based applications by presenting a unified model for responding to client events in code that runs at the server. The framework also automatically maintains the state of a page and the controls on that page during the page processing life cycle. For more information see [ASP.NET Web Pages Overview](http://msdn.microsoft.com/en-us/library/428509ah.aspx).

The ASP.NET page and controls framework also enables you to encapsulate common UI functionality in easy-to-use, reusable controls. Controls are written once, can be used in many pages, and are integrated into the ASP.NET Web page that they are placed in during rendering.

The ASP.NET page and controls framework also provides features to control the overall look and feel of your Web site via themes and skins. You can define themes and skins and then apply them at a page level or at a control level. For more information, see [ASP.NET Themes and Skins](http://msdn.microsoft.com/en-us/library/ykzx33wh.aspx).

In addition to themes, you can define master pages that you use to create a consistent layout for the pages in your application. A single master page defines the layout and standard behavior that you want for all the pages (or a group of pages) in your application. You can then create individual content pages that contain the page-specific content you want to display. When users request the content pages, they merge with the master page to produce output that combines the layout of the master page with the content from the content page. For more information see [ASP.NET Master Pages](http://msdn.microsoft.com/en-us/library/wtxbf3hh.aspx).

The ASP.NET page framework also enables you to define the pattern for URLs that will be used in your site. This helps with search engine optimization (SEO) and makes URLs more user-friendly. For more information, see [ASP.NET Routing](http://msdn.microsoft.com/en-us/library/cc668201.aspx).

The ASP.NET page and control framework is designed to generate HTML that conforms to accessibility guidelines. For more information, see [Accessibility in Visual Studio and ASP.NET](http://msdn.microsoft.com/en-us/library/ms228004.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif ASP.NET Compiler

All ASP.NET code is compiled, which enables strong typing, performance optimizations, and early binding, among other benefits. Once the code has been compiled, the common language runtime further compiles ASP.NET code to native code, providing improved performance.

ASP.NET includes a compiler that will compile all your application components including pages and controls into an assembly that the ASP.NET hosting environment can then use to service user requests. For more information, see [ASP.NET Compilation Overview](http://msdn.microsoft.com/en-us/library/ms178466.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Security Infrastructure

In addition to the security features of .NET, ASP.NET provides an advanced security infrastructure for authenticating and authorizing user access as well as performing other security-related tasks. You can authenticate users using Windows authentication supplied by IIS, or you can manage authentication using your own user database using ASP.NET forms authentication and ASP.NET membership. Additionally, you can manage the authorization to the capabilities and information of your Web application using Windows groups or your own custom role database using ASP.NET roles. You can easily remove, add to, or replace these schemes depending upon the needs of your application. For more information, see the following topics:

* [ASP.NET Security](http://msdn.microsoft.com/en-us/library/91f66yxt.aspx)
* [Managing Users by Using Membership](http://msdn.microsoft.com/en-us/library/tw292whz.aspx)
* [Managing Authorization Using Roles](http://msdn.microsoft.com/en-us/library/9ab2fxh0.aspx)
* [Forms Authentication Provider](http://msdn.microsoft.com/en-us/library/9wff0kyh.aspx)

ASP.NET always runs with a particular Windows identity so you can secure your application using Windows capabilities such as NTFS Access Control Lists (ACLs), database permissions, and so on. For more information about the identity of ASP.NET, see [Configuring ASP.NET Process Identity](http://msdn.microsoft.com/en-us/library/dwc1xthy.aspx) and [ASP.NET Impersonation](http://msdn.microsoft.com/en-us/library/xh507fc5.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif State-Management Facilities

ASP.NET provides intrinsic state management functionality that enables you to store information between page requests, such as customer information or the contents of a shopping cart. You can save and manage application-specific, session-specific, page-specific, user-specific, and developer-defined information. This information can be independent of any controls on the page.

