

ISTITUTO TECNICO INDUSTRIALE

G. M. ANGIOY

SASSARI



CORSO DI PROGRAMMAZIONE

INTERFACCIA PER BASI DI DATI

DISPENSA 21.01

[21-01_Access_\[ver_09\]](#)



Questa dispensa è rilasciata sotto la licenza Creative Common CC BY-NC-SA. Chiunque può copiare, distribuire, modificare, creare opere derivate dall'originale, ma non a scopi commerciali, a condizione che venga riconosciuta la paternità dell'opera all'autore e che alla nuova opera vengano attribuite le stesse licenze dell'originale.

Versione del: **07/11/2015**

Revisione numero: **09**

Prof. Andrea Zoccheddu
Dipartimento di Informatica

**DIPARTIMENTO
INFORMATICA E TELECOMUNICAZIONI**





INTERFACCE PER BASI DI DATI

BASI DI DATI

DATABASE E ACCESS

CONCETTO DI BASE DI DATI



http://it.wikipedia.org/wiki/Base_di_dati#Generalit.C3.A0

Definizione

In informatica, il termine base di dati, database o banca dati (a volte abbreviato con la sigla DB), indica un sistema di gestione di un insieme organizzato di dati. Le informazioni contenute in un database sono strutturate e collegate tra loro secondo un particolare modello logico scelto dal progettista del database.

DATABASE ACCESS



http://it.wikipedia.org/wiki/Microsoft_Access#Formato_dati

Definizione

Microsoft Access è un pacchetto applicativo distribuito da Microsoft con cui realizzare e gestire basi di dati e relative interfacce. In effetti Microsoft Access è un'interfaccia software verso l'engine jet (motore di base di dati) per la gestione di basi di dati di tipo relazionale (RDBMS) che integra nativamente in sé un modulo per lo sviluppo rapido di applicativi (RAD, Rapid Application Development) gestionali.

A differenza di altri ambienti di sviluppo, in Access un unico file comprende tutti gli elementi utilizzabili per lo sviluppo di applicazioni complete: tabelle, query, maschere, report, macro, pagine e moduli. È comunque possibile, con tutte le versioni, progettare applicazioni nelle quali si mantenga la separazione fisica tra tabelle di dati (Back-End o BE) ed i restanti elementi (Front-End o FE). Un database realizzato con Access è un file che mantiene insieme tutte queste informazioni.

TABELLE IN ACCESS



http://it.wikipedia.org/wiki/Microsoft_Access#Formato_dati

Definizione

Un database di Access contiene molte tabelle ciascuna che memorizza registrazioni (righe) composte di molti elementi (campi o colonne) ciascuna rappresentante una singola informazione strutturata.

Le tabelle sono i contenitori dove vengono memorizzati i dati; ogni riga della tabella rappresenta una singola informazione, suddivisa in campi; ogni colonna (o campo) è un frammento di informazione di ogni singolo record.



Esempio

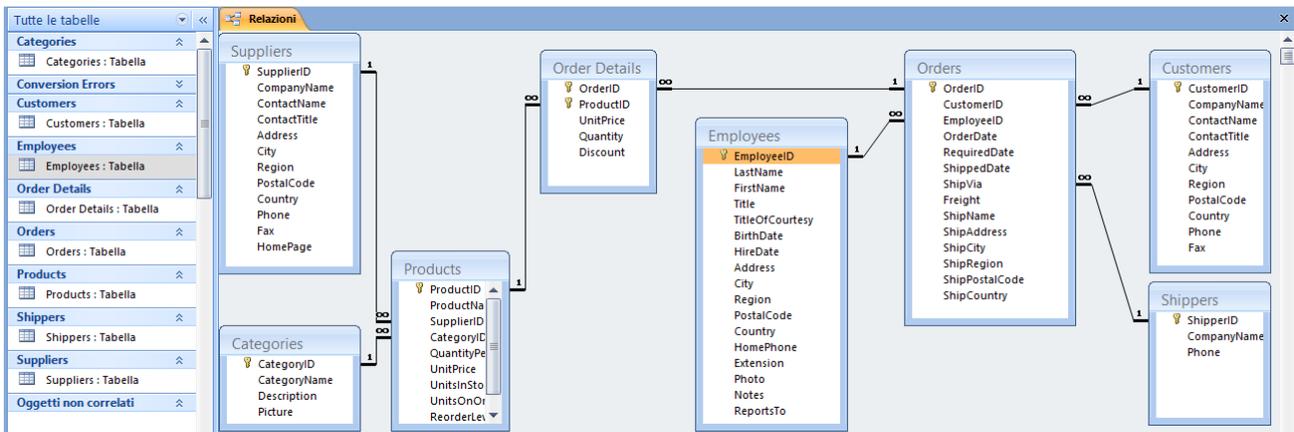
Un esempio di tabella con Access è proposta nella figura qui a lato.

Studenti								
IDS	Matricola	Cognome	Nome	Età	Sesso	ID_Classe		
1	12345	Sale	Rosa	17	F	1		
2	12346	Carta	Bianca	18	F	3		
3	12348	Marino	Pino	19	F	1		
4	12347	Piu	Mino	17	M			
5	12349	Marino	Pino	18	M	1		
6	12351	Carta	Rosa	17	F	3		
7	12350	Piano	Remo	16	M	3		
8	12352	Piano	Guida	18	F	1		

DATABASE DI RIFERIMENTO

UN ESEMPIO DI BASE DI DATI

Come database di riferimento useremo il NorthWind, un classico esempio proposto da Microsoft; è un DB di una fittizia azienda commerciale che prevede tabelle standard come Fornitori, Prodotti e Categorie, Dettagli degli ordini, Ordini, Impiegati, Clienti e Distributori:



È possibile scaricare il file di database dal seguente link:



<http://ccsmsoffice.wikispaces.com/file/links/Northwind.accdb>

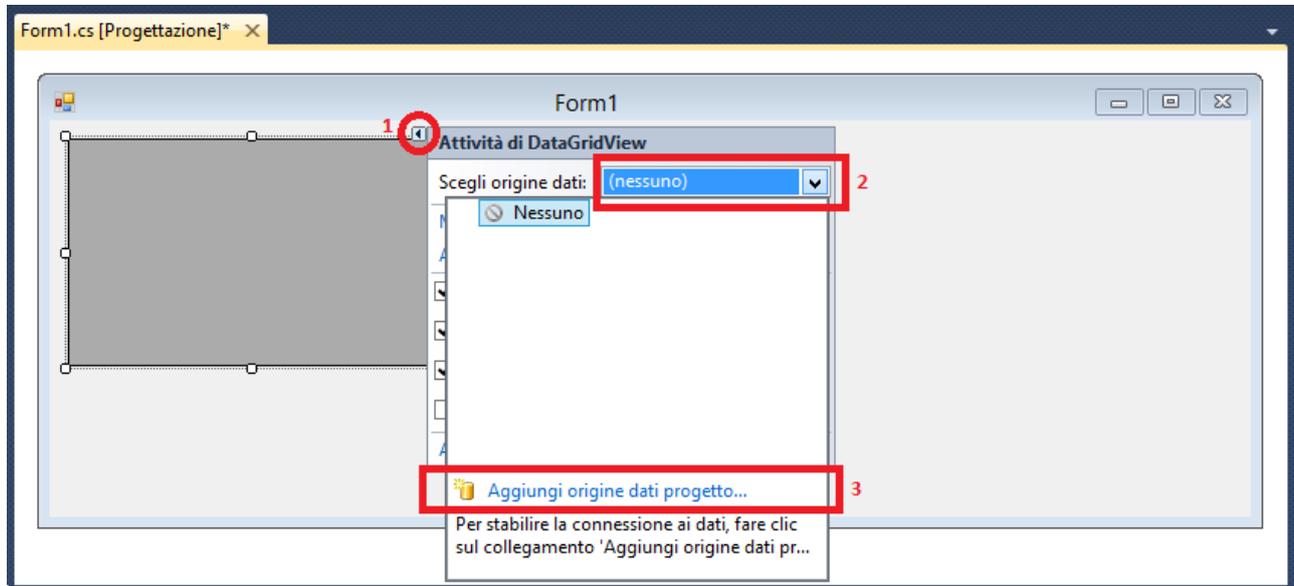
e lo si può aprire con l'applicazione Access fornita anche insieme ad alcune versioni del pacchetto Office.



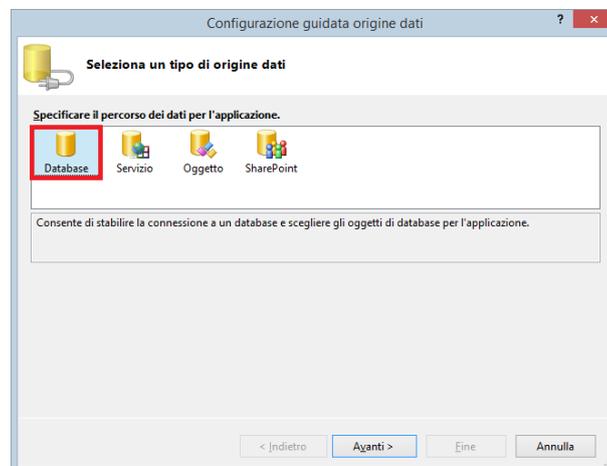
UNA PRIMA INTERFACCIA

PROGETTO GUIDATO

- Aprire un nuovo progetto con Visual Studio in linguaggio C#
- Inserire un DataGridView nel Form1 e aprire l'opzione di scelta origine dati (vedi figura) che probabilmente all'inizio è impostata su **(nessuno)**:



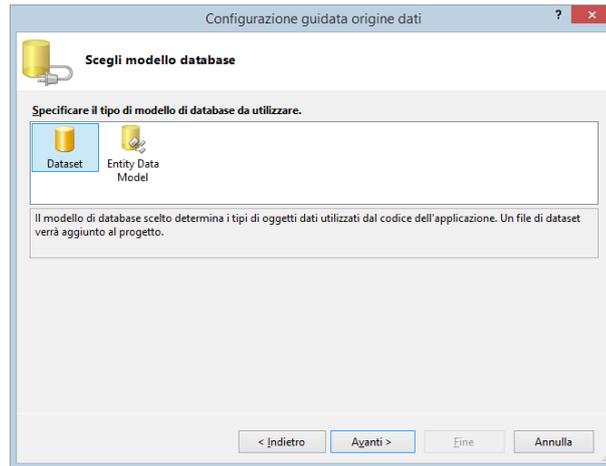
- Scegliere l'opzione «**Aggiungi origine dati progetto ...**»:
- Compare una finestra di dialogo che propone diversi tipi di origine dati;
- si scelga Database e poi si preme il pulsante Avanti





• Compare una seconda finestra di dialogo che propone diversi modelli di basi di dati;

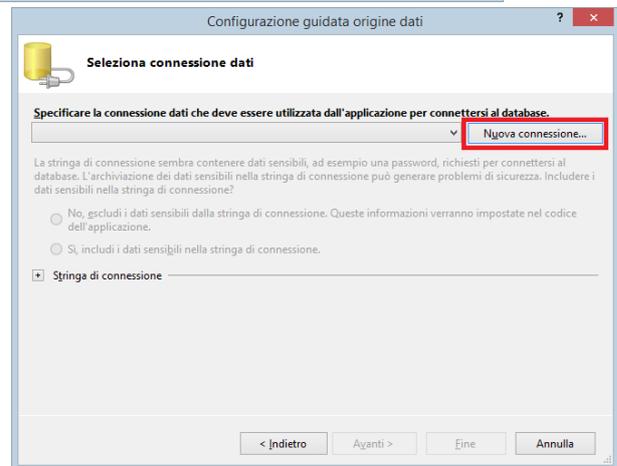
• si scelga Dataset e poi si preme il pulsante Avanti



• Compare una terza finestra di dialogo che richiede di specificare la connessione dei dati a cui fare riferimento;

• All'inizio è probabile che la connessione sia vuota; questa sarà una stringa che rappresenta una sorta di comando di connessione

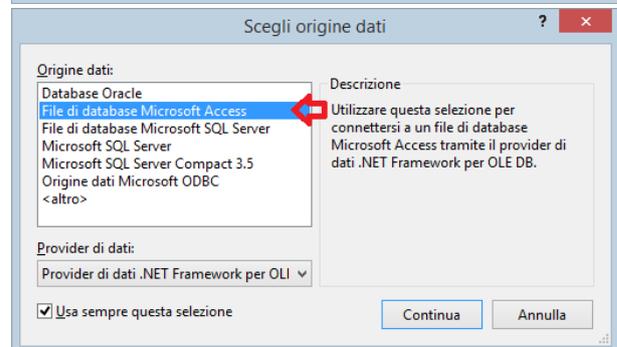
• Si preme il pulsante **Nuova connessione** che compare sulla destra



• Compare una piccola finestra di dialogo che elenca i principali tipi di origine dei dati

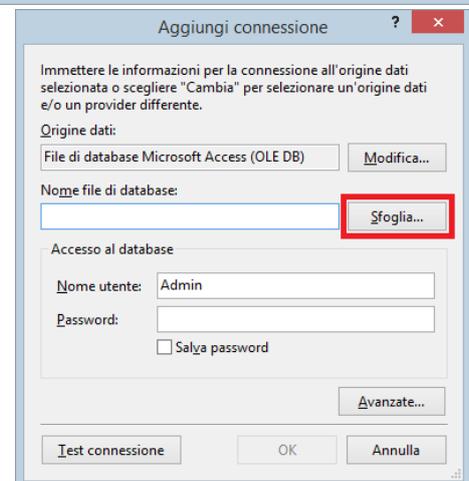
• Si selezioni dall'elenco l'opzione File di database Microsoft Access

• Si preme il pulsante **Continua** per proseguire con la configurazione guidata



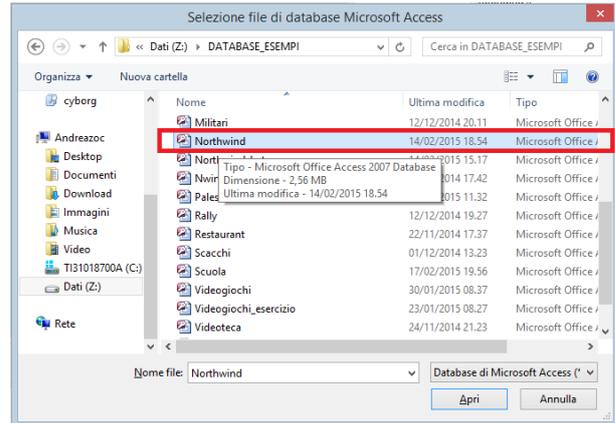
• Compare una minuta finestra di dialogo che ripropone il tipo di connessione scelta e che richiede il nome del file in cui è contenuto il database di Access

• Si preme il pulsante **Sfogli** per reperire e scegliere il file di Access che vogliamo collegare al Form1

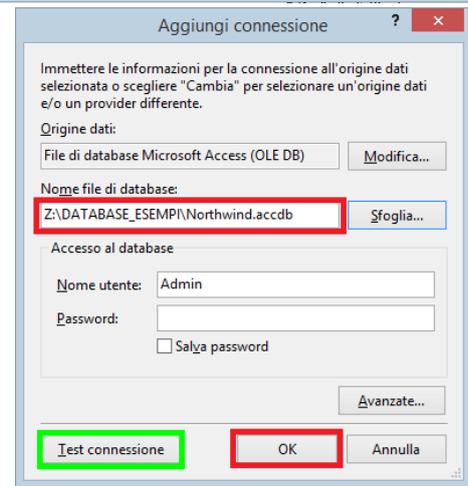




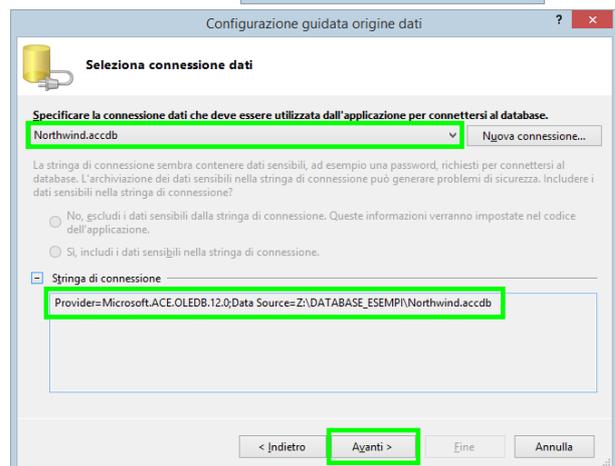
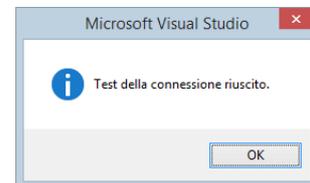
- Compare la classica finestra di esplorazione dei file di Windows con cui navigare tra le risorse del computer per individuare e selezionare un file Access
- Muoversi nel file system per trovare la cartella contenente il file desiderato e selezionarlo; nel nostro esempio il file è denominato **Northwind.accdb**
- Si preme infine il pulsante **Apri** per confermare l'apertura del file di Access che vogliamo collegare al Form1



- La procedura ritorna alla finestra di dialogo di connessione in cui è stata aggiunto il nome completo del file da aprire (percorso, nome e estensione)
- L'accesso al database può essere necessario se il database è stato protetto; in tal caso potrebbe essere necessario specificare il nome utente e la password; nel nostro esempio è inutile
- Possiamo invece provare il test di connessione per verificare il funzionamento; si preme il pulsante **Test connessione** e si attende la finestra di dialogo di conferma (vedi figura)
- Infine si preme **OK** per proseguire

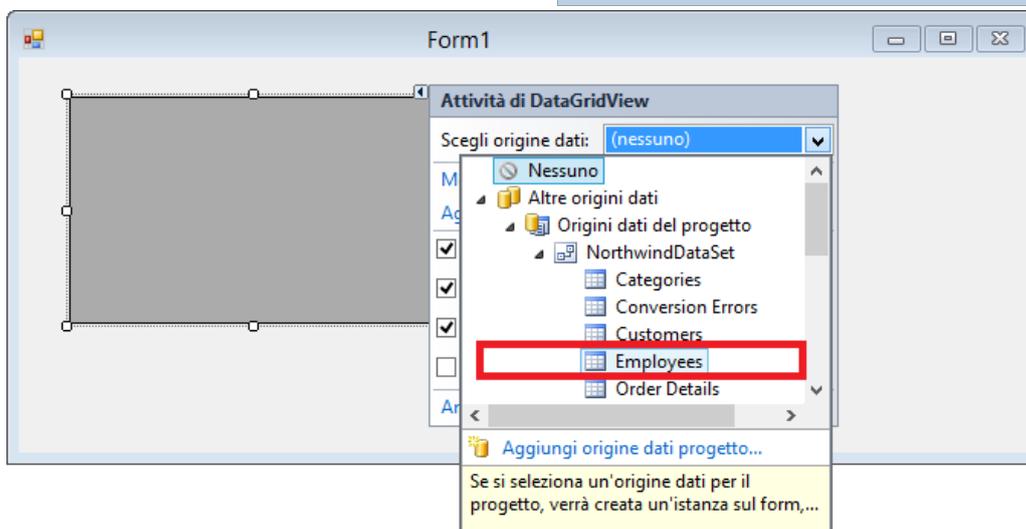
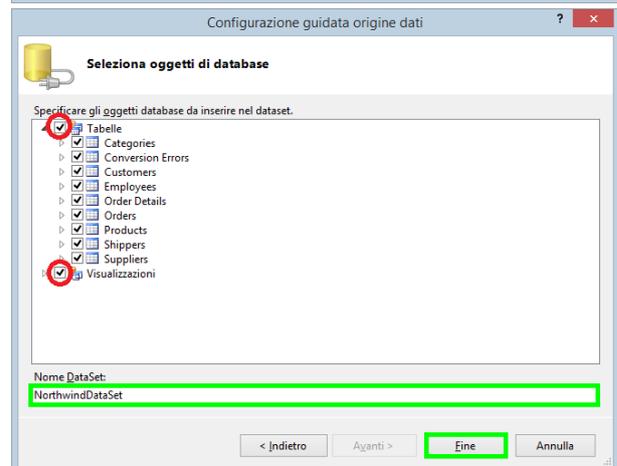
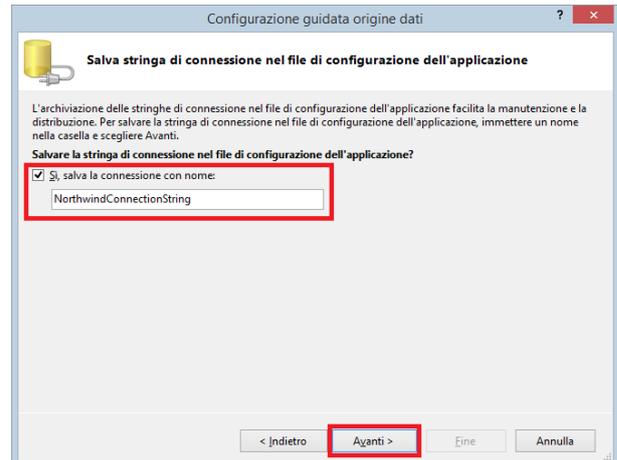
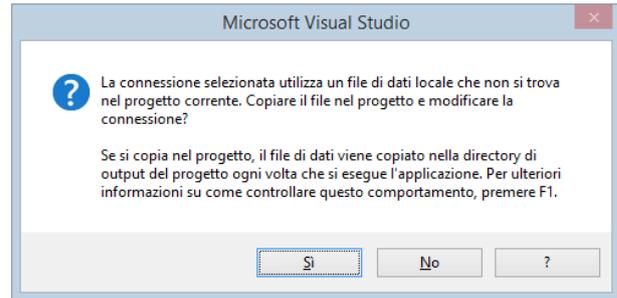


- Si ritorna alla procedura guidata dove sono comparsi il nome del file **Northwind.accdb** e la stringa di connessione riportata in basso
- Questa stringa di connessione ci tornerà utile in seguito e la ritroveremo nel codice del programma; per il momento non è necessario analizzarla
- Per proseguire premere il pulsante **Avanti**



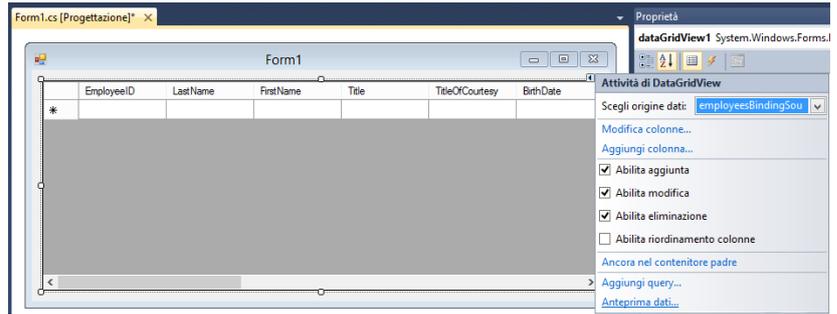


- Appare quindi una finestra di dialogo che chiede se effettuare una copia del database nella cartella di progetto
- In pratica ci chiede se vogliamo usare il database originale oppure una sua copia
- Per il momento scegliamo di lavorare su una copia per evitare di danneggiare il database originale e si fa clic su **Si**
- La procedura prosegue e compare una finestra di dialogo che chiede se salvare la stringa di connessione nel programma
- In pratica ci chiede se vogliamo dare un nome a questa connessione; lasciamo spuntata la casella che conferma il nome e premiamo il pulsante Avanti
- La procedura procede con una finestra di dialogo che chiede quali oggetti vogliamo inserire nel database
- Per il momento non approfondiamo questo aspetto, ma ci limitiamo a osservare che le tabelle sono quasi sempre necessarie
- Spuntiamo entrambe le scelte e premiamo Fine
- Finalmente torniamo al progetto di Visual Studio

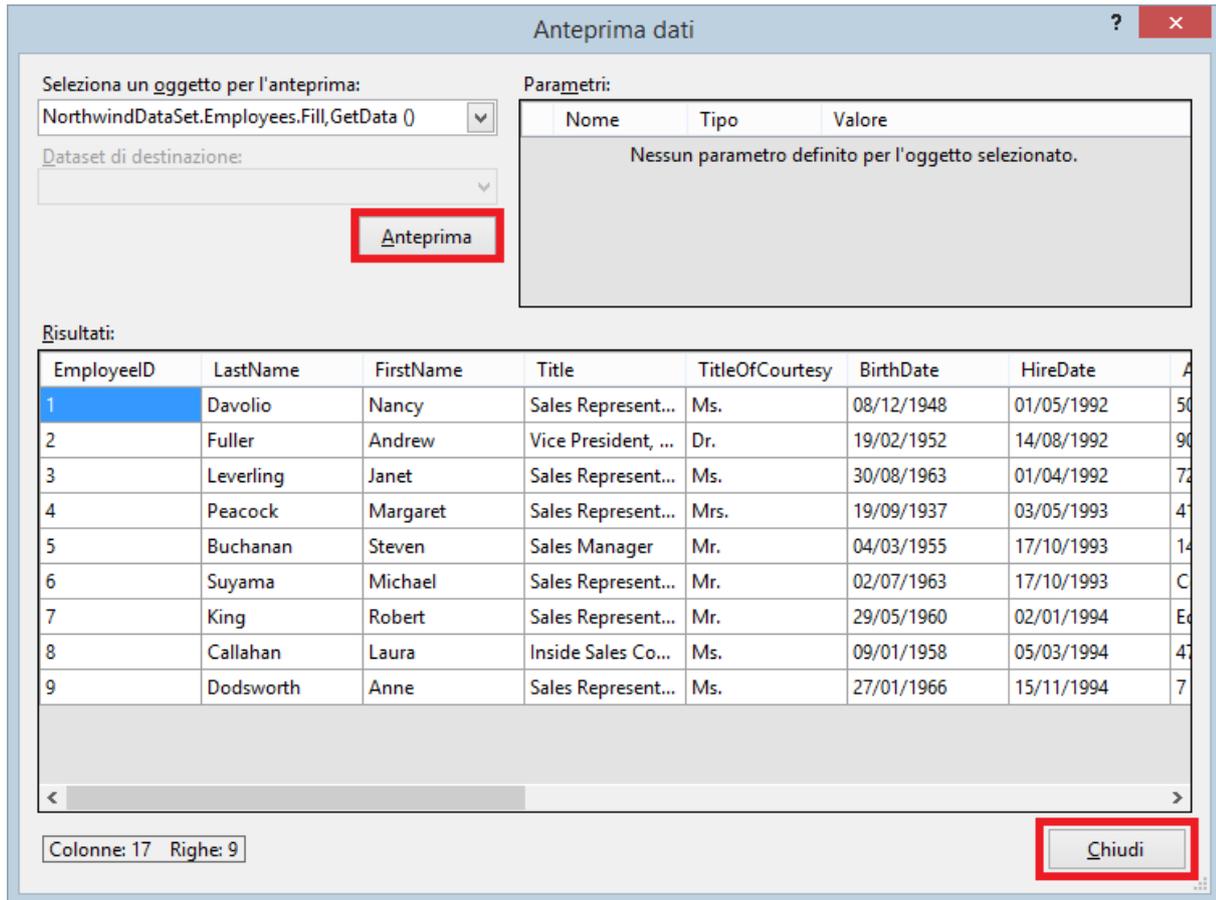




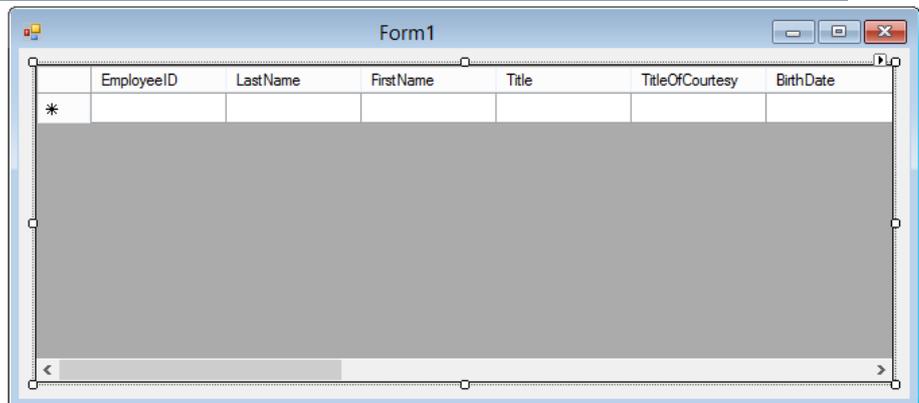
- La casella a discesa propone un nodo per il DataSet NorthwindDataSet e in cascata i suoi oggetti
- Selezioniamo la tabella Employees (impiegati) e poi scegliamo anche Anteprima dati ...



- Compare la finestra seguente, dove dobbiamo premere Anteprima



- Dopo aver chiuso l'anteprima, aggiustiamo il Form1 disponendo la griglia per visualizzate tutte le colonne
- Il Form1 assume l'aspetto della figura a lato





- Adesso avviamo il debug del progetto e possiamo vedere il risultato finale
- Si provi a modificare qualche campo, **tranne** i dati della prima colonna (**ID_Opera**)

EmployeeID	LastName	FirstName	Title	TitleOfCourtesy	BirthDate
1	Davolio	Nancy	Sales Represent...	Ms.	08/12/1948
2	Fuller	Andrew	Vice President, S...	Dr.	19/02/1952
3	Leverling	Janet	Sales Represent...	Ms.	30/08/1963
4	Peacock	Margaret	Sales Represent...	Mrs.	19/09/1937
5	Buchanan	Steven	Sales Manager	Mr.	04/03/1955
6	Suyama	Michael	Sales Represent...	Mr.	02/07/1963
7	King	Robert	Sales Represent...	Mr.	29/05/1960
8	Callahan	Laura	Inside Sales Coor...	Ms.	09/01/1958
9	Dodsworth	Anne	Sales Represent...	Ms.	27/01/1966

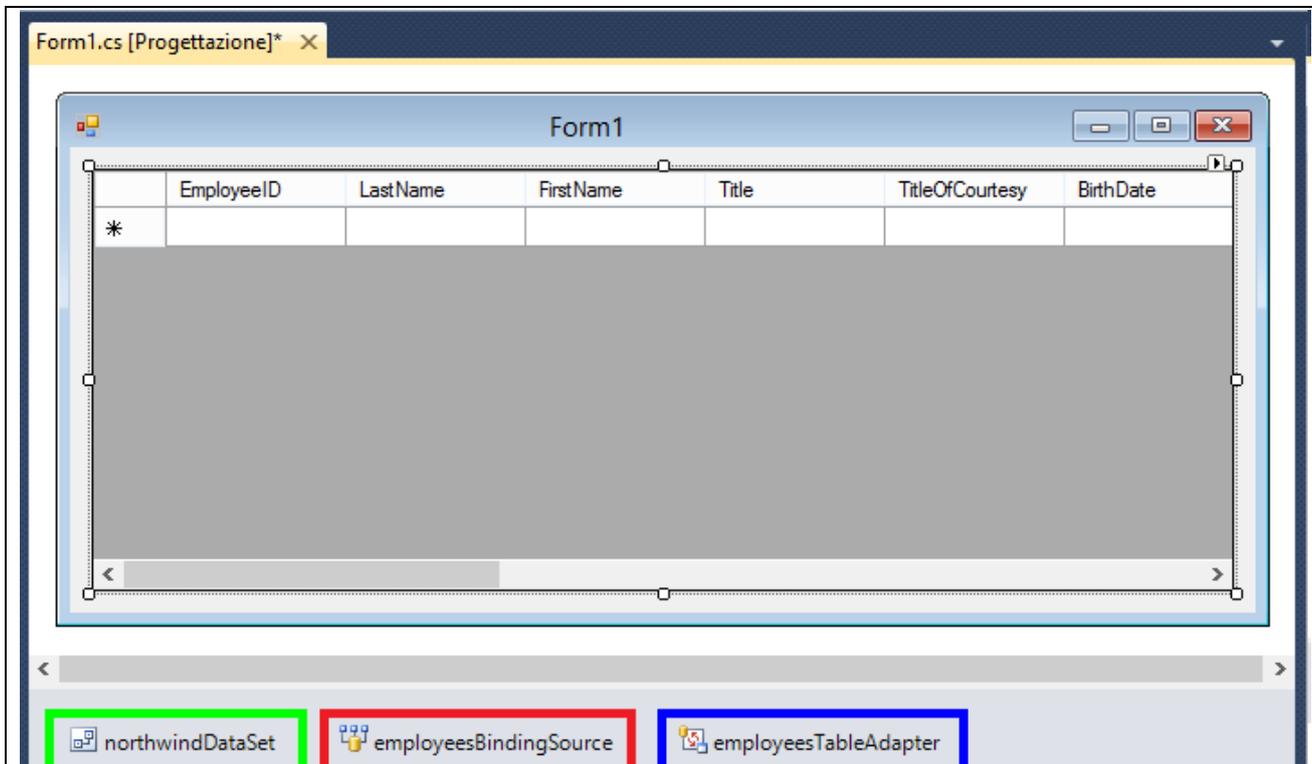
- Chiudere il progetto



ANALISI DEI COMPONENTI

COMPONENTI E DATABASE

Osservando il Form1 realizzato nella precedente esperienza possiamo notare che sono stati inseriti nel progetto alcuni componenti non visuali.



I componenti appaiono fuori dal Form1, posti nella area sottostante esso, e sono almeno tre:

Aspetto	Componente	Descrizione sintetica
northwindDataSet	DataSet	Contenitore di dati locale
employeesBindingSource	BindingSource	Origine dati per altri controlli
employeesTableAdapter	TableAdapter	

DATASET



<https://msdn.microsoft.com/it-it/library/8bw9ksd6.aspx>

I DataSet sono oggetti che contengono tabelle di dati in cui è possibile memorizzare temporaneamente i dati da utilizzare nell'applicazione. Se l'applicazione prevede che vengano utilizzati dei dati, è possibile caricarli in un DataSet. In questo modo viene fornita all'applicazione **una cache locale in memoria** dei dati da utilizzare.

I dati contenuti nel dataset possono essere utilizzati anche se l'applicazione viene disconnessa dal database. Le informazioni relative alle modifiche apportate ai dati del DataSet vengono conservate all'interno di quest'ultimo così che, quando l'applicazione verrà riconnessa, gli aggiornamenti potranno essere rilevati e inviati al database.



BINDINGSOURCE



[https://msdn.microsoft.com/it-it/library/vstudio/h974h4y2\(v=vs.100\).aspx](https://msdn.microsoft.com/it-it/library/vstudio/h974h4y2(v=vs.100).aspx)

Il componente BindingSource incapsula un'origine dati per l'associazione a controlli.

Il componente BindingSource ha due funzioni. Fornisce un livello di riferimento indiretto per l'associazione dei controlli in un form ai dati. Questa operazione viene eseguita associando il componente BindingSource all'origine dati e quindi associando i controlli presenti sul form al componente BindingSource. Tutte le ulteriori interazioni con i dati, inclusa l'esplorazione, l'ordinamento, il filtro e l'aggiornamento, vengono eseguite mediante chiamate al componente BindingSource.

In secondo luogo, il componente BindingSource può fungere da origine dati tipizzata in modo sicuro. L'aggiunta di un tipo al componente BindingSource con il metodo Add determina la creazione di un elenco di quel tipo.

TABLEADAPTER



<https://msdn.microsoft.com/it-it/library/bz9tthwx.aspx>

Gli oggetti TableAdapter consentono di stabilire una comunicazione tra l'applicazione e un database. In particolare, un oggetto TableAdapter consente di connettersi a un database e di eseguire query o stored procedure e quindi restituisce una nuova tabella di dati popolata con i dati ottenuti oppure riempie un oggetto DataTable esistente con i dati ottenuti. I TableAdapter vengono utilizzati inoltre per inviare i dati aggiornati dall'applicazione al database.

Gli utenti di versioni precedenti di Visual Studio possono considerare un oggetto TableAdapter come un oggetto DataAdapter con un oggetto connessione incorporato e la capacità di contenere più query. Ciascuna query aggiunta a un TableAdapter viene esposta come metodo pubblico che viene chiamato con la stessa semplicità con cui si chiama qualsiasi altro metodo o funzione su un oggetto.

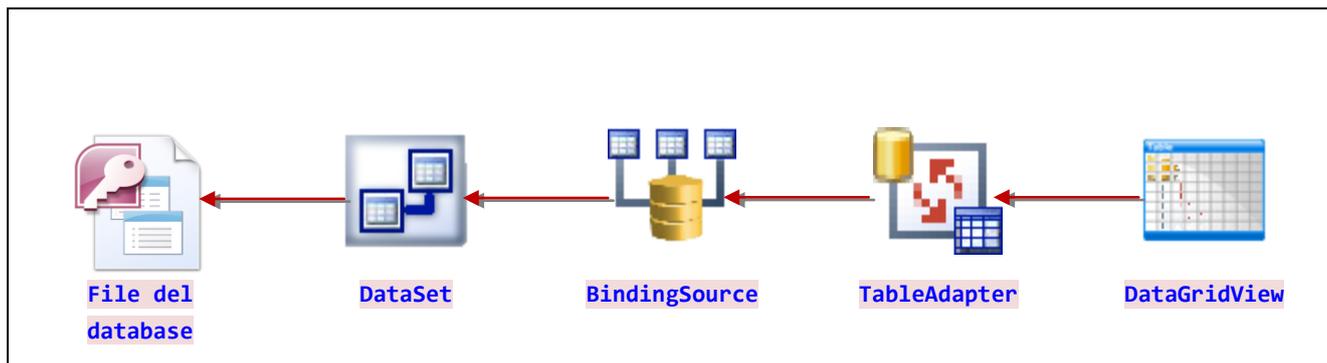
Oltre alla funzionalità standard di un oggetto DataAdapter, i TableAdapter offrono altri metodi tipizzati per l'incapsulamento di query che condividono uno schema comune con l'oggetto DataTable tipizzato associato. In altre parole, un TableAdapter può contenere un numero qualsiasi di query, a condizione che restituiscano dati conformi al medesimo schema.

Nella versione precedente di Visual Studio per le comunicazioni tra un'applicazione e un database venivano utilizzati gli adattatori dati ADO.NET. Mentre gli adattatori dati rimangono un componente principale di Provider di dati .NET Framework, gli oggetti TableAdapter sono componenti generati nella finestra di progettazione che migliorano la funzionalità degli oggetti DataAdapter. Gli oggetti TableAdapter contengono in genere metodi Fill e Update che consentono di recuperare e aggiornare dati in un database.

SCHEMA DI COLLEGAMENTO

In generale per costruire un'applicazione Visual Studio che si connette al database, si utilizza in primo luogo un DataSet che si connette al database effettivo; al dataset può essere agganciato un BindingSource che implementa una astrazione di uno specifico oggetto; infine si usa un TableAdapter che implementa un gestore di una tabella.

Lo schema risulta quindi essere simile alla seguente illustrazione:



Il **DataSet** è un componente non visuale che si connette al file del database; il componente permette di acquisire gli oggetti (tabelle e query) contenute nel database e di memorizzarne una copia dei dati per una elaborazione.

Il **BindingSource** è un componente non visuale che si connette al Dataset specifico; il componente si connette ad uno specifico oggetto, come una tabella, ne individua la struttura (per es. il numero e il tipo delle colonne), e può memorizzarne una copia dei dati.

Il **TableAdapter** è un componente non visuale che si connette al BindingSource specifico; il componente si connette ad una sorgente di dati e consente di adattarsi al formato tabellare, che può essere visualizzato con un DataGridView.

PROGETTO GUIDATO (FASE 1)

Da Internet scaricare il seguente database:



<http://ccsmsoffice.wikispaces.com/file/links/Northwind.accdb>

E salvarlo in una cartella accessibile



nell'esempio si farà riferimento alla cartella:

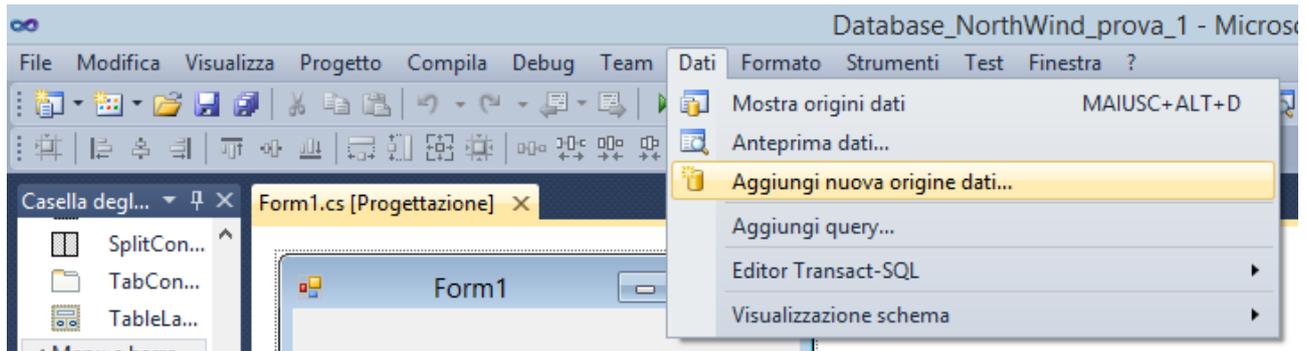
`Z:\DATABASE_ESEMPI`

Creare un nuovo progetto con Visual Studio in linguaggio C# e chiamarlo

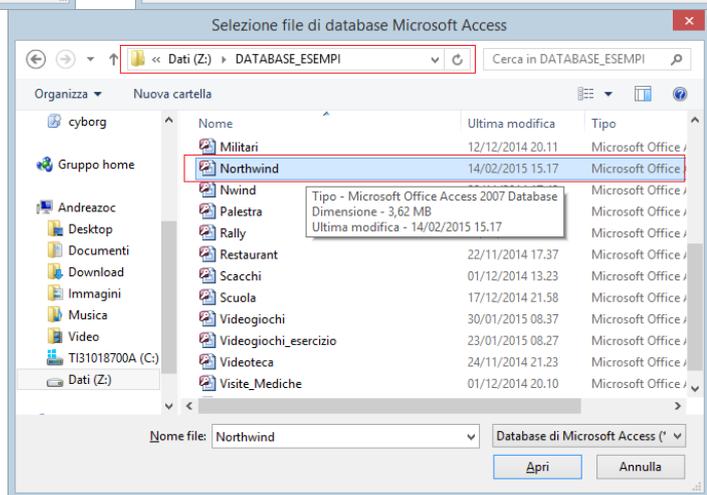
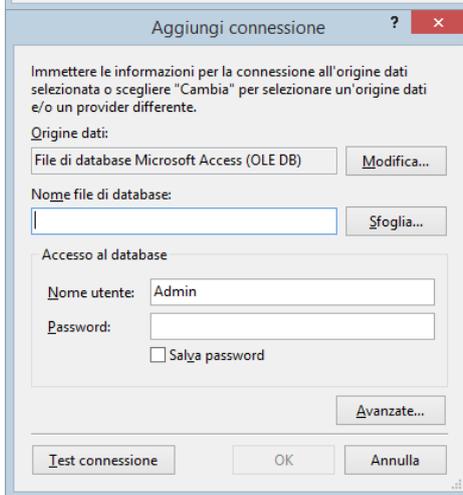
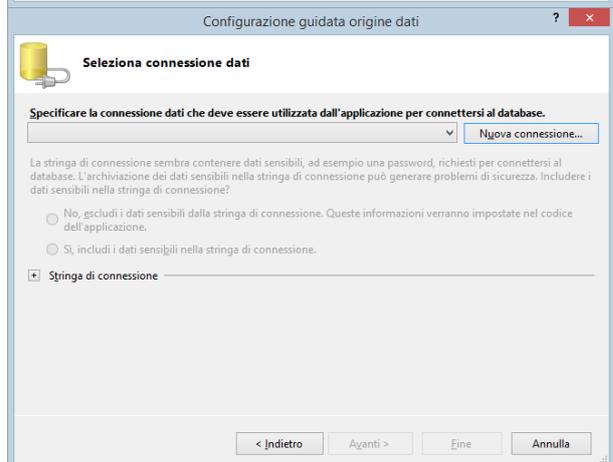
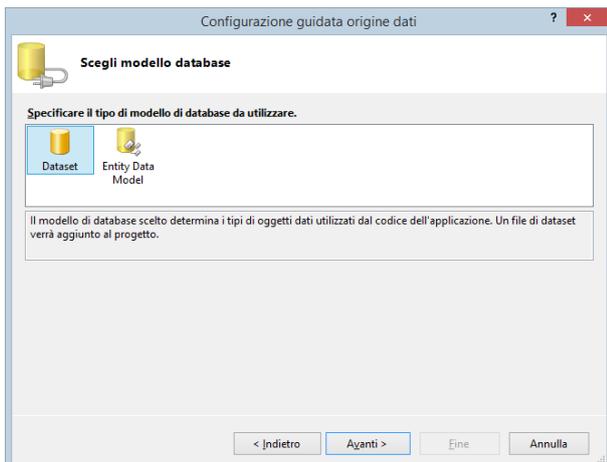
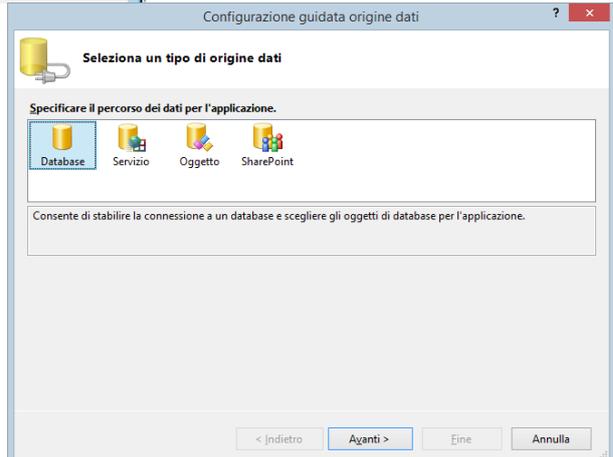


`Database_NorthWind_prova_1`

Dal menu Dati scegliere la voce Aggiungi nuova origine dati che serve per avviare la Configurazione guidata origine dati; dovrebbe comparire una finestra di dialogo come nella seguente figura:



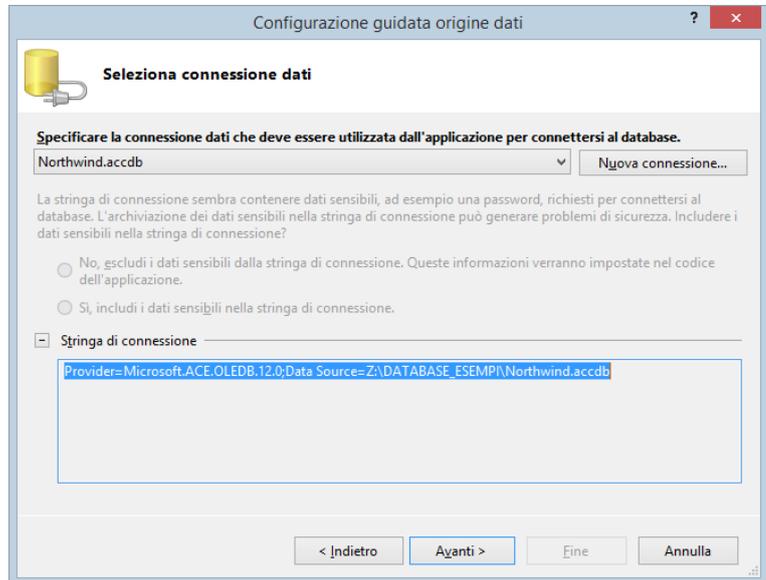
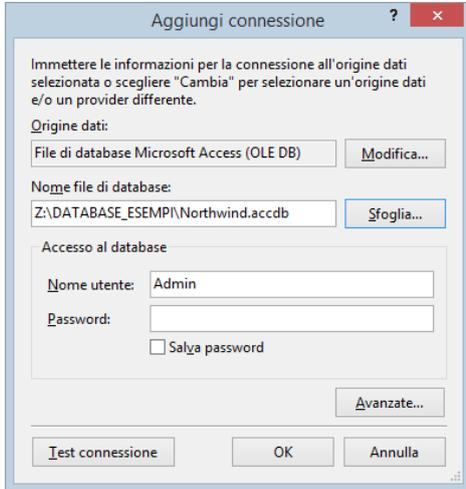
- Scegliere la prima icona (Database) e premere Avanti
- Scegliere DataSet e premere Avanti
- Scegliere poi Nuova connessione per cercare il database appena scaricato



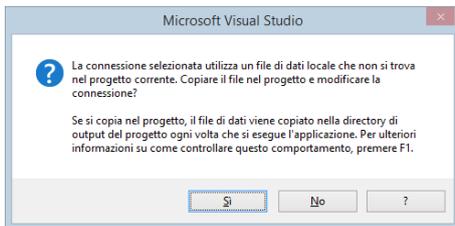


- Premere Sfoglia e cercare nella cartella il database NorthWind salvato prima
- Poi confermare col pulsante Apri

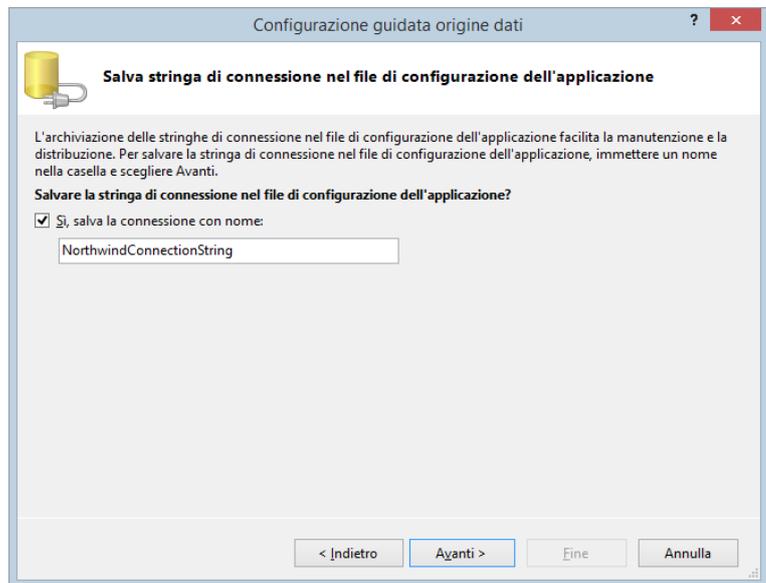
- La procedura guidata torna sulla finestra di dialogo per aggiungere la connessione



- Nella finestra di dialogo scegliere OK



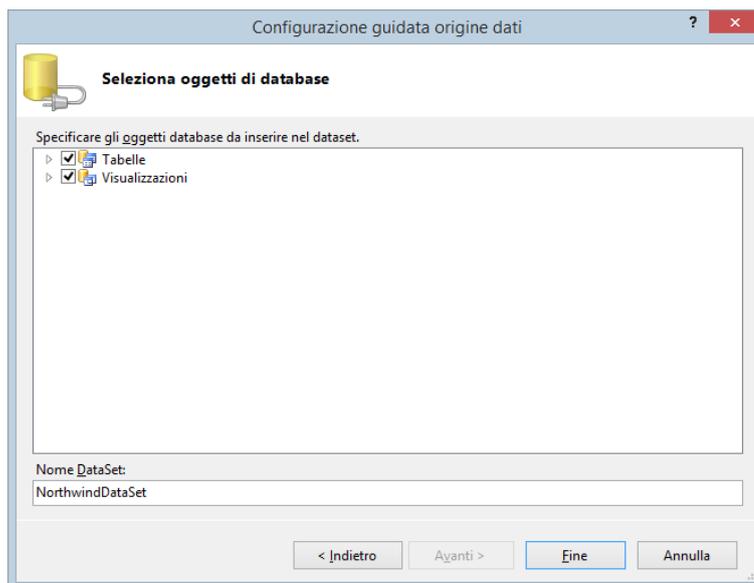
- Premere Avanti



- Per importare il database premere Sì
- Salvare la stringa di connessione appena creata col nome proposto dal sistema (lasciare il segno di spunta nella casella) e
- Premere Avanti per continuare la configurazione guidata dei dati



- Selezionare tabelle e visualizzazioni e lasciare il nome del DataSet proposto di default (NorthWindDataSet) che compare in basso
- Premere fine per terminare la configurazione guidata

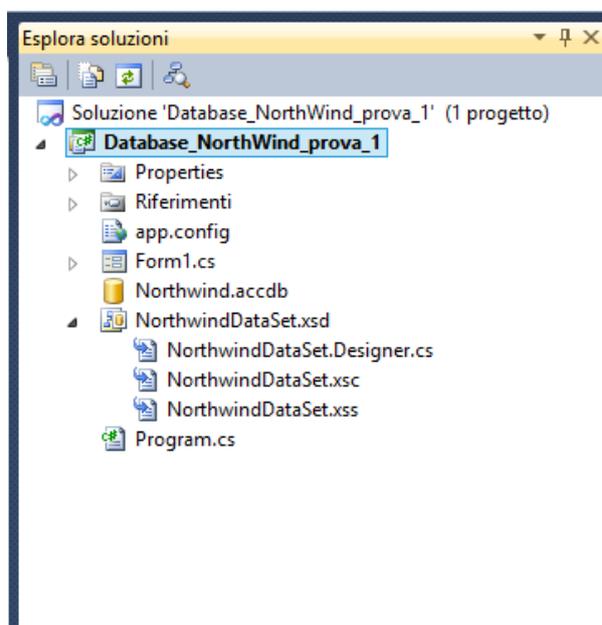


- Verificare dalla finestra Esplora soluzioni che sia stato creato il DataSet

- In Esplora Soluzioni si può osservare come il progetto si chiama Database_NorthWind_prova_1 (se è stato salvato con questo nome inizialmente)

- Nel progetto sono contenuti il Form1, un file NorthWind.accdb (il database MS Access), un elemento chiamato NorthWindDataSet.xsd che è una classe tipizzata derivata dalla classe DataSet e che al suo interno contiene gli elementi:

- Designer.cs (file C# di definizione)
- .xsc
- .xss



COSA È SUCCESSO?

Questo primo progetto guidato è servito a creare quel componente non visuale che si chiama DataSet tipizzato, la cui documentazione si trova al seguente link:



Fonte

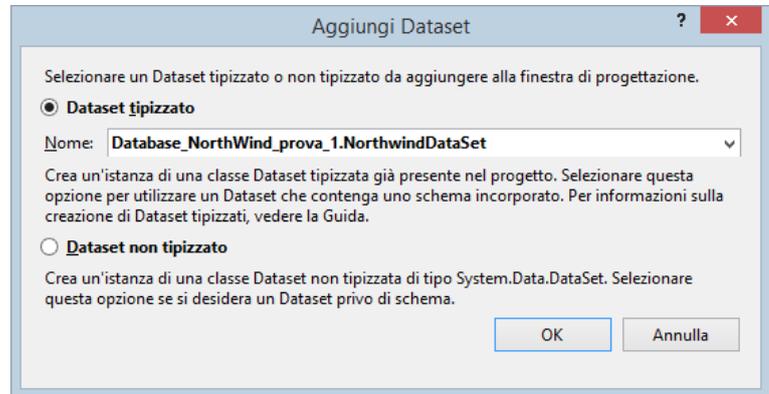
<https://msdn.microsoft.com/it-it/library/04y282hb.aspx>

Il DataSet tipizzato è una classe (tipo di dato oggetto) che è costruito a partire da un DataSet generico, ma arricchito con gli elementi del database a cui ci si vuole collegare; in pratica è un DataSet costruito in modo da collegarsi direttamente a quello specifico schema di dati del file di database a cui l'applicazione si collegherà.

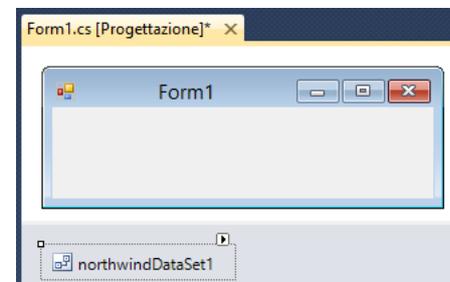


PROGETTO GUIDATO (FASE 2)

- Aprire il precedente progetto (se è chiuso)
- Inserire nel Form1 un componente DataSet (sezione Dati dei controlli) 
- Compare una finestra di dialogo come quella illustrata di lato
- L'opzione Dataset tipizzato (quella scelta di default) propone un nome di Dataset che coincide con quello appena creato
- Premere OK per confermare

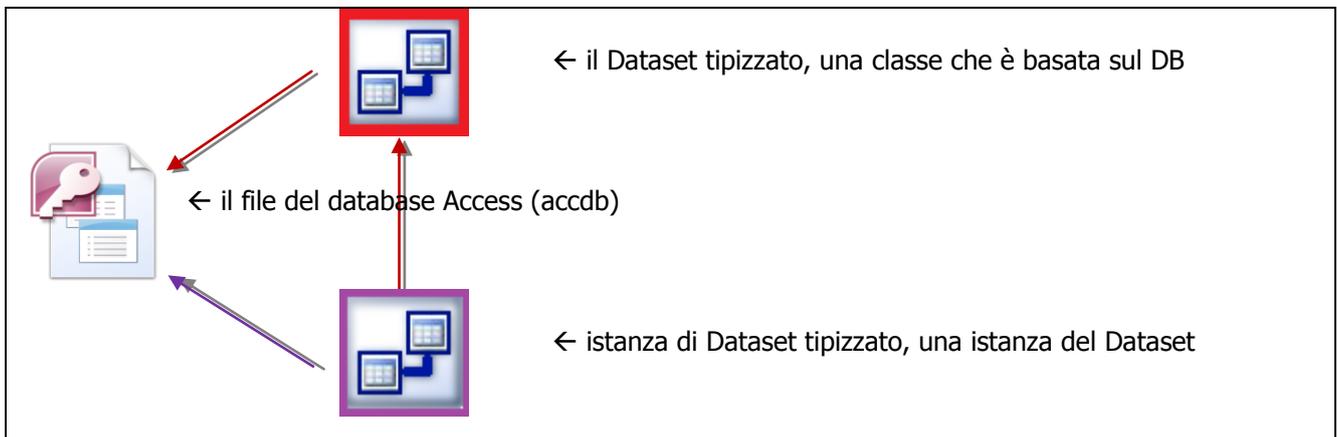


- Nel riquadro di progettazione compare, sotto il Form1, un componente denominato `northwindDataSet` che è una istanza del tipo `NorthWindDataSet` ovvero il `DataSet` creato nella fase 1



COSA È SUCCESSO?

Questo secondo progetto guidato è servito a creare una istanza di DataSet. Lo schema risulta quindi essere simile alla seguente illustrazione:



Il `File Access` è il database a cui ci si vuole collegare

Il `DataSet` tipizzato è una classe di oggetti, realizzata sullo schema del file del database

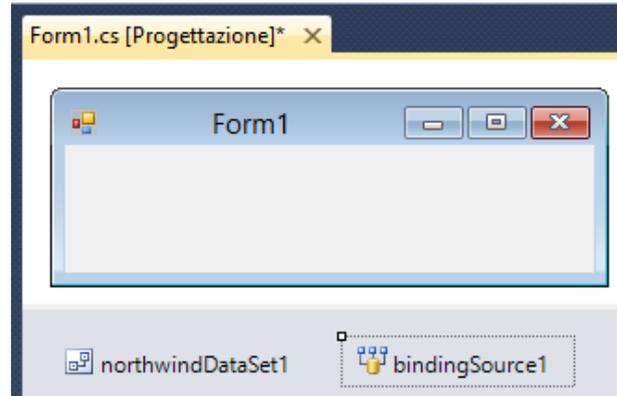
La `Istanza dataSet` è una istanza del DataSet che implementa una vera connessione al file DB

PROGETTO GUIDATO (FASE 3)

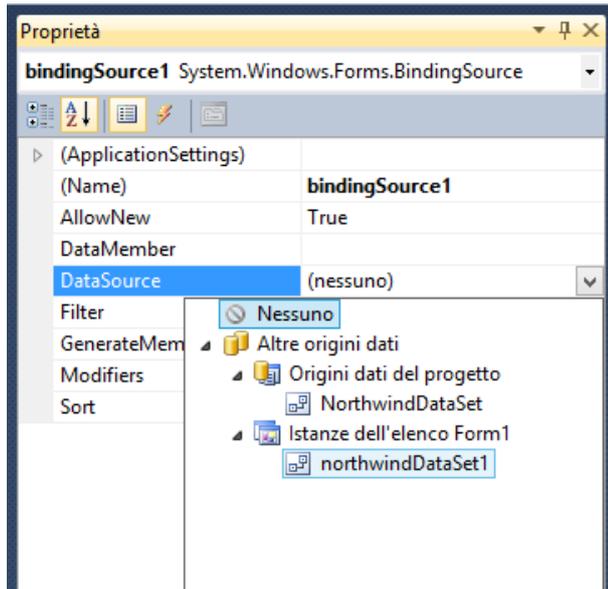
- Aprire il precedente progetto (se è chiuso)
- Inserire nel Form1 un componente `BindingSource` (sezione Dati dei controlli) 



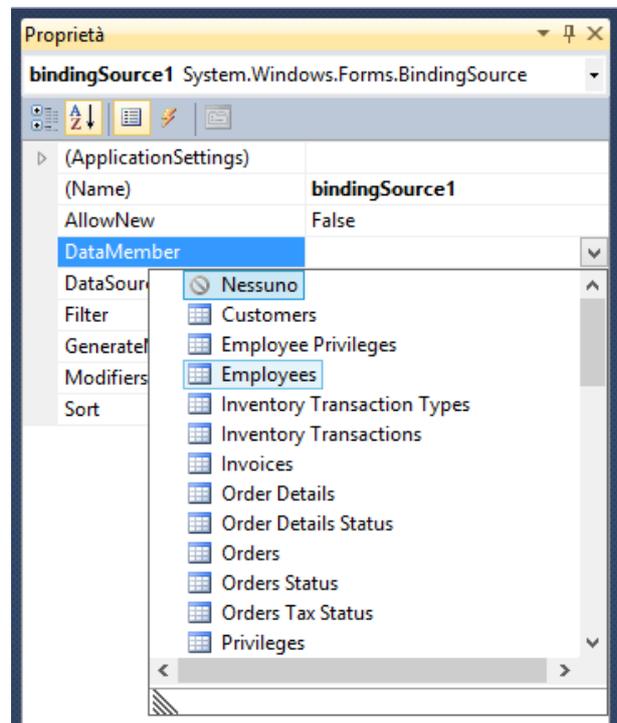
- Viene inserito un controllo non visuale `bindingSource1` sotto il `Form1`, accanto a `northwindDataSet1`
- Osservare le proprietà del controllo `bindingSource1`, nella apposita finestra
- Cercare la proprietà `DataSource` e aprire la casella a discesa e aprire l'albero delle scelte come nella figura qui sotto

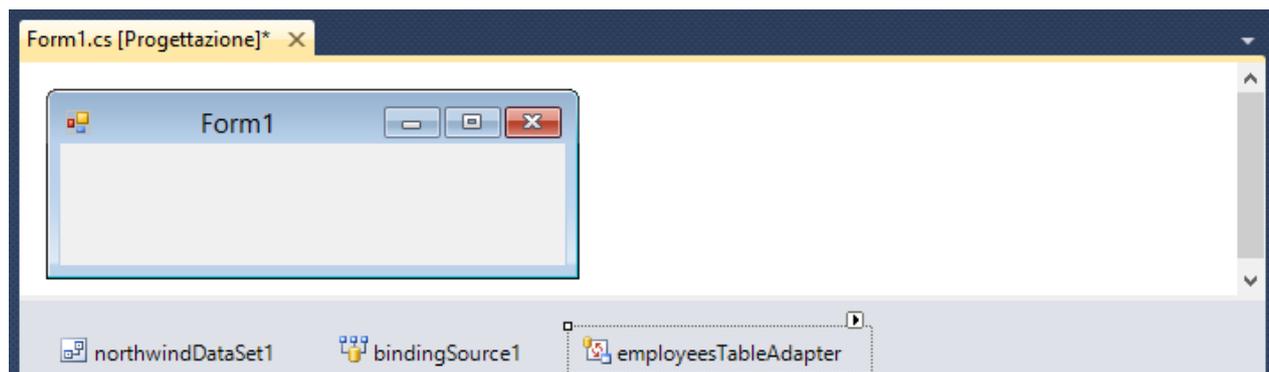


- Come si può osservare il controllo può essere agganciato sia al `NorthwindDataSet` (la classe) sia alla istanza `northwindDataSet1` (la istanza)
- In questa fase scegliamo la seconda opzione ovvero l'istanza

