

# Como crear una aplicación web con Visual Studio

---

En esta pequeño tutorial, crearemos nuestro primer proyecto web y agregaremos una nueva página a él.

Para este ejemplo crearemos un sitio web usando un sistema de archivos porque de esta forma no necesitaremos trabajar con Microsoft Internet Information Services (IIS). En lugar de esto, se creará y ejecutará la página en el sistema de archivos local.

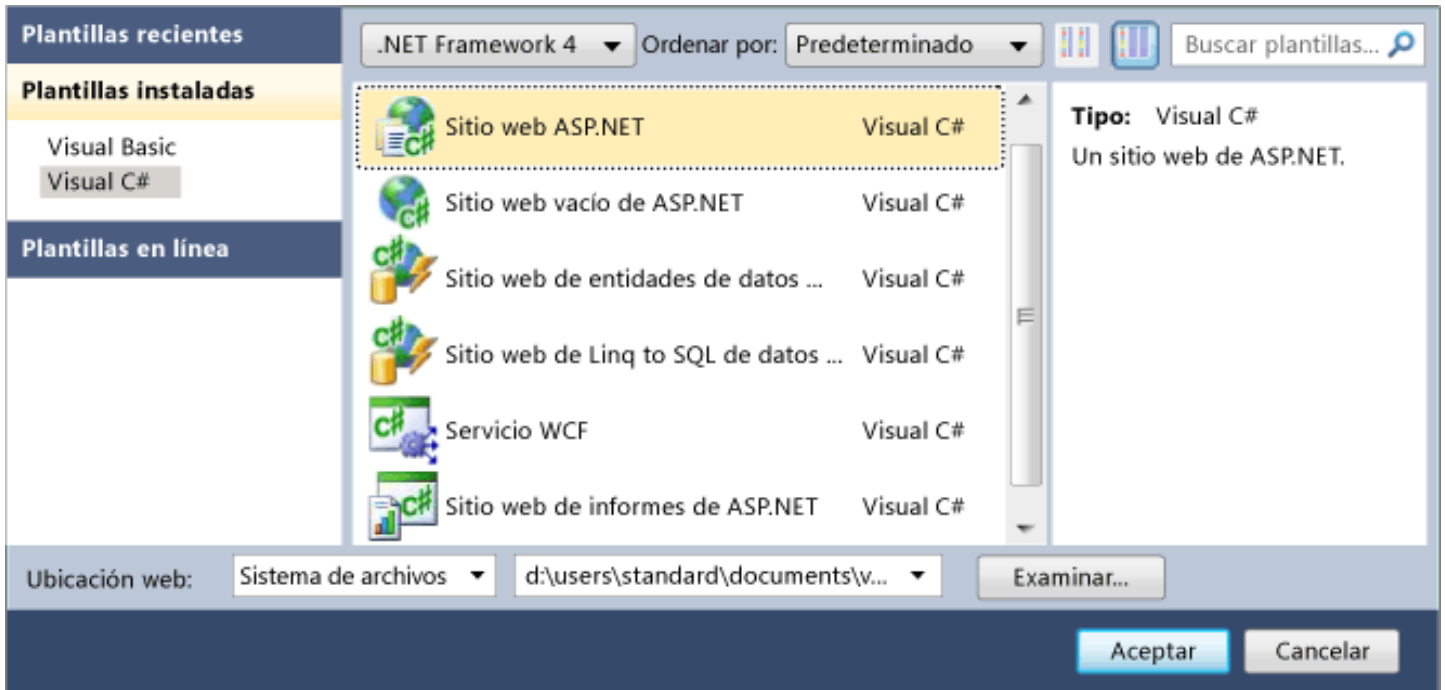
Un proyecto de sitio web de sistema de archivos es uno que almacena páginas y otros archivos en una carpeta que elige en alguna parte en su equipo. Otras opciones de proyecto Web incluyen el siguiente:

- Un proyecto de sitio web de IIS local, que almacena sus archivos en una subcarpeta de la raíz IIS local (normalmente, `\inetpub\wwwroot\`).
- Un proyecto de sitio FTP, que almacena los archivos en un servidor remoto que obtiene a través de Internet acceso a utilizando Protocolo de transferencia de archivos (FTP).
- Un proyecto de sitio web de IIS remoto, que almacena los archivos en un servidor remoto al que puede tener acceso por una red local.
- Un proyecto de aplicación web, que es similar a un proyecto de sitio web de sistema de archivos excepto usted lo compila antes de la implementación y lo implementa como un archivo de biblioteca de vínculos dinámicos (.dll). Para obtener más información sobre la diferencia entre el sitio web proyecta y proyectos de aplicación web, vea [Proyectos de aplicación web frente a proyectos de sitio web](#).

## Para crear un sitio web del sistema de archivos

1. Abra Microsoft Visual Studio 2010.
2. En el menú Archivo, haga clic en Nuevo sitio Web.

El cuadro de diálogo Nuevo sitio web aparece, como se muestra en la siguiente ilustración:



3. En Plantillas instaladas, haga clic en Visual Basic o C# y, a continuación, seleccione Sitio web ASP.NET.

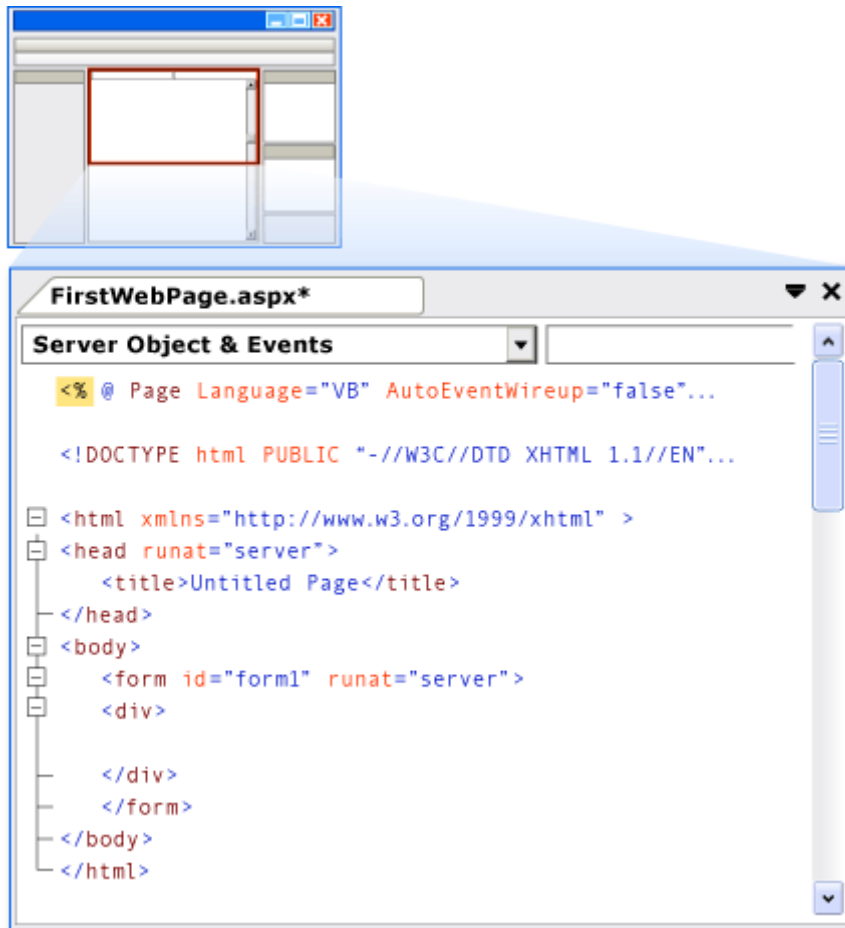
Al crear un proyecto de sitio web, especifica una plantilla. Cada plantilla crea un proyecto Web que contiene archivos diferentes y carpetas. En este tutorial, está creando un sitio web basado en la plantilla Sitio web ASP.NET, que crea los archivos y carpetas que se utilizan normalmente en sitios web de ASP.NET.

4. En el cuadro Ubicación Web (Web Location), seleccione Sistema de archivos y a continuación, escriba el nombre de la carpeta donde desea mantener las páginas de su sitio web.

Por ejemplo, escriba el nombre de carpeta C:\BasicWebSite.

5. Haga clic en Aceptar.

Visual Studio crea un proyecto Web que incluye la funcionalidad del prebuilt para el diseño (una página maestra, las páginas de contenido de About.aspx y Default.aspx y una hoja de estilos en cascada), para Ajax (archivos de script de cliente), y para la autenticación (pertenencia ASP.NET). Cuando se crea una nueva página, de forma predeterminada Visual Studio muestra la página en vista Código fuente, donde puede ver los elementos HTML de la página. En la siguiente ilustración se muestra la vista Código fuente de una página web predeterminada.



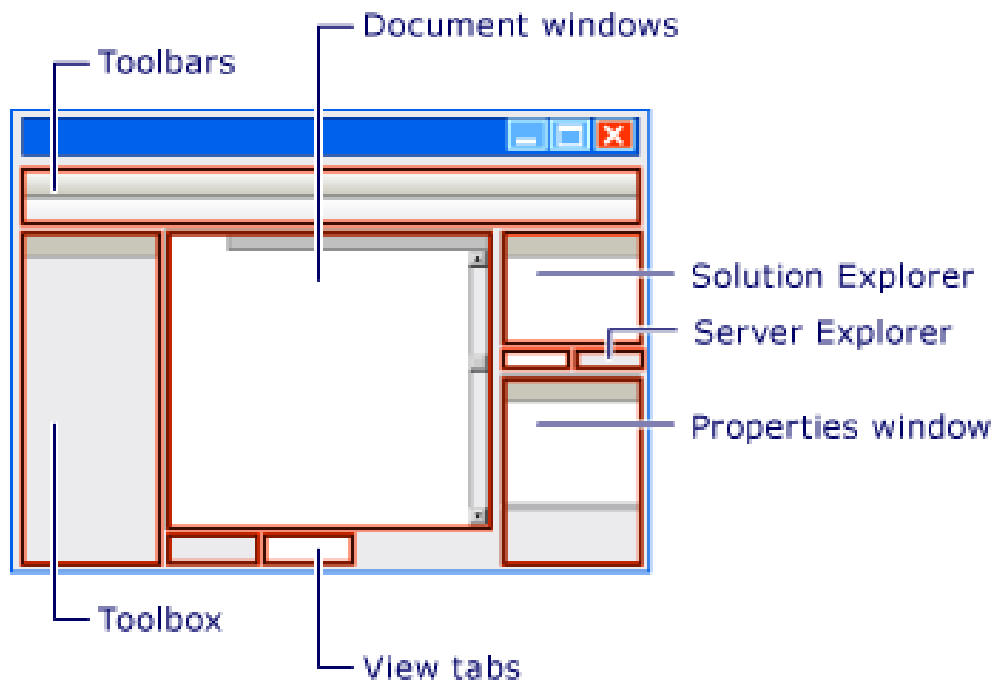
## Un paseo del Visual Studio Desarrollo web Entorno

Antes de continuar con trabajar en la página, es útil para familiarizarse con el entorno de desarrollo de Visual Studio. La siguiente ilustración le muestra las ventanas y herramientas que están disponibles en Visual Studio 2010.

### Nota

En este diagrama, se muestran las ventanas predeterminadas y las ubicaciones predeterminadas de las ventanas. El menú Ver permite mostrar ventanas adicionales así como reorganizar y cambiar el tamaño de las ventanas según las preferencias del usuario. Si los cambios ya se han realizado en la organización de la ventana, qué ve no coincidirá con la ilustración.

## El entorno de Visual Studio



### Para familiarizarse con el diseñador de Web

Examine la ilustración anterior y relacione el texto con la lista siguiente, que describe las ventanas y herramientas que se utilizan con más frecuencia. No todas las ventanas y herramientas visibles se muestran aquí, sólo las marcadas en la ilustración anterior.

- **Barras de herramientas (Toolbars):** Proporcionan los comandos que permiten dar formato al texto, buscar texto, etc. Algunas barras de herramientas sólo están disponibles cuando se trabaja en la vista Diseño.
- **Ventana Explorador de soluciones (Solution Explorer):** Muestra los archivos y carpetas del sitio Web.
- **Ventana Documento (Document Window):** Muestra los documentos en los que se está trabajando en ventanas con fichas. Puede pasar de un documento a otro haciendo clic en las fichas.
- **Ventana Propiedades (Properties Window):** Permite cambiar la configuración de la página, los elementos HTML, los controles y otros objetos.
- **Fichas de vistas (View tabs):** Presentan distintas vistas del mismo documento. La vista Diseño es un área de edición prácticamente WYSIWYG. La vista Código fuente es el editor HTML de la página. Divida las presentaciones de la vista la vista Diseño y la vista Código fuente para el documento. Trabajaré con las vistas Diseño y Código fuente más adelante en este tutorial. Si prefiere abrir las páginas web en la vista Diseño, en el menú Herramientas haga clic en Opciones, seleccione el nodo Diseñador HTML y cambie la opción Iniciar páginas en.
- **Cuadro de herramientas (Toolbox):** Proporciona los controles y elementos HTML que se pueden arrastrar a la página. Los elementos del Cuadro de herramientas se agrupan por funcionamiento común.

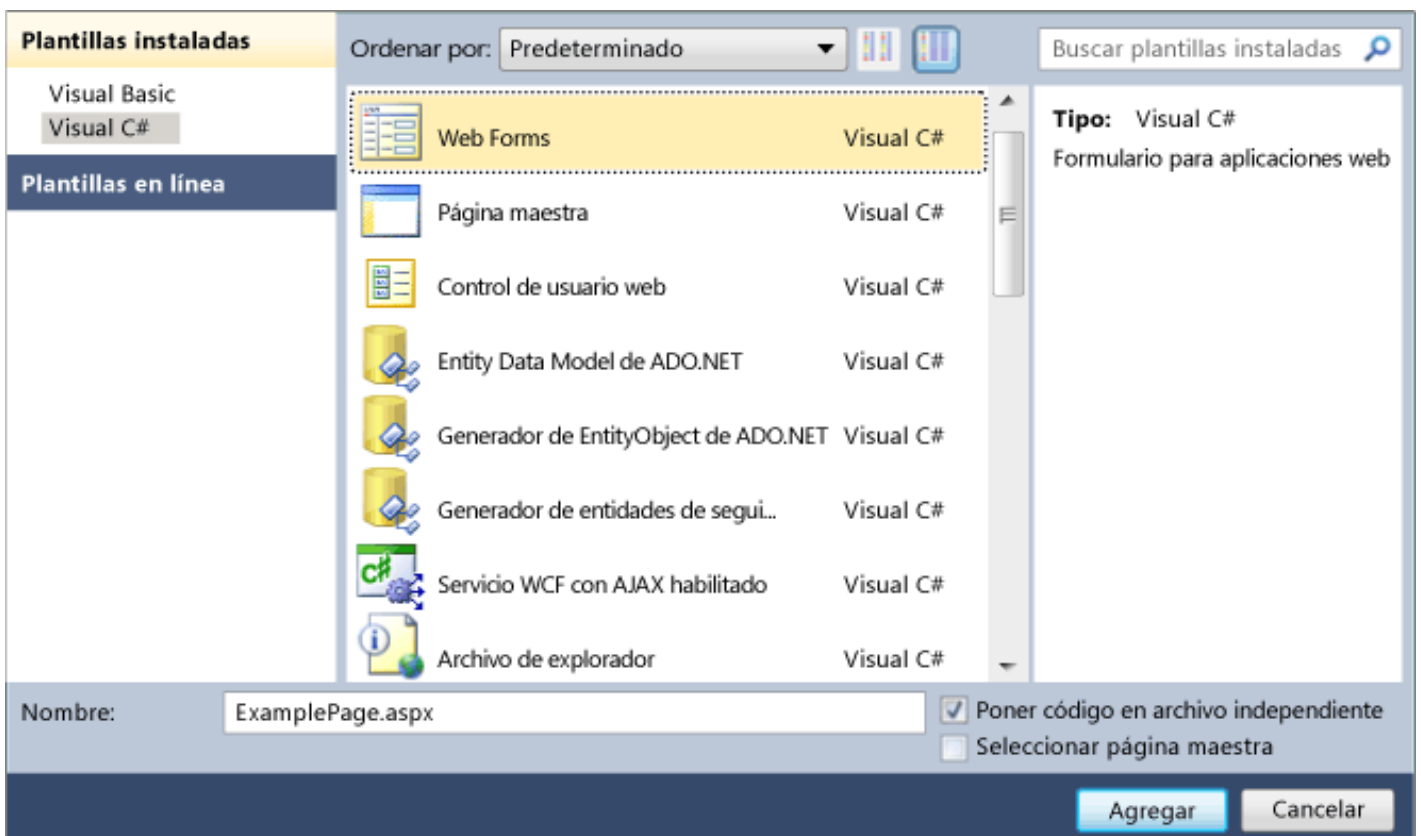
- **Explorador de servidores/Explorador de bases de datos (Server Explorer):** Muestra las conexiones de las bases de datos. Si Explorador de servidores no está visible, en el menú Ver, haga clic en Explorador de servidores o Explorador de bases de datos.

## Crear una Nueva Página de los formularios Web Forms ASP.NET

Al crear un nuevo sitio web mediante la plantilla de proyecto de Sitio web ASP.NET, Visual Studio agrega una página ASP.NET (página de formularios Web Forms) denominada Default.aspx. Puede utilizar la página Default.aspx como página principal de su sitio web. Sin embargo, en este tutorial, creará una nueva página y trabajará con ella.

### Para agregar una página al sitio Web

1. Cierre la página Default.aspx. Para ello, haga clic con el botón secundario en la pestaña que muestra el nombre de archivo y, a continuación, hace clic en Cerrar.
2. En Explorador de soluciones, haga clic con el botón secundario en el sitio web (por ejemplo, C:\BasicWebSite) y, a continuación, haga clic en Agregar nuevo elemento.
3. Se abrirá el cuadro de diálogo Agregar nuevo elemento. La siguiente ilustración muestra un ejemplo del cuadro de diálogo Agregar nuevo elemento.



4. En la lista de plantillas, seleccione Formulario Web Forms.
5. En el cuadro Nombre, escriba MiPrimeraPaginaWeb.
6. Cuando creó el proyecto de sitio web, especificó que un idioma predeterminado basó en la plantilla de proyecto que seleccionó. Sin embargo, cada vez que se crea una página o un componente nuevo para el sitio web, se puede seleccionar el lenguaje de programación para esa página o componente. Puede utilizar diferentes lenguajes de programación en el mismo sitio Web.
7. Haga clic en Agregar (Add).
8. Visual Studio crea la nueva página y la abre.

### **Agregar HTML a la página**

En esta parte del tutorial, agregará texto estático a la página.

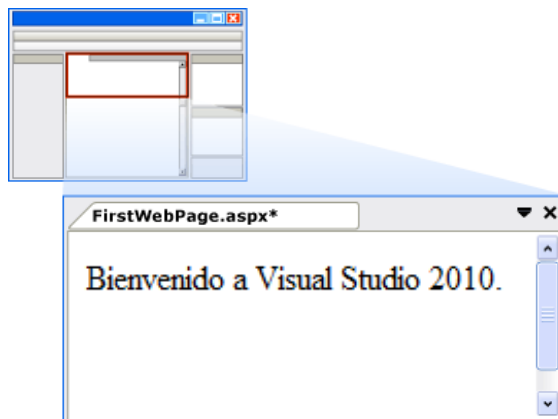
### **Para agregar texto a la página**

1. En la parte inferior de la ventana de documento, haga clic en la ficha Diseño para pasar a la vista Diseño.

La Vista de diseño muestra la página en la que se está trabajando en modo WYSIWYG. En este momento, no hay texto ni controles en la página, por lo que está en blanco, aparte de una línea discontinua que forma un rectángulo. Este rectángulo representa un elemento div en la página.

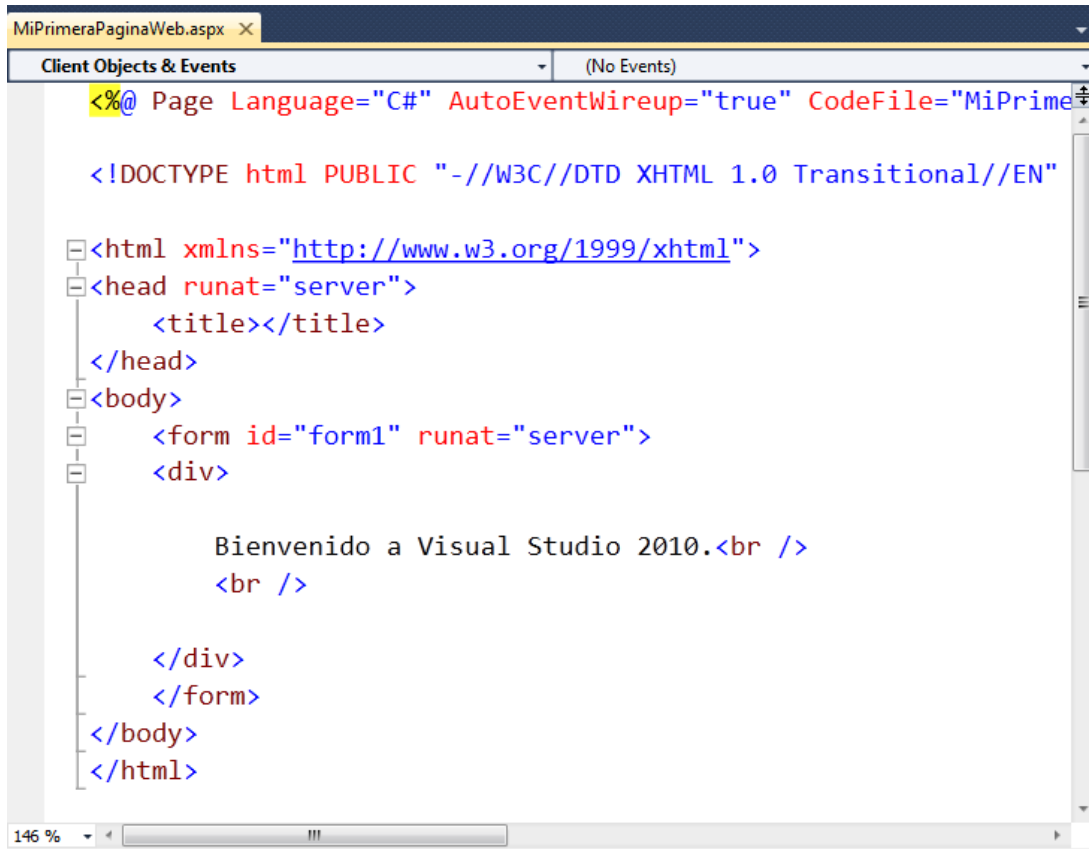
2. Haga clic en el interior del rectángulo formado por la línea discontinua.
3. Escriba Bienvenido a Visual Studio 2010 y presione ENTER dos veces.

La siguiente ilustración muestra el texto que escribió en la Vista de diseño.



#### 4. Cambie a la vista Código fuente.

Puede ver el código HTML que ha creado escribiendo en la Vista de diseño, como se muestra en la ilustración siguiente.



```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="MiPrime
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
</head>
<body>
  <form id="form1" runat="server">
  <div>
    Bienvenido a Visual Studio 2010.<br />
    <br />
  </div>
  </form>
</body>
</html>
```

### Ejecutar la página

Antes de continuar agregando controles a la página, puede intentar ejecutarla. Para ejecutar una página, necesita un servidor web. En un sitio Web de producción, se utiliza IIS como servidor web. Sin embargo, para probar una página, puede utilizar el Servidor de desarrollo de Visual Studio, que se ejecuta localmente y no requiere IIS. Para los proyectos de sitio web de sistema de archivos, el servidor web predeterminado en Visual Studio y Microsoft Visual Web Developer Express es el Servidor de desarrollo de Visual Studio.

### Para ejecutar la página

1. En el Explorador de soluciones, haga clic con el botón secundario del mouse en MiPrimeraPaginaWeb.aspx y seleccione Establecer como página principal.
2. Presione CTRL+F5 para ejecutar la página.

Visual Studio inicia el Servidor de desarrollo de Visual Studio. Un icono aparece en la barra de tareas de Windows para indicar que el Servidor de desarrollo de Visual Studio se está ejecutando, como se muestra en la ilustración.



3. La página se muestra en el explorador. Aunque la página que creó tiene una extensión de nombre de archivo de .aspx, ejecuta actualmente como cualquier página HTML.
4. Para mostrar una página en el explorador, también se puede hacer clic con el botón secundario del mouse en la página en el Explorador de soluciones y seleccionar Ver en el explorador.
5. Cierre el explorador.

## Agregar y programar controles

---

Ahora agregará controles de servidor a la página. Los controles de servidor, entre los que se incluyen botones, etiquetas, cuadros de texto y otros controles familiares, proporcionan las funciones típicas de procesamiento de formularios para las páginas Web ASP.NET. Sin embargo, puede programar los controles con código que se ejecuta en el servidor, no el cliente.

Agregará un control Button, un control TextBox y un control Label a la página y código de la escritura para administrar el evento Click para el control Button.

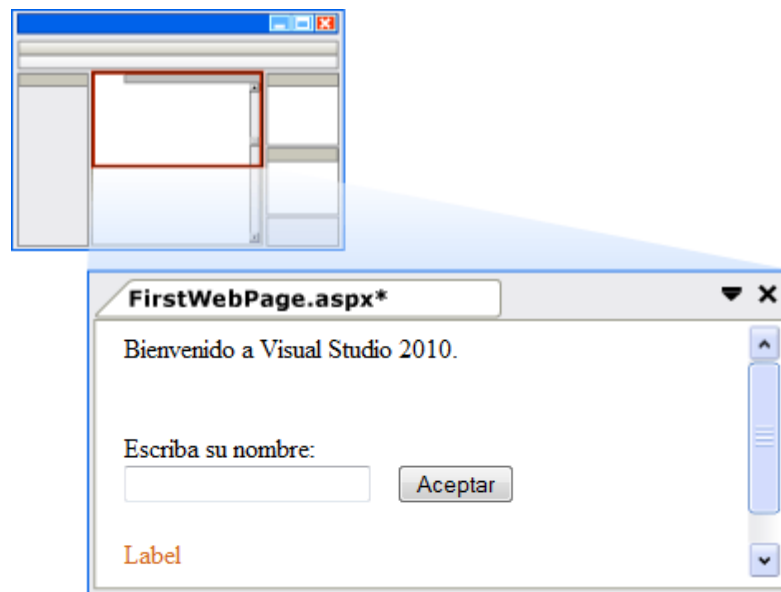
### Para agregar controles a la página

1. Haga clic en la pestaña Diseño para cambiar a la Vista de diseño.
2. Sitúe el punto de inserción al final del texto "Bienvenido a Visual Studio 2010" y presione ENTER al menos cinco veces para ampliar el espacio del cuadro del elemento div.
3. En el Cuadro de herramientas, expanda el grupo Estándar.
4. Arrastre un control TextBox a la página y colóquelo en el medio del cuadro de elemento div que tiene Inicio a Visual Web Developer en la primera línea.
5. Arrastre un control Button a la página y colóquelo al derecho del control TextBox.



6. Arrastre un control Label a la página y colóquelo en una línea independiente debajo del control Button, luego cambie la propiedad ID del label por lblNombre. Con este ID reconoceremos al control en la aplicación.
7. Sitúe el punto de inserción encima del control TextBox y, a continuación, escriba el texto "Escriba su nombre:".

Este texto HTML estático es el título del control TextBox. Puede mezclar HTML estático y controles de servidor en la misma página. La siguiente ilustración muestra cómo aparecen los tres controles en la Vista de diseño.

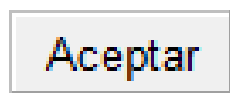


## Establecer las propiedades de los controles

Visual Studio le ofrece varias maneras de establecer las propiedades de controles en la página. En esta parte del tutorial, establecerá propiedades en las vistas de diseño y código fuente.

### Para establecer las propiedades de los controles

1. Seleccione el control Button, y a continuación en la ventana Propiedades, coloque la propiedad texto Agregar. El texto que escribió aparece en el botón en el diseñador, como se muestra en la siguiente ilustración.



#### Nota

**Si no se muestra la ventana Propiedades, presione F4 para mostrarlo.**

2. Cambie a la vista Código fuente.

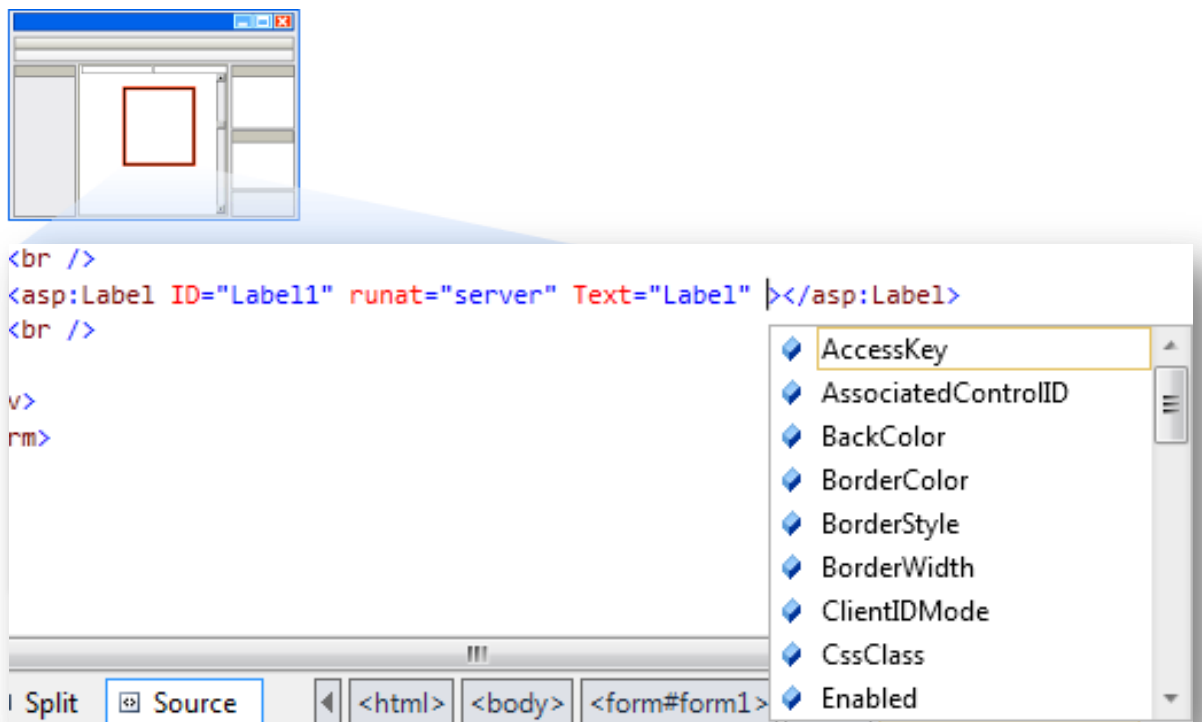
La vista Código fuente muestra el HTML para la página, incluso los elementos que Visual Studio ha creado para los controles de servidor. Los controles se declaran utilizando sintaxis de tipo HTML, con la excepción de que las etiquetas utilizan el prefijo *asp* e incluyen el atributo *runat="server"*.

Las propiedades del control se declaran como atributos. Por ejemplo, cuando estableció la propiedad Text del control Button en el paso 1, en realidad estableció el atributo Text del marcado del control.

Observe que todos los controles están dentro de un elemento form, que también tiene el atributo runat="server". El atributo runat="server" y el prefijo asp de las etiquetas de los controles marcan los controles para que ASP.NET los procese en el servidor cuando se ejecuta la página. El código que se encuentra fuera de los elementos <form runat="server"> y <script runat="server"> se envía sin cambios al explorador; este es el motivo por el que el código ASP.NET debe estar dentro de un elemento cuya etiqueta de apertura contiene el atributo runat="server".

3. Sitúe el punto de inserción detrás de asp:Label en la etiqueta <asp:Label> y, a continuación, presione la BARRA ESPACIADORA.

Aparece una lista desplegable que muestra las propiedades que se pueden establecer para un control Label. Esta característica, denominada IntelliSense, ayuda en la vista Código fuente con la sintaxis de los controles de servidor, los elementos HTML y otros elementos de la página. La siguiente ilustración muestra la lista desplegable de IntelliSense para el control Label.



4. Seleccione ForeColor y, a continuación, escriba un signo igual y comillas (=").

IntelliSense muestra una lista de colores.

#### Nota

**Puede mostrar una lista desplegable de IntelliSense en cualquier momento presionando CTRL+J.**

5. Seleccione un color para el texto del control Label. Asegúrese de seleccionar un color suficientemente oscuro para que se pueda leer el texto sobre un fondo de color blanco.

El atributo ForeColor se completa con el color que ha seleccionado, incluidas las comillas de cierre.

## Programar el control Button

En este ejercicio, escribiremos el código que lee el nombre especificado por el usuario en el cuadro de texto y lo muestra en el Label.

### Para agregar un controlador de eventos de botón predeterminado

1. Cambie a la vista Diseño.
2. Haga doble clic en el control Button.

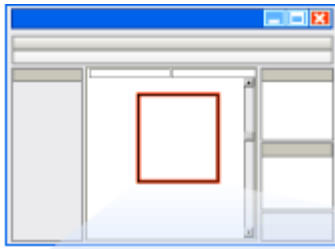
Visual Studio intercambia a la vista Código fuente y crea un esquema del controlador de eventos para el evento predeterminado de control Button, el evento Click.

#### Nota

**Hacer doble clic en un control en la vista de diseño es solo una de las diversas formas de crear controladores de eventos.**

3. Dentro del controlador, escriba lblNombre seguido por un punto (.).

Al escribir el punto después de Etiqueta, Visual Studio muestra una lista de miembros disponibles para el control Label (lblNombre), como se muestra en la siguiente ilustración.



```
protected void btnAceptar_Click(object sender, EventArgs e)
{
    lblNombre.|
}
```

4. Complete el controlador de eventos Click para el botón de manera que tenga el aspecto mostrado en el ejemplo de código siguiente.

```
protected void btnAceptar_Click(object sender, EventArgs e)
{
    lblNombre.Text = txtNombre.Text + ", Bienvenido a Visual Studio 2010!";
}
```

5. Desplácese hasta el elemento <asp:Button>. Observe que el elemento <asp:Button> tiene ahora el atributo onclick="btnAceptar\_Click".

Este atributo enlaza el evento Click del botón al método de controlador que codificó en el paso anterior.

Los métodos de control de eventos pueden tener cualquier nombre; el nombre que ve es el nombre predeterminado creado por Visual Studio. Lo importante es que el nombre utilizado para el atributo onclick debe coincidir con el nombre de un método de la página.

## Ejecutar la página

Ahora puede probar los controles de servidor en la página.

### Para ejecutar la página

1. Presione CTRL+F5 para ejecutar la página en el explorador.

La página se ejecuta de nuevo, utilizando el Servidor de desarrollo de Visual Studio.

2. Escriba un nombre en el cuadro de texto y haga clic en el botón.

El nombre especificado se muestra en el control Label. Tenga en cuenta que cuando se hace clic en el botón, la página se publica en el servidor web. ASP.NET vuelve a crear la página, ejecuta el código (en este caso, se ejecuta el controlador de eventos Click del control Button) y envía la nueva página al explorador. Si observa la barra de estado del explorador, puede ver que la página realiza un viaje de ida y vuelta al servidor web cada vez que se hace clic en el botón.

3. En el explorador, vea el código fuente de la página que está ejecutando.

En el código fuente de la página, solo se ve HTML ordinario; no se ven los elementos `<asp:>` con los que ha trabajado en la vista Código fuente. Cuando la página se ejecuta, ASP.NET procesa los controles de servidor y representa en la página los elementos HTML que realizan las funciones que representan el control. Por ejemplo, el control `<asp:Button>` se representa como el elemento `<input type="submit">` de HTML.

4. Cierre el explorador.

## Trabajar con controles adicionales

---

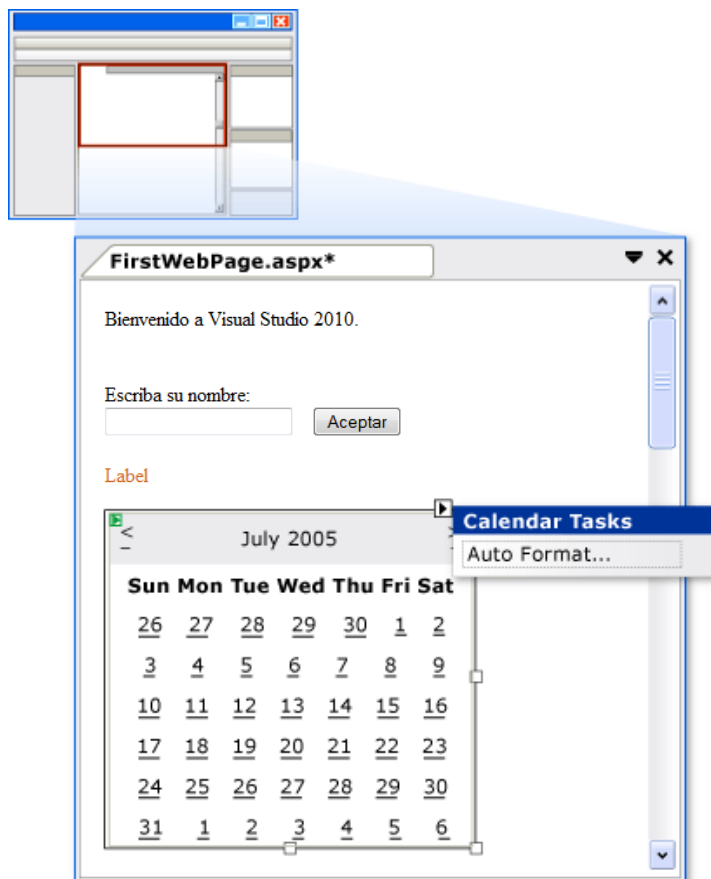
En esta parte del tutorial, trabajará con el control Calendar, que muestra las fechas mes por mes. El control Calendar es más complejo que los controles de botón, cuadro de texto y etiqueta con los que ha trabajado, e ilustra algunas funciones adicionales de los controles de servidor.

En esta sección, agregará un control `System.Web.UI.WebControls.Calendar` a la página y le dará formato.

### Para agregar un control Calendar

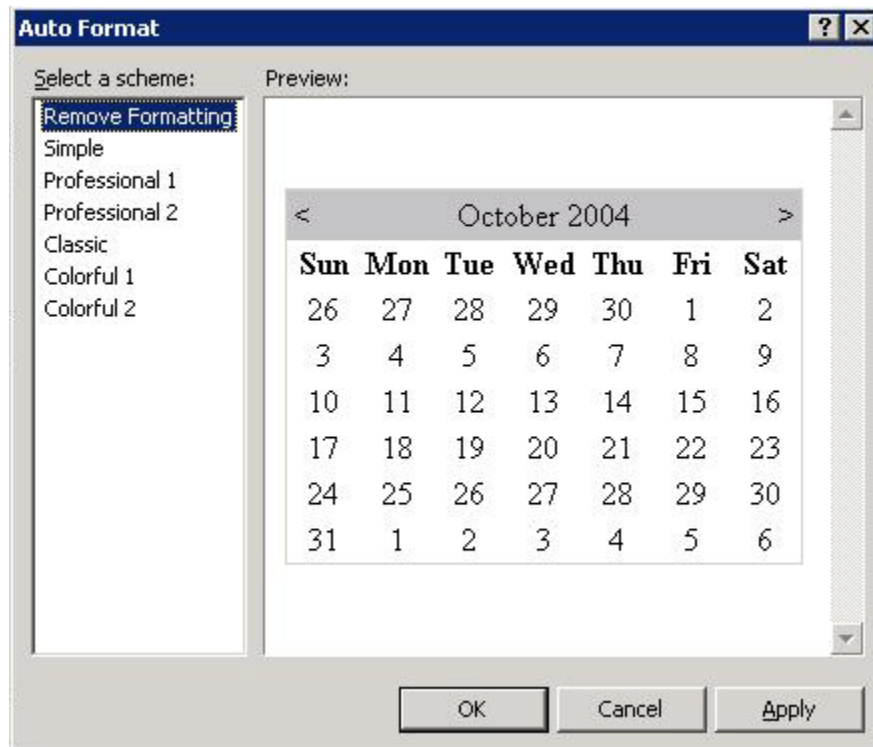
1. En Visual Studio, intercambie a la vista Diseño.
2. De la sección Estándar del Cuadro de herramientas, arrastre un control Calendar a la página y colóquelo debajo del elemento div que contiene los otros controles:

Aparece el panel de etiquetas inteligentes del calendario. El panel muestra comandos que ayudan a realizar las tareas más comunes del control seleccionado. En la siguiente ilustración se muestra cómo se presenta el control Calendar en la Vista de diseño.



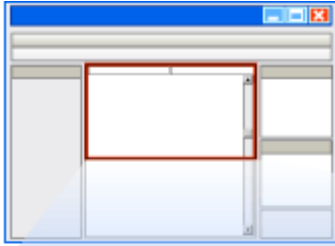
3. En el panel de etiquetas inteligentes, elija Formato automático.

Se muestra el cuadro de diálogo Formato automático, que permite seleccionar un esquema de formato para el calendario. La siguiente ilustración muestra el cuadro de diálogo Formato automático para el control Calendar.



4. En la lista Seleccionar esquema, seleccione Simple y, a continuación, haga clic en Aceptar.
5. Cambie a la vista Código fuente.

Puede ver el elemento `<asp:Calendar>`. Este elemento es mucho más largo que los elementos de los controles sencillos creados anteriormente. También incluye subelementos, como `<WeekEndDayStyle>`, que representa las distintas configuraciones de formato. En la siguiente ilustración se muestra el control Calendar en la vista Código fuente. (El marcado exacto que ve en vista Código fuente podría diferir ligeramente de la ilustración.)



```
<asp:Calendar ID="Calendario" runat="server" BackColor="White"
    BorderColor="White" BorderWidth="1px" Font-Names="Verdana" Font-Size="9pt"
    ForeColor="Black" Height="190px" NextPrevFormat="FullMonth"
    onselectionchanged="Calendar1_SelectionChanged" Width="350px">
    <DayHeaderStyle Font-Bold="True" Font-Size="8pt" />
    <NextPrevStyle Font-Bold="True" Font-Size="8pt" ForeColor="#333333"
        VerticalAlign="Bottom" />
    <OtherMonthDayStyle ForeColor="#999999" />
    <SelectedDayStyle BackColor="#333399" ForeColor="White" />
    <TitleStyle BackColor="White" BorderColor="Black" BorderWidth="4px"
        Font-Bold="True" Font-Size="12pt" ForeColor="#333399" />
    <TodayDayStyle BackColor="#CCCCCC" />
</asp:Calendar>
```

## Programar el control Calendar

En esta sección, programará el control Calendar para que muestre la fecha actualmente seleccionada.

### Para programar el control Calendar

1. En vista Diseño, haga doble clic en el control Calendar.

Un nuevo controlador de eventos se crea en la vista Código fuente.

2. Complete el controlador de eventos SelectionChanged con el código resaltado siguiente.

```
protected void Calendar1_SelectionChanged(object sender, EventArgs e)
{
    lblNombre.Text = Calendario.SelectedDate.ToLongDateString();
}
```



## **Ejecutar la página**

Ya puede probar el calendario.

### **Para ejecutar la página**

1. Presione CTRL+F5 para ejecutar la página en el explorador.
2. Haga clic en una fecha del calendario.

La fecha especificada se muestra en el control Label.

3. En el explorador, vea el código fuente de la página.

El control Calendar se ha representado en la página como una tabla, con cada día como un elemento td que contiene un elemento a.

4. Cierre el explorador.