ASP.NET offers distributed state facilities, which enable you to manage state information across multiple instances of the same application on one computer or on several computers. For more information, see [ASP.NET State Management Overview](http://msdn.microsoft.com/en-us/library/75x4ha6s.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif ASP.NET Configuration

ASP.NET applications use a configuration system that enables you to define configuration settings for your Web server, for a Web site, or for individual applications. You can make configuration settings at the time your ASP.NET applications are deployed and can add or revise configuration settings at any time with minimal effect on operational Web applications and servers. ASP.NET configuration settings are stored in XML-based files. Because these XML files are text files, it is simple to make configuration changes to your Web applications. You can extend the configuration scheme to suit your requirements. For more information, see [ASP.NET Configuration Overview](http://msdn.microsoft.com/en-us/library/ms178683.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Health Monitoring and Performance Features

ASP.NET includes features that enable you to monitor health and performance of your ASP.NET application. ASP.NET health monitoring enables reporting of key events that provide information about the health of an application and about error conditions. These events show a combination of diagnostics and monitoring characteristics and offer a high degree of flexibility in terms of what is logged and how it is logged. For more information, see [ASP.NET Health Monitoring Overview](http://msdn.microsoft.com/en-us/library/bb398933.aspx).

ASP.NET supports two groups of performance counters available to your applications:

* The ASP.NET system performance counter group
* The ASP.NET application performance counter group

For more information, see [Monitoring ASP.NET Application Performance](http://msdn.microsoft.com/en-us/library/3xxk09t8.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Debugging Support

ASP.NET takes advantage of the run-time debugging infrastructure to provide cross-language and cross-computer debugging support. You can debug both managed and unmanaged objects, in addition to all languages supported by the common language runtime and script languages. For details, see [ASP.NET Debugging](http://msdn.microsoft.com/en-us/library/ms228091.aspx).

In addition, the ASP.NET page framework provides a trace mode that enables you to insert instrumentation messages into your ASP.NET Web pages. For more information, see [ASP.NET Tracing Overview](http://msdn.microsoft.com/en-us/library/bb386420.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Web Services Framework

ASP.NET supports XML Web services. An XML Web service is a component that contains business functionality that enables applications to exchange information across firewalls by using standards like HTTP and XML messaging. XML Web services are not tied to a particular component technology or object-calling convention. As a result, programs written in any language, using any component model, and running on any operating system can access XML Web services. For more information, see [ASP.NET Data Access](http://msdn.microsoft.com/en-us/library/6759sth4.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Extensible Hosting Environment and Application Life-Cycle Management

ASP.NET includes an extensible hosting environment that controls the life cycle of an application from when a user first accesses a resource (such as a page) in the application to the point at which the application is shut down. While ASP.NET relies on a Web server (IIS) as an application host, ASP.NET provides much of the hosting functionality itself. The architecture of ASP.NET enables you to respond to application events and create custom HTTP handlers and HTTP modules. For more information, see [ASP.NET Application Life Cycle Overview for IIS 5.0 and 6.0](http://msdn.microsoft.com/en-us/library/ms178473.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Extensible Designer Environment

ASP.NET includes enhanced support for creating designers for Web server controls for use with a visual design tool such as Visual Studio. Designers enable you to build a design-time user interface for a control, so that developers can configure your control's properties and content in the visual design tool. For more information, see [ASP.NET Control Designers Overview](http://msdn.microsoft.com/en-us/library/wxh45wzs.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Web Applications Based on the MVC Pattern

ASP.NET MVC helps Web developers build standards-based Web applications that are easy to maintain because it decreases the dependency among application layers by using the Model-View-Controller (MVC) pattern. It also provides complete control over the page markup. ASP.NET MVC improves testability by supporting Test Driven Development (TDD). For more information, see [ASP.NET MVC](http://go.microsoft.com/fwlink/?LinkId=216782).

http://i.msdn.microsoft.com/Global/Images/clear.gif ASP.NET Dynamic Data

ASP.NET Dynamic Data is a framework that lets you create data-driven ASP.NET Web applications easily. It does this by automatically discovering data-model metadata at run time and deriving UI behavior from it. A scaffolding framework provides a functional Web site for viewing and editing data. You can easily customize the scaffolding framework by changing elements or creating new ones to override the default behavior. Existing applications can easily integrate scaffolding elements together with ASP.NET pages.

You will find that you can get applications up and running more easily and with less code than before. At the same time, you can add custom features to ASP.NET Dynamic Data to accommodate your own requirements. For more information, see [ASP.NET Dynamic Data Overview](http://msdn.microsoft.com/en-us/library/ee845452.aspx).