



Capitolo 15

Applicazioni Web

- 15.1 **ASP.NET**
- 15.2 **Internet Information Services**
- 15.3 **URL**
- 15.4 **Creazione di un'applicazione Web**
- 15.5 **Un'applicazione Web per database**
- 15.6 **Conclusione**
- 15.7 **Quiz**

Nel corso del presente volume sono state scritte applicazioni Windows. Di fatto molte applicazioni con cui interagiscono gli utenti sono applicazioni Windows. Per esempio Microsoft Word e Outlook sono applicazioni molto diffuse.

Tuttavia gli utenti interagiscono sempre più spesso con applicazioni Web. Un tipo di applicazione Web molto diffuso è l'e-commerce, dove la “e” sta per “elettronico”. Per esempio, nel sito Web di Amazon o di un'altra libreria online, è possibile selezionare un libro o un altro prodotto, inserire il prodotto scelto in un carrello virtuale e al termine portarsi alla cassa virtuale, inserire le informazioni relative alla propria carta di credito ed effettuare un acquisto. È quindi possibile seguire la consegna del prodotto nel suo viaggio.

Nel presente capitolo si esamina la creazione di un'applicazione Web che visualizza informazioni presenti in un database, simile all'applicazione Windows creata nel Capitolo 14.

15.1 ASP.NET

ASP.NET è un termine noto ai progettisti di applicazioni Web e indica i componenti di codice utilizzati per le applicazioni Web, analogamente al modo in cui ADO.NET fa riferimento ai componenti di codice utilizzati per l'accesso ai database.

Come per ADO.NET, la parte “.NET” di ASP.NET è nota. ASP è l'acronimo di Active Service Pages, una tecnologia Microsoft per le applicazioni Web precedente ad ASP.NET. Per chi conosce ASP, è molto più facile lavorare con ASP.NET. ASP mescola codice HTML e codice script; al contrario, ASP.



NET consente di sviluppare applicazioni Web quasi allo stesso modo con cui si sviluppano applicazioni Windows.

ASP.NET è nato con il predecessore di Visual Studio 2005, Visual Studio .NET. A quel tempo il numero di versione di ASP.NET era 1.x (prima 1.0, quindi 1.1). Con Visual Studio 2005, il numero di versione è 2.x, a partire da 2.0.

Esistono altre tecnologie concorrenti per la creazione di applicazioni Web. ASP.NET è di Microsoft e di conseguenza largamente supportata in Visual Studio 2005.

15.2 Internet Information Services

Visual Studio 2005 richiede uno dei seguenti sistemi operativi: Windows 2000 Professional, Windows XP Home o Professional, Windows 2000 Server o Windows 2003 Server. Su tutti i sistemi, ad eccezione di Windows XP Home Edition, Internet Information Services (IIS) è un componente opzionale che può essere installato con il sistema operativo. IIS potrebbe non essere installato sul proprio computer, perché può non fare parte dell'installazione predefinita del sistema operativo. Tuttavia, se IIS non è installato, può essere aggiunto, come descritto nel presente paragrafo.

NOTA *Non è possibile installare IIS su Windows XP Home Edition, se non vengono effettuate alcune modifiche al Registro di sistema non supportate da Microsoft e quindi sconsigliate.*

A differenza di ASP.NET 1.x e Visual Studio.NET, ASP.NET 2.x e Visual Studio 2005 non richiedono l'installazione di IIS per la creazione di applicazioni Web eseguite localmente (vale a dire sul proprio computer). Tuttavia, a meno di avere Windows XP Home Edition, l'installazione di IIS offre più opzioni, quali per esempio rendere accessibili le pagine Web da altri computer, oltre a quello locale, e non costituisce ulteriori costi.

Determinare se IIS è già installato

Per determinare se IIS è già installato sul computer, aprire Installazione applicazioni dal Pannello di controllo. Dalla barra di menu a sinistra, scegliere Installazione componenti di Windows. Viene visualizzata la finestra Aggiunta guidata componenti di Windows, mostrata nella Figura 15.1.

Nella Figura 15.1, Internet Information Services (IIS) è selezionato, ma con sfondo scuro, a indicare che sono installati alcuni componenti di IIS, ma non tutti. Se la casella è selezionata con sfondo bianco, come per Internet Explorer nella Figura 15.1, sono installati tutti i componenti di IIS. Se IIS non è selezionato, come per Servizio indicizzazione nella Figura 15.1, IIS non è installato.

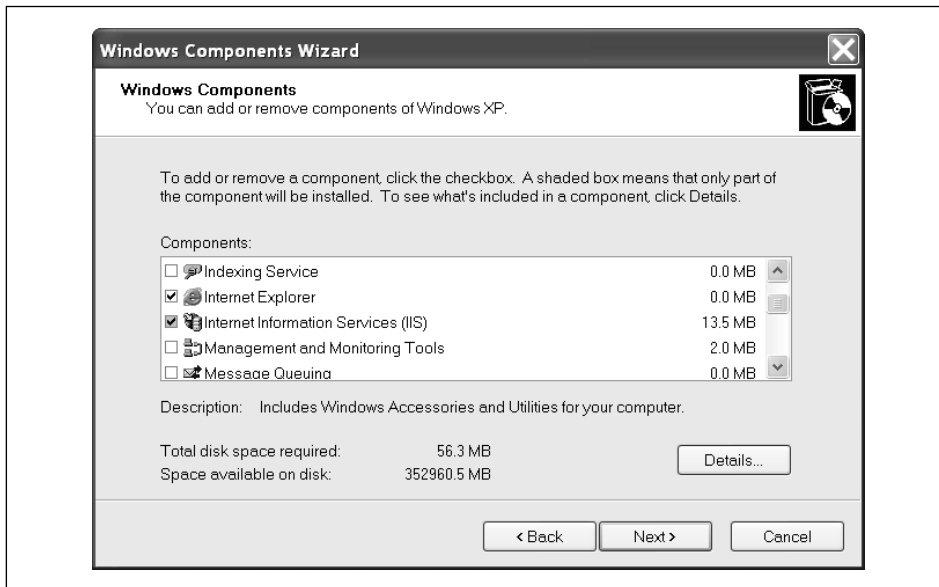


Figura 15.1 Aggiunta guidata componenti di Windows.

Se IIS è selezionato, ma con sfondo scuro come nella Figura 15.1, è necessario controllare quali componenti sono installati. Per effettuare questa operazione, nella finestra Aggiunta guidata componenti di Windows, evidenziare Internet Information Services (IIS) e fare clic sul pulsante Dettagli. Viene visualizzata, come nella Figura 15.2, una finestra di dialogo che mostra i singoli componenti di *Internet Information Services (IIS)*.

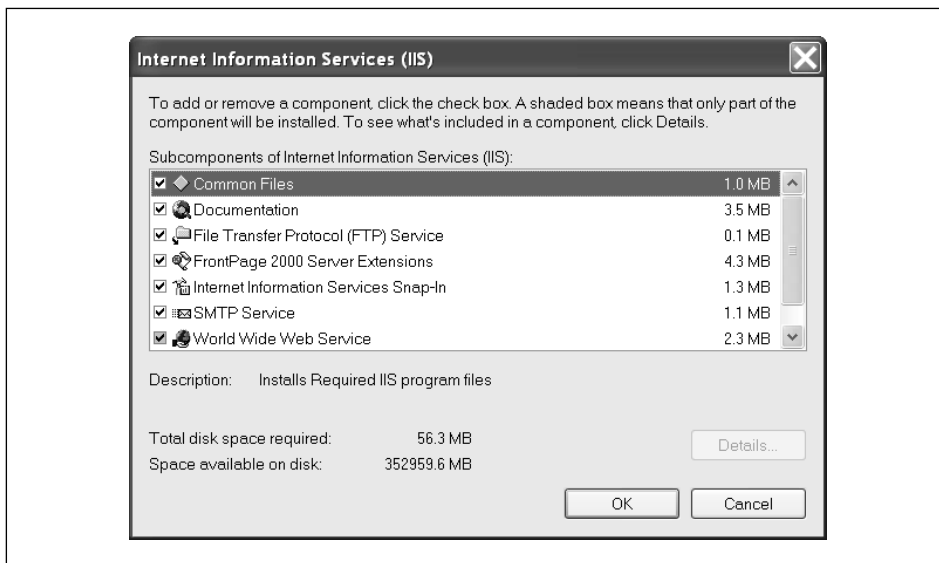


Figura 15.2 Componenti di IIS.

Nella Figura 15.2, sono selezionate quasi tutte le caselle di controllo, perché tali componenti sono installati nel computer dell'autore. Il proprio computer può trovarsi in una situazione diversa, secondo i componenti di IIS installati in precedenza.

Non sono necessari i servizi **FTP** (*File Transfer Protocol*) e **SMTP** (*Simple Mail Transfer Protocol*), ma si consiglia l'installazione degli altri componenti.

Installazione di IIS

Se è necessario installare IIS o suoi componenti, è necessario utilizzare il CD di installazione del sistema operativo. Se IIS non è selezionato nella finestra *Aggiunta guidata componenti di Windows* (Figura 15.1), selezionarla e fare clic sul pulsante *Avanti*. Se IIS è selezionato nella finestra *Aggiunta guidata componenti di Windows*, ma la casella di controllo ha lo sfondo scuro, fare semplicemente clic sul pulsante *Avanti*.

Facendo clic sul pulsante *Avanti* viene visualizzata la finestra di dialogo *Internet Information Services (IIS)* mostrata nella Figura 15.2. Scegliere tutti i componenti selezionando le caselle non selezionate, con la possibile eccezione dei servizi FTP ed SMTP. Quindi fare clic sul pulsante *OK*; si torna ad *Aggiunta guidata componenti di Windows*. Nella procedura guidata, dopo avere inserito il CD di installazione del sistema operativo nell'unità CD-ROM, fare clic sul pulsante *Avanti* e proseguire fino al termine dell'aggiunta dei componenti di IIS. Se richiesto, riavviare il computer.

Avvio del servizio Amministrazione di IIS

Il servizio *Amministrazione di IIS* è un servizio utilizzato per l'amministrazione di IIS. Sebbene esistano metodi alternativi per amministrare IIS, il servizio *Amministrazione di IIS* può risultare il più semplice.

Aprire in *Pannello di controllo* la cartella *Strumenti di amministrazione*, mostrata nella Figura 15.3.

Scegliere il collegamento *Servizi* per aprire la cartella *Servizi*. Scegliere la scheda *Esteso* ed evidenziare *Amministrazione di IIS*. Come mostra la Figura 15.4, a sinistra si trova una descrizione del servizio *Amministrazione di IIS* e opzioni per arrestare, sospendere e riavviare il servizio.

Tali opzioni sono disponibili perché il servizio è già avviato. In tal caso esiste la conferma del fatto che il servizio *Amministrazione di IIS* è avviato e il passaggio è completo.

Se invece il servizio *Amministrazione di IIS* è stato arrestato o non è mai stato avviato, l'unica opzione disponibile è l'avvio del servizio, come mostrato nella Figura 15.5. In tal caso scegliere *Avvia* per avviare il servizio.

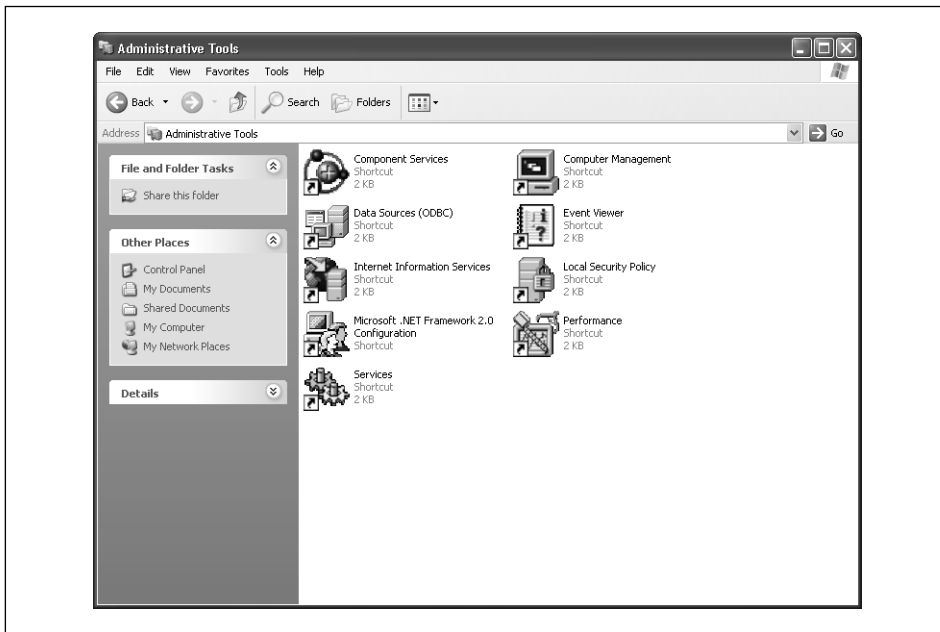


Figura 15.3 Strumenti di amministrazione in Pannello di controllo.

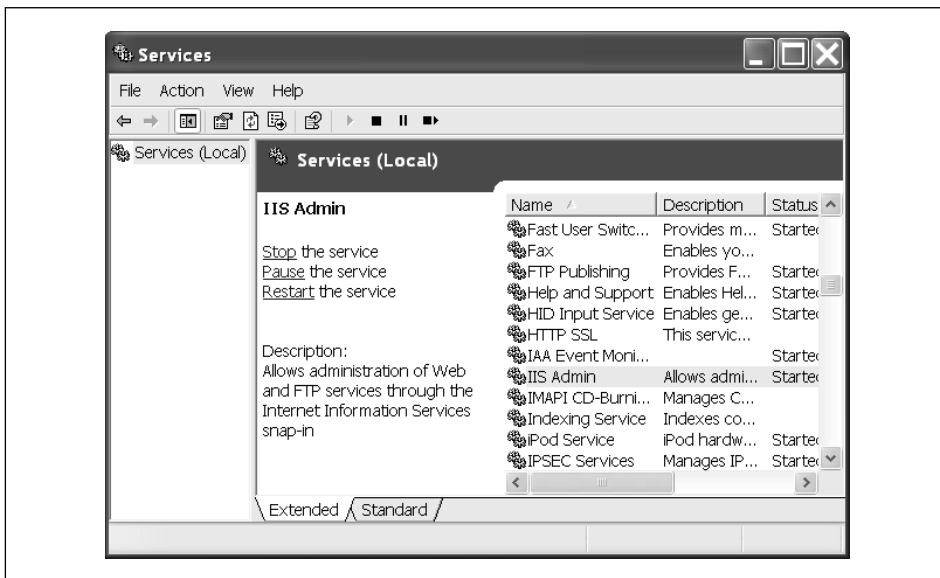


Figura 15.4 Cartella Servizi con il servizio di amministrazione di IIS selezionato.

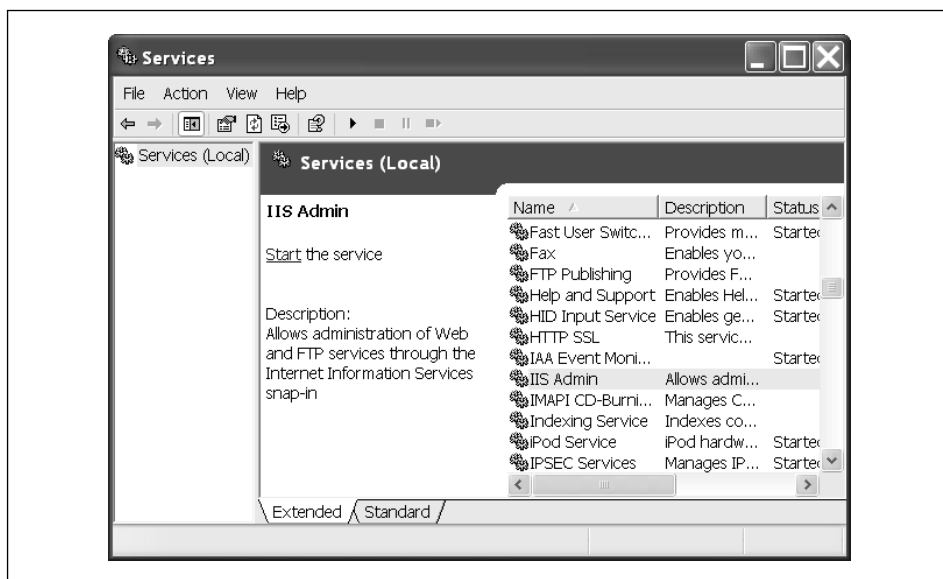


Figura 15.5 Opzioni per l'avvio del servizio Amministrazione di IIS.

Avvio del sito Web predefinito

Una volta controllato l'avvio del servizio Amministrazione di IIS, chiudere la cartella Servizi e tornare alla cartella Strumenti di amministrazione mostrata nella Figura 15-3. Quindi scegliere il collegamento Internet Information Services per aprire la finestra di dialogo Internet Information Services mostrata nella Figura 15.6.

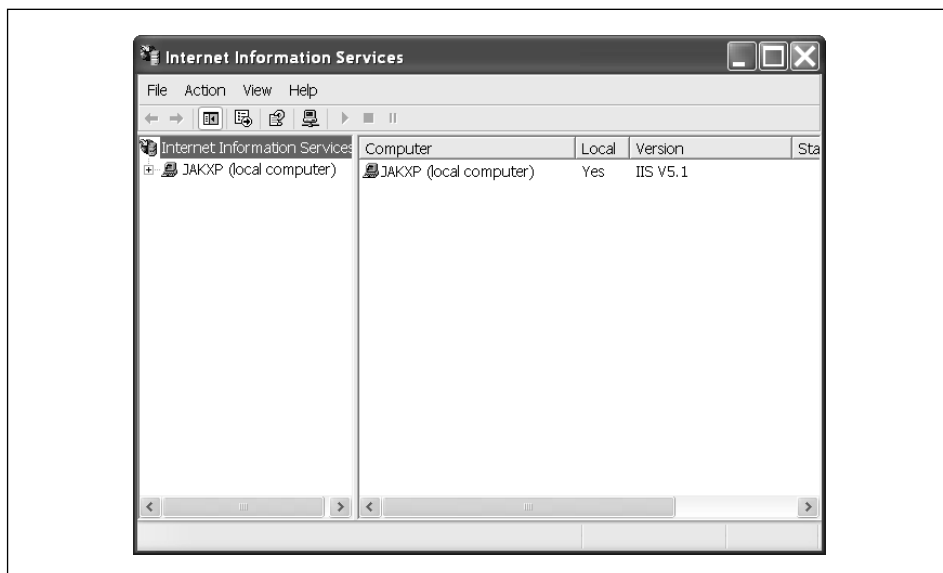


Figura 15.6 Finestra di dialogo Internet Information Services.

Fare clic sul segno + accanto al nome del computer locale, quindi fare clic sul segno + accanto alla cartella Siti Web sotto di esso. La Figura 15.7 mostra una sottocartella chiamata Sito Web predefinito.

Se Sito Web predefinito è seguito da un'indicazione fra parentesi che indica che è stato arrestato, fare clic con il tasto destro del mouse su Sito Web predefinito e scegliere Avvia dal menu di scelta rapida.

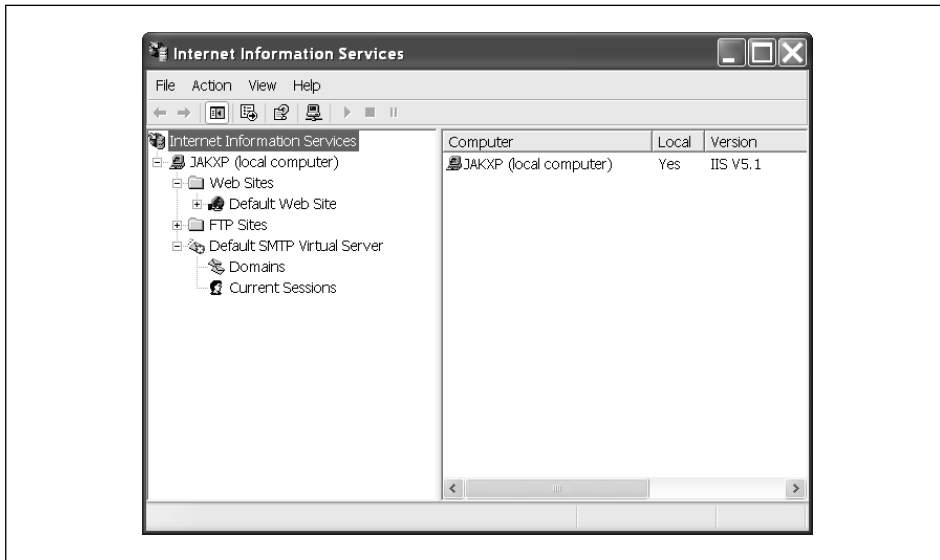


Figura 15.7 Sito Web predefinito nella finestra di dialogo Internet Information Services.

15.3 URL

La propria abitazione dispone di un indirizzo mediante cui può essere individuata. Analogamente una pagina Web dispone di un indirizzo mediante cui è possibile individuarla attraverso il browser Web.

Solitamente l'indirizzo di un'abitazione è indicato sotto forma di un nome di strada, piazza e così via, seguito da un numero, per esempio via Dei Giardini 15. Analogamente l'indirizzo di una pagina Web, noto come URL (l'acronimo di Uniform Resource Locator), è costituito da una forma ben definita.

La spiegazione di seguito utilizza come esempio l'URL per la home page di Microsoft, <http://www.microsoft.com/default.aspx>.

La prima parte dell'indirizzo (in questo caso http) indica il protocollo da utilizzare. HTTP è l'acronimo di Hypertext Transfer Protocol e definisce il modo in cui sono formattati e trasmessi i messaggi e le azioni che devono effettuare server Web e browser Web in risposta a diversi comandi. Per esempio, quando si inserisce un URL nel browser, in realtà il browser invia un comando HTTP al server Web, chiedendo il recupero e la trasmissione della pagina Web richiesta.

Esistono protocolli diversi da HTTP. Uno di essi, HTTPS, è una forma protetta di HTTP utilizzata spesso per le transazioni su Internet con carta di credito; un altro, **FTP** (*File Transfer Protocol*) viene utilizzato per il trasferimento di file.

La seconda parte dell'indirizzo (in questo caso `www.microsoft.com`) è il nome di dominio in cui si trova la risorsa. Solitamente i nomi di dominio iniziano con **www**, abbreviazione di *World Wide Web*, e terminano con `com` (per i siti commerciali) o con altre estensioni, quali per esempio `net` od `org`. Nella parte centrale si trova un nome (in questo caso Microsoft), che spesso corrisponde all'organizzazione o alla persona che possiede il sito Web. Per esempio, il sito Web dell'autore è `http://www.genghiskhent.com`, basato sul soprannome assegnatogli dai suoi studenti, Genghis Khent.

La terza parte dell'indirizzo è la pagina Web specifica a cui accedere (in questo caso `default.aspx`). Le pagine Web hanno nomi simili ai nomi di altri file, un nome descrittivo seguito da un punto e da un'estensione.

Nelle applicazioni Windows l'estensione indica l'applicazione utilizzata per aprire il file, quale per esempio `.doc` per Microsoft Word, `.xls` per Microsoft Excel e così via. Le pagine Web possono avere estensioni quali per esempio `.htm` e `.html`. L'estensione `.aspx` indica che la pagina Web fa parte di un'applicazione ASP.NET.

Il proprio computer come server Web

Un server Web è un computer che invia (serve) pagine Web. Per esempio, se si visita la home page di Microsoft, `http://www.microsoft.com/default.aspx`, inserendo l'indirizzo nel browser Web (quale per esempio Internet Explorer, Netscape o Mozilla), un computer su Internet recupera una pagina sul sito Web di Microsoft e ne invia il contenuto al browser, dove viene visualizzato.

Nel presente capitolo, tuttavia, il proprio computer agisce da server Web per le applicazioni Web che verranno create.

Scrivere nel proprio browser Web l'URL `http://localhost/` (come si è già detto, questa operazione non è possibile con Windows XP Home). La Figura 15.8 mostra la pagina Web visualizzata dal sistema operativo Windows XP.

Ci si può chiedere cosa rappresenti `localhost`: rappresenta il proprio computer, che ora agisce da server Web.

Percorsi fisici e virtuali

Quando si scrive `http://www.microsoft.com` nel browser Web, si accede a una pagina memorizzata nell'hard disc di un computer utilizzato da Microsoft come server Web.

Analogamente, quando si scrive `http://localhost` e viene visualizzata la pagina Web mostrata nella Figura 15.8, anche tale pagina Web è memorizzata nell'hard disc del proprio computer.

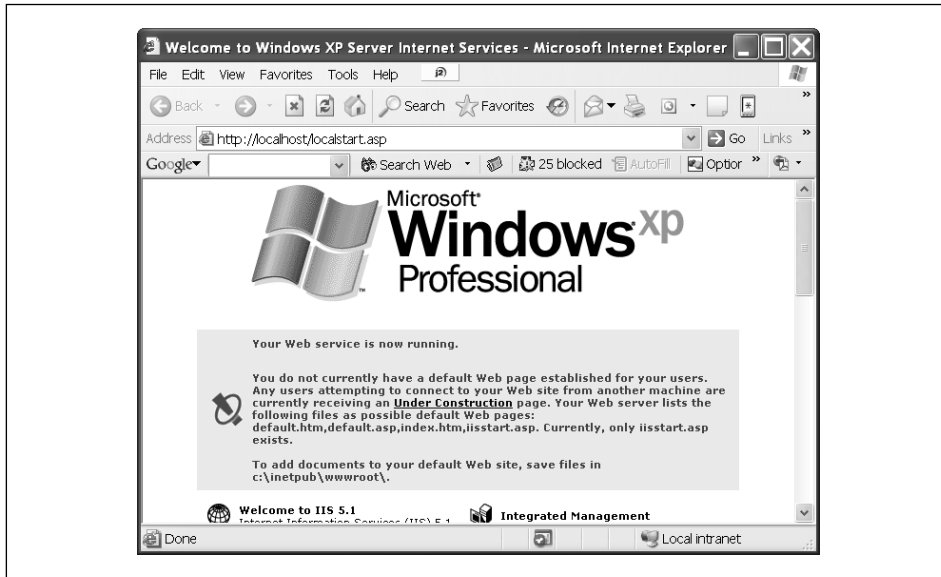


Figura 15.8 Pagina Web predefinita.

Per impostazione predefinita, `http://localhost` effettua una mappatura sulla cartella `C:\inetpub\wwwroot` dell'hard disc. È possibile controllare tale affermazione facendo clic con il tasto destro del mouse su Sito Web predefinito (Figura 15.7) e scegliendo Proprietà dal menu di scelta rapida, per visualizzare la finestra di dialogo Proprietà di Sito Web predefinito, mostrata nella

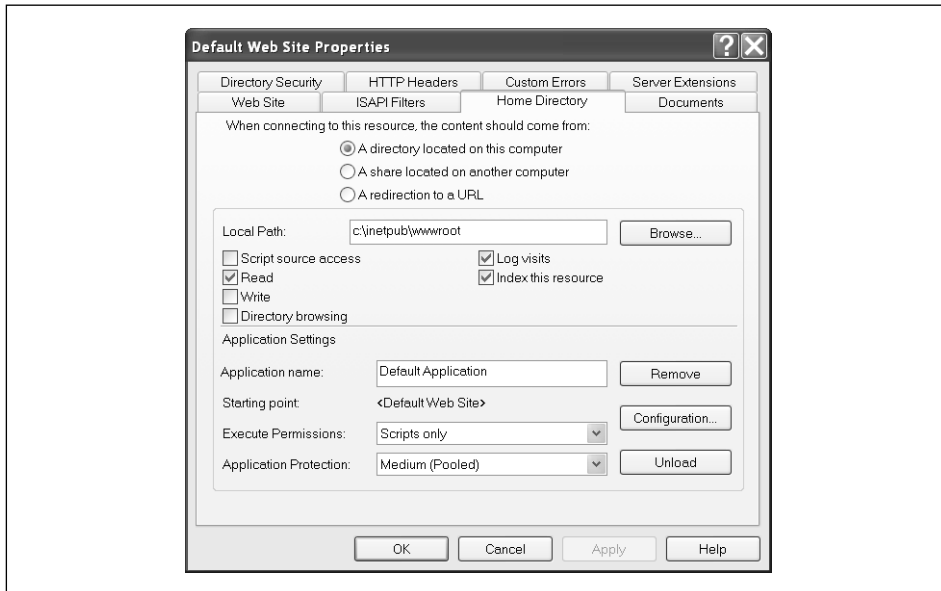


Figura 15.9 Finestra di dialogo Proprietà di sito Web predefinito.

Figura 15.9 con la scheda Home Directory selezionata. Il percorso locale è `c:\inetpub\wwwroot`.

La barra degli indirizzi nella Figura 15.8 mostra che l'URL della pagina Web è `http://localhost/localstart.asp`. Quindi l'URL `http://localhost/localstart.asp` effettua una mappatura sul file `C:\inetpub\Wwwroot\localstart.asp` nel proprio hard disc.

L'URL Web `http://localhost/localstart.asp` è noto come *percorso virtuale* della pagina Web. Il percorso di file `C:\inetpub\wwwroot\localstart.asp` è noto come percorso fisico della pagina Web; tuttavia puntano entrambi alla medesima posizione.

15.4 Creazione di un'applicazione Web

La creazione di un'applicazione Web è diversa dalla creazione di un'applicazione Windows: si utilizza il comando di menu `File | New | Website` in luogo del comando di menu `File | New | Project`.

Il comando di menu `File | New | Website` visualizza la finestra di dialogo `New Web Site` mostrata nella Figura 15.10.

Il riquadro superiore mostra i modelli disponibili; scegliere `ASP.NET Web Site`. Questa è la scelta adatta per la creazione di un sito Web con il supporto ASP.NET, l'operazione desiderata nell'esempio.

Nella casella di riepilogo a discesa `Location`, scegliere `File System`. Le altre possibilità, `FTP` e `HTTP`, protocolli descritti in precedenza nel capitolo,

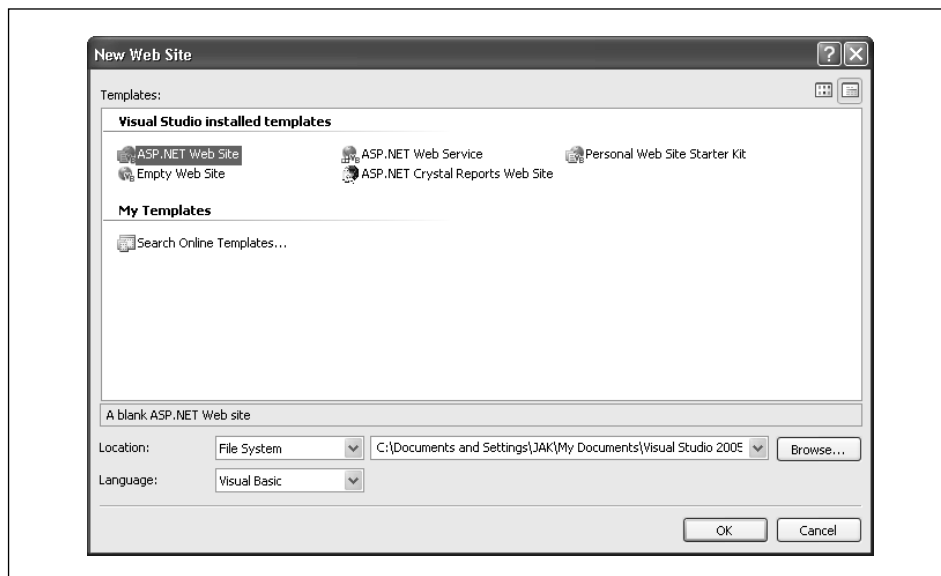


Figura 15.10 Finestra di dialogo `New Web Site`.

sono per la creazione di siti Web ASP.NET su altri computer. Nel presente capitolo il sito Web viene creato sul computer locale.

Nella casella di riepilogo a discesa *Language*, scegliere *Visual Basic*. Le altre possibilità, *Visual C#* e *Visual J#*, sono altri linguaggi in *Visual Studio 2005* che possono essere utilizzati per la creazione di un'applicazione ASP.NET.

Fare clic sul pulsante *Browse* per selezionare la posizione nell'hard disc in cui si desidera creare i file per l'applicazione Web ASP.NET, per esempio una cartella *Visual Basic* creata nella cartella *Visual Studio Projects in Documents*. Dopo il percorso per la cartella *Visual Basic* (per esempio *D:\Documents and Settings\JAK\Documents\Visual Studio Projects\Visual Basic*) scrivere *WebSite1* come nome per il progetto. Naturalmente, se si desidera, per il proprio progetto è possibile scegliere un nome e un percorso diversi.

Al termine fare clic sul pulsante *OK*; *Visual Studio 2005* crea la struttura funzionante per un'applicazione ASP.NET.

ASP.NET Development Server

Quando *Visual Studio 2005* ha terminato la creazione dell'applicazione ASP.NET, eseguire l'applicazione scegliendo *Start* o *Start Without Debugging* dal menu *Debug*. Il risultato è una pagina Web vuota, come mostrato nella Figura 15.11.

L'URL visualizzato nella barra degli indirizzi del browser Web nella Figura 15.11 è `http://localhost:1040/WebSite1/Default.aspx`. La parte `http://localhost` dell'URL è spiegata in precedenza nel paragrafo "Il proprio computer come

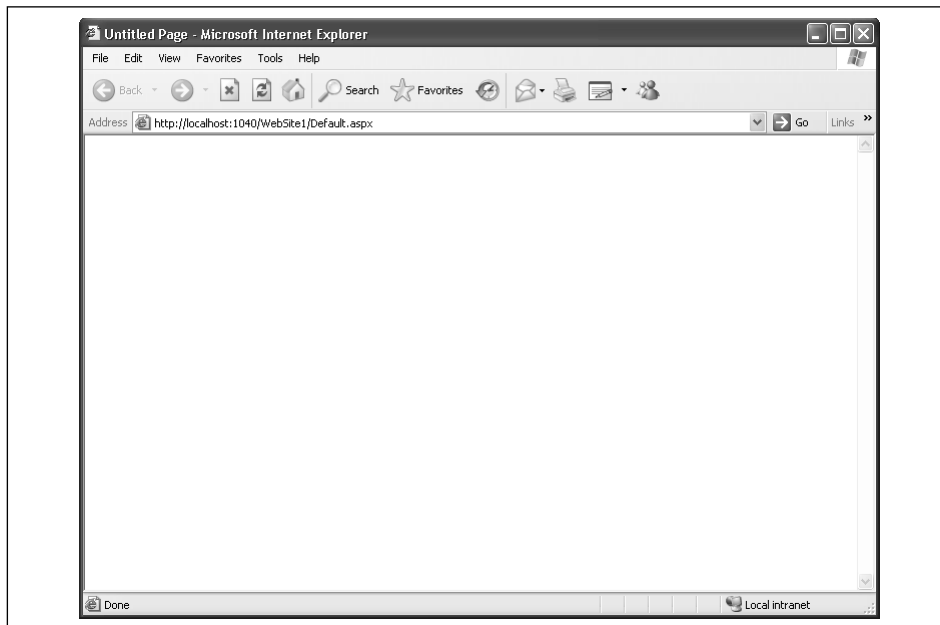


Figura 15.11 Pagina Web ASP.NET.

server Web”. WebSite1 è il nome dell’applicazione Web e default.aspx è il nome della pagina Web (o del form Web) creata da Visual Studio 2005 per impostazione predefinita, molto simile a un form Windows creato per impostazione predefinita quando si crea un’applicazione Windows. La novità è rappresentata dalla parte “:1040” successiva a localhost. I due punti (:) indicano che il numero di seguito è un numero di porta (in questo caso 1040).

NOTA *Il particolare numero di porta assegnato da Visual Studio 2005 può essere diverso da 1040.*

Una porta è una connessione logica (e non fisica) in un computer. Per esempio, quando si accede a una pagina Web con il browser Web, la richiesta (e la risposta del server Web) passa attraverso la porta 80.

Come detto in precedenza nel paragrafo “Internet Information Services,” ASP.NET 2.x e Visual Studio 2005 non richiedono l’installazione di IIS per la creazione di applicazioni Web eseguite localmente (vale a dire sul proprio computer e non su un computer su Internet). Le applicazioni Web locali sono gestite mediante ASP.NET Development Server, che utilizza diversi numeri di porta (in questo caso 1040).

Nella barra di sistema può essere presente un’icona per ASP.NET Develop-

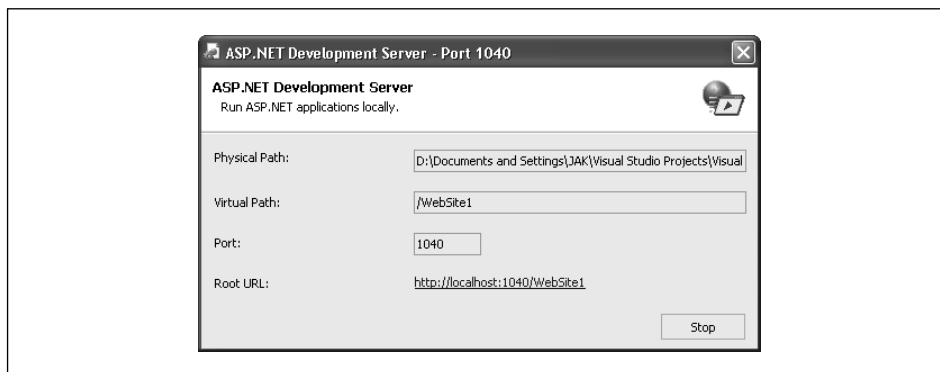


Figura 15.12 Finestra di dialogo ASPNET Development Server.

ment Server; se presente, fare doppio clic su di essa. Viene visualizzata la finestra di dialogo ASP.NET Development Server, mostrata nella Figura 15.12.

La finestra di dialogo ASP.NET Development Server visualizza le seguenti informazioni (sebbene non nel medesimo ordine indicato di seguito):

- **Percorso fisico.** Il percorso scelto nella finestra di dialogo New Web Site mostrata nella Figura 15.10.
- **Porta.** La porta scelta da ASP.NET Development Server per l’accesso alle applicazioni Web locali (in questo caso 1040).
- **URL di root.** La root o base per le applicazioni Web (http://localhost:1040), seguita dal nome dell’applicazione Web (in questo caso WebSite1).

- **Percorso virtuale.** Il percorso dall'URL di root di `http://localhost:1040` all'applicazione Web.

Questo è ciò che si può fare per ora con questa applicazione Web vuota. Chiudere la finestra di dialogo ASP.NET Development Server mostrata nella Figura 15.12 e la pagina Web vuota mostrata nella Figura 15.11.

IDE di applicazione ASP.NET

La Figura 15.13 mostra l'Integrated Development Environment (IDE) per l'applicazione ASP.NET creata facendo clic su OK nella finestra di dialogo New Web Site mostrata nella Figura 15.10.

Come per le applicazioni Windows, anche il form nelle applicazioni Web, spesso chiamato form Web, ha una vista progettazione (mostrata nella Figura 15.13), completa di Toolbox e Solution Explorer, e una vista codice, mostrata nella Figura 15.14. Questa analogia tra gli IDE per le applicazioni Windows e Web facilita l'apprendimento dello sviluppo di applicazioni Web.

Sebbene i rispettivi IDE di applicazioni Windows e Web siano simili, non sono uguali. Per esempio, il form Web dispone, oltre che delle viste progettazione e codice, di una vista HTML, mostrata nella Figura 15.15 e a cui si accede facendo clic sulla scheda Source, in cui è possibile visualizzare il codice HTML del form, che rappresenta una pagina Web.

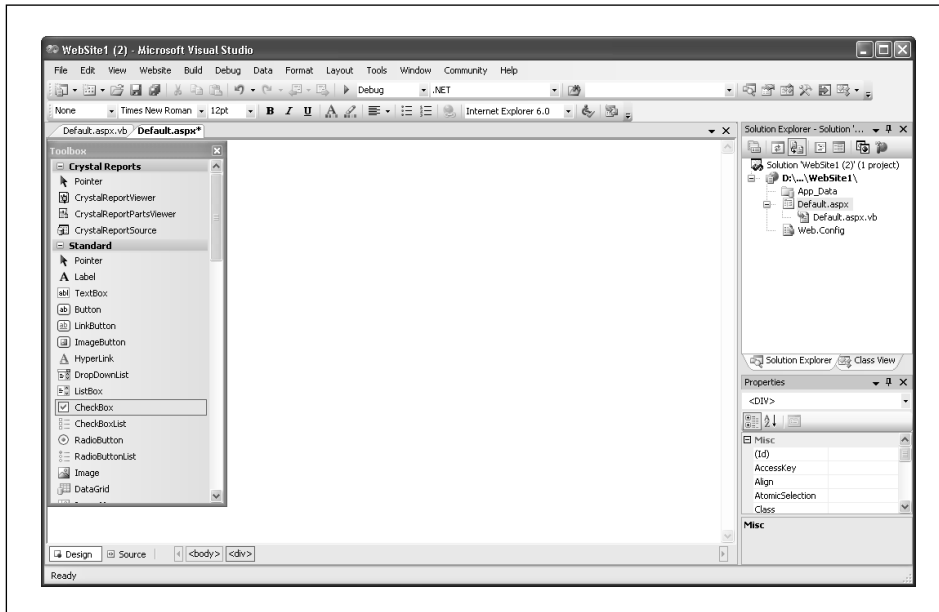


Figura 15.13 IDE di applicazione ASP.NET.

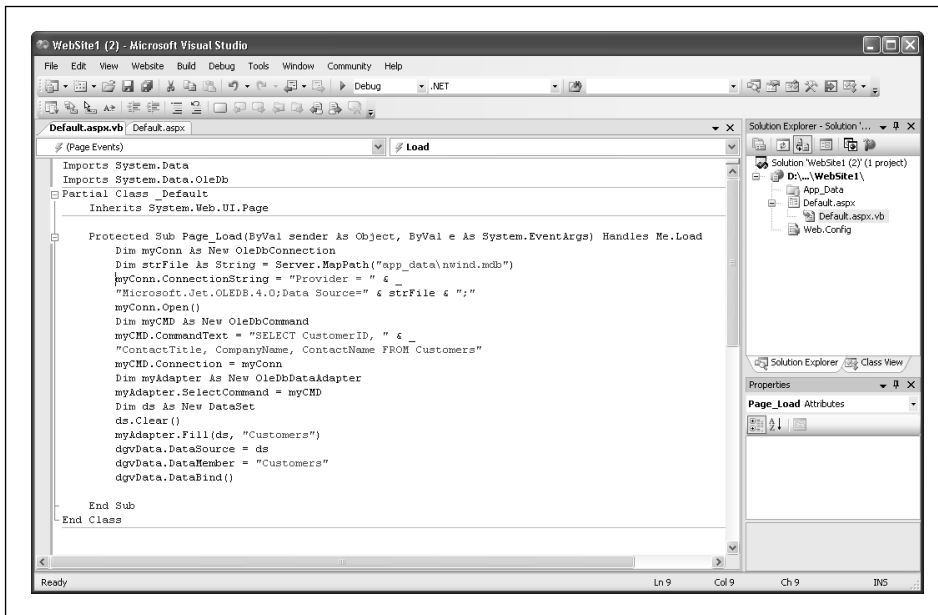


Figura 15.14 Vista codice.

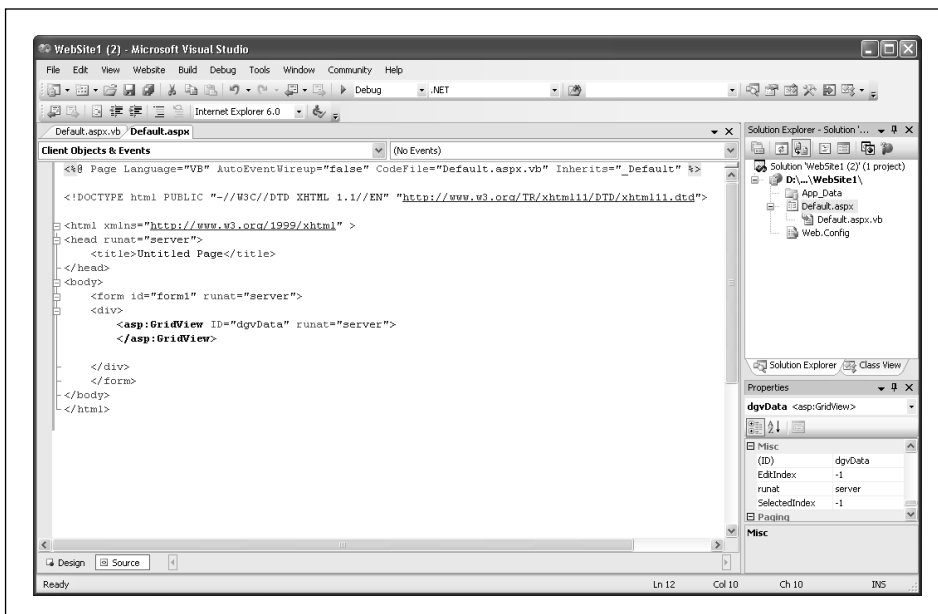


Figura 15.15 Vista HTML del form.

15.5 Un'applicazione Web per database

Verrà ora creata un'applicazione Web simile all'applicazione Windows creata nel Capitolo 14. L'applicazione Windows visualizzava in un controllo DataGridView i contenuti di quattro campi della tabella Customers del database Northwind Traders. Analogamente, l'applicazione Web che verrà creata visualizza il contenuto dei medesimi quattro campi della tabella Customers del database Northwind Traders, ma in un browser Web, come mostrato nella Figura 15.16.

Aggiunta di un controllo GridView

L'applicazione Windows creata nel Capitolo 14 dispone di un controllo DataGridView attraverso cui si visualizzano le informazioni sul database. L'applicazione Web equivalente al controllo Windows DataGridView ha un nome leggermente diverso, GridView.

Iniziare dall'applicazione Web creata nel paragrafo precedente. Visualizzare il form Web in vista progettazione e fare clic sulla scheda Design. Quindi cercare nella Toolbox un GridView nel gruppo Data, come mostrato nella Figura 15.17.

Se GridView non è visualizzato nella Toolbox, è necessario aggiungerlo. Fare clic con il tasto destro del mouse sulla Toolbox e scegliere Choose Items... dal menu di scelta rapida. Viene visualizzata la finestra di dialogo Choose ToolBox Items mostrata nella Figura 15.18.

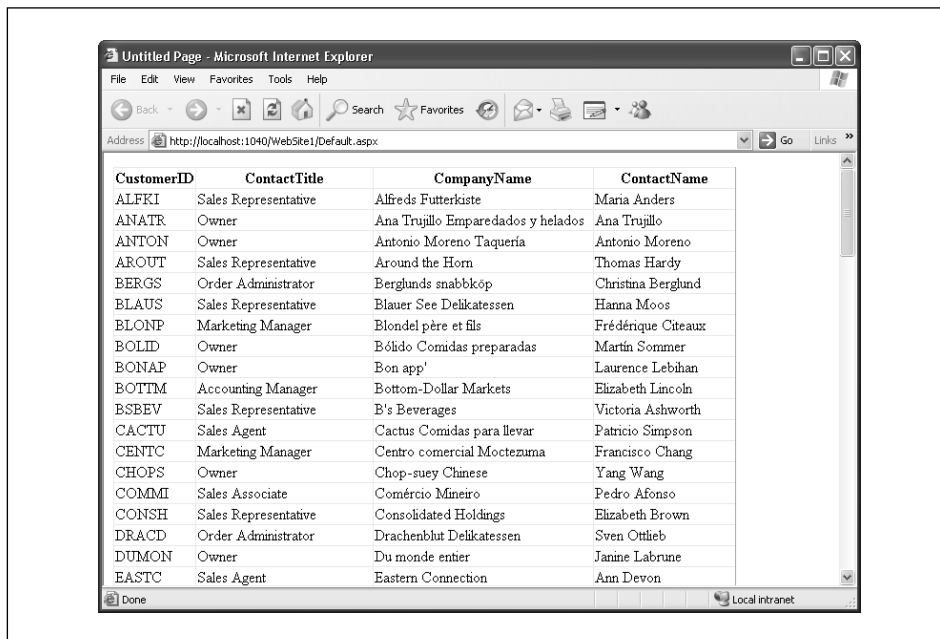


Figura 15.16 Applicazione Web in esecuzione.



Figura 15.17 GridView nella Toolbox.

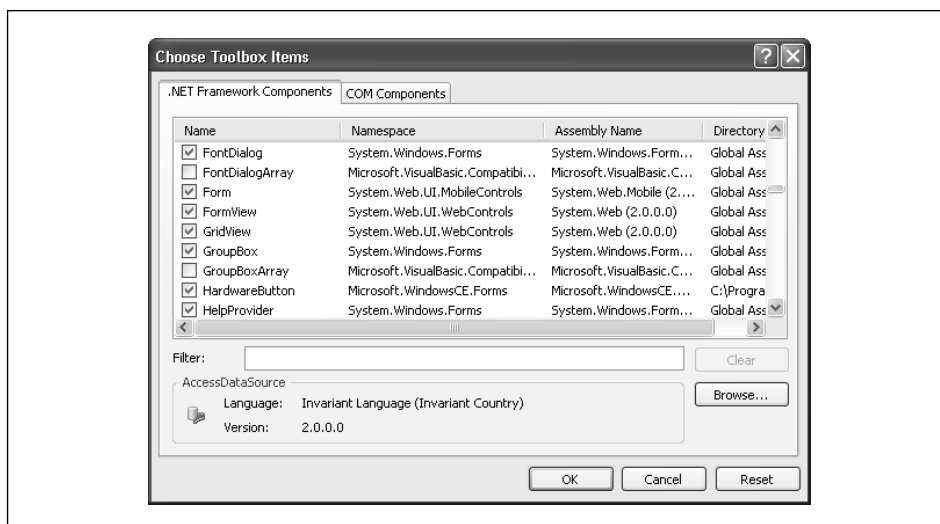


Figura 15.18 La finestra di dialogo Choose Toolbox Items.

Selezionare la casella di controllo per il GridView il cui spazio dei nomi sia System.Web.UI.WebControls. Quindi fare clic sul pulsante OK per chiudere la finestra di dialogo Choose ToolBox Items. Ora GridView è stato aggiunto alla Toolbox, come mostrato nella Figura 15.17.

Quando GridView si trova nella Toolbox, aggiungerlo al form Web trascinandolo o facendo doppio clic su di esso, come per un controllo di un form Windows.

Quando si aggiunge il controllo GridView, viene visualizzato un riquadro GridView Tasks, come mostrato nella Figura 15.19. È possibile accettare i valori predefiniti nel riquadro. Tuttavia utilizzando la finestra Properties, rinominare il controllo GridView (utilizzando la sua proprietà ID) in dgvData per mantenere il suo nome coerente con il controllo DataGridView nell'applicazione Windows, perché si sta cercando di portare il codice dall'applicazione Windows all'applicazione Web.

Individuazione del database sul server Web

GridView è l'unico controllo che verrà aggiunto al form Web. Non esiste un'applicazione Web equivalente al controllo OpenFileDialog utilizzato nell'applicazione Windows del Capitolo 14.

Inoltre non si desidera che l'utente selezioni il percorso del database. In un'applicazione Windows spesso il database può trovarsi sul computer dell'utente; di conseguenza è logico che l'utente individui e selezioni il file di database utilizzando il controllo OpenFileDialog. Al contrario, in un'applicazione Web, il database non si trova sul computer dell'utente ma su un server Web su Internet. Per motivi di sicurezza, all'utente non deve essere concesso di scorrere nei file sul server Web come può fare con i file del proprio computer; l'applicazione Web deve invece specificare dove si trova il file di database.

Spesso il database si trova in una sottocartella dell'applicazione Web per facilitare l'operazione di individuazione mediante il codice, come descritto di seguito. Per impostazione predefinita l'applicazione ASP.NET creata da Visual Studio 2005 ha una sottocartella chiamata App_Data, probabilmente l'abbreviazione di Application Data (dati dell'applicazione). Copiare il file nwind.mdb (o Northwind.mdb) nella cartella App_Data dal punto in cui è stato salvato il file nwind.mdb durante la creazione dell'applicazione di database Windows nel Capitolo 14.

Una volta individuato il database nell'hard disc nei file dell'applicazione Web, l'operazione restante è l'individuazione del database nel codice.

Come descritto nel Capitolo 14, la proprietà ConnectionString dell'oggetto OleDbConnection richiede il percorso e il nome del file di database. Nell'ap-

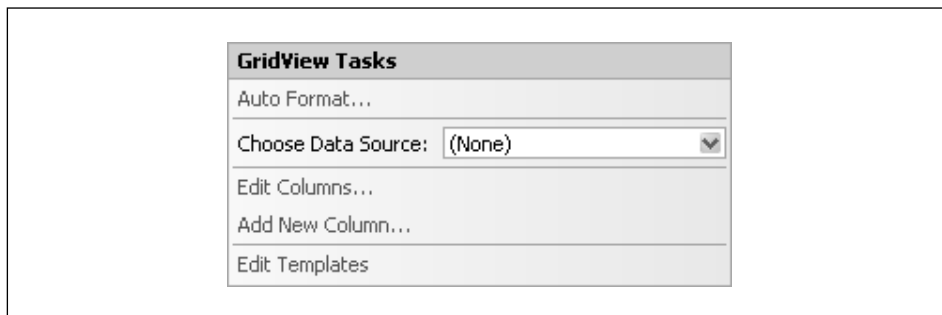


Figura 15.19 Riquadro GridView Tasks.

plicazione di database Windows del Capitolo 14, il percorso e il nome del file di database (rappresentati dalla variabile String `strFile`) sono stati ottenuti dalla proprietà `FileName` del controllo `OpenFileDialog`:

```
Dim myConn As New OleDbConnection
Dim dr As DialogResult
dr = dlgOpen.ShowDialog()
If dr = DialogResult.OK Then
    Dim strFile As String = dlgOpen.FileName
    myConn.ConnectionString = "Provider = " & _
        "Microsoft.Jet.OLEDB.4.0;Data Source=" & strFile & ";"
```

Non è possibile ottenere il percorso e il nome del file di database allo stesso modo in questa applicazione Web, perché non esiste alcun controllo `OpenFileDialog`. Tuttavia è nota la posizione del file di database, nella sottocartella `app_data` dell'applicazione Web. Quindi il percorso virtuale del database è `http://localhost:1040/WebSite1/app_data/nwind.mdb`.

Tuttavia la proprietà `ConnectionString` richiede il percorso fisico, non virtuale. In questo caso il percorso fisico è noto, perché il file di database si trova sul proprio computer. Tuttavia quando si lavora con server Web remoti, il percorso fisico può non essere sempre noto, oppure l'amministratore del server Web può modificarlo. Di conseguenza è necessario essere in grado di tradurre il percorso virtuale in un percorso fisico.

La classe `HttpServerUtility`, a cui si può anche fare riferimento mediante la classe `Server`, dispone di un metodo `MapPath` che restituisce il percorso fisico del file corrispondente al percorso virtuale sul server Web. La dichiarazione indicata di seguito assegna la variabile String `strFile` al percorso fisico del file di database:

```
Dim strFile As String = Server.MapPath _
    ("app_data\nwind.mdb")
```

NOTA *Può essere necessario modificare il riferimento a `nwind.mdb` in `Northwind.mdb`, se quest'ultimo è il nome di file sul proprio computer.*

Il metodo `MapPath` inizia mappando il percorso fisico corrispondente al percorso virtuale sull'applicazione Web, `http://localhost:1040/WebSite1`. L'argomento viene quindi aggiunto a tale percorso fisico. Il metodo restituisce il percorso fisico corrispondente al percorso virtuale completo sul file di database, `http://localhost:1040/WebSite1/app_data_nwind.mdb`.

Di conseguenza il codice del Capitolo 14 viene sostituito dal codice indicato di seguito:

```
Dim myConn As New OleDbConnection
Dim strFile As String = Server.MapPath _
    ("app_data\nwind.mdb")
myConn.ConnectionString = "Provider = " & _
    "Microsoft.Jet.OLEDB.4.0;Data Source=" & strFile & ";"
```

Aggiunta di codice

Il passaggio successivo è la scrittura di codice. Per effettuare tale operazione, passare alla vista codice del form Web.

In primo luogo si importano gli spazi di nomi System.Data e System.Data.OleDb per lo stesso motivo per cui sono stati importati nel Capitolo 14. Le due dichiarazioni Imports devono essere inserite immediatamente prima del nome di classe:

```
Imports System.Data
Imports System.Data.OleDb
Partial Class _default
```

Quindi, come nel Capitolo 14, tutto il codice va inserito nell'evento Load, questa volta della pagina Web. Questa procedura di evento appartiene all'oggetto Page, che rappresenta il form Web.

Per creare una procedura di evento, analogamente ai form Windows, scegliere (Page Events) dalla casella di riepilogo a discesa di sinistra. Scegliere quindi l'evento (in questo caso Load) dall'elenco a discesa di destra. In questo modo si crea uno stub di procedura di evento. Quindi scrivere il codice in modo che la procedura di evento Page_Load sia come indicato di seguito:

```
Private Sub Page_Load (ByVal sender As System.Object, _
ByVal e As System.EventArgs) Handles Me.Load
    Dim myConn As New OleDbConnection
    Dim strFile As String = Server.MapPath _
        ("app_data\nwind.mdb")
    myConn.ConnectionString = "Provider = " & _
        "Microsoft.Jet.OLEDB.4.0;Data Source=" & strFile & ";"
    myConn.Open()
    Dim myCMD As New OleDbCommand
    myCMD.CommandText = "SELECT CustomerID, " & _
        "ContactTitle, CompanyName, ContactName FROM Customers"
    myCMD.Connection = myConn
    Dim myAdapter As New OleDbDataAdapter
    myAdapter.SelectCommand = myCMD
    Dim ds As New DataSet
    ds.Clear()
    myAdapter.Fill(ds, "Customers")
    dgvData.DataSource = ds
    dgvData.DataMember = "Customers"
    dgvData.DataBind()
End Sub
```

Questo codice presenta solo due differenze sostanziali rispetto al corrispondente codice nella procedura di evento Form Load del Capitolo 14. La prima è l'utilizzo del metodo MapPath, come descritto nel paragrafo "Individuazione del database sul server Web". La seconda è l'ultima dichiarazione, la chiamata

al metodo `DataBind` di `GridView`. Questo metodo viene utilizzato molto nelle applicazioni Web per collegare i dati da un'origine di dati (in questo caso un `DataSet`) a un controllo (in questo caso `GridView`).

ATTENZIONE *Se non si chiama il metodo `DataBind`, l'applicazione Web viene eseguita senza errori, ma il `GridView` è vuoto, perché non è stato collegato all'origine di dati.*

Eeguire l'applicazione Web dal menu Debug, come per un'applicazione Windows. Viene visualizzata la pagina Web, con il `GridView` compilato con informazioni, come mostrato in precedenza nella Figura 15.16. Al termine chiudere la pagina Web utilizzando il pulsante di chiusura per chiudere l'applicazione.

15.6 Conclusione

Naturalmente l'argomento delle applicazioni Web non è stato esaurito nel presente capitolo: all'argomento sono dedicati interi libri e corsi. Tuttavia il presente capitolo offre una panoramica relativa alla creazione di un'applicazione Web funzionante, che visualizza informazioni provenienti da un database.

Questo è l'ultimo capitolo del libro, ma non dovrebbe essere l'ultimo capitolo dell'apprendimento di Visual Basic 2005. L'autore spera invece che il presente volume abbia offerto basi utili per un ulteriore apprendimento.

15.7 Quiz

1. Cos'è ASP.NET?
2. Cos'è un URL?
3. Cos'è HTTP?
4. Cosa indica l'estensione `.aspx`?
5. Qual è la differenza tra un percorso di pagina Web virtuale e fisico?
6. Quale modello di progetto può essere utilizzato per la creazione di un'applicazione Web?
7. Qual è il controllo Web corrispondente al controllo `DataGridView` utilizzato nelle applicazioni Windows?
8. Qual è il metodo della classe `HttpServerUtility` che restituisce il percorso di file fisico corrispondente al percorso virtuale specificato su un server Web?
9. Qual è il nome della classe che rappresenta l'equivalente nelle applicazioni Web della classe `Form` in un'applicazione Windows?
10. Qual è il metodo di `GridView` che deve essere chiamato in un'applicazione Web in modo che `GridView` non sia vuoto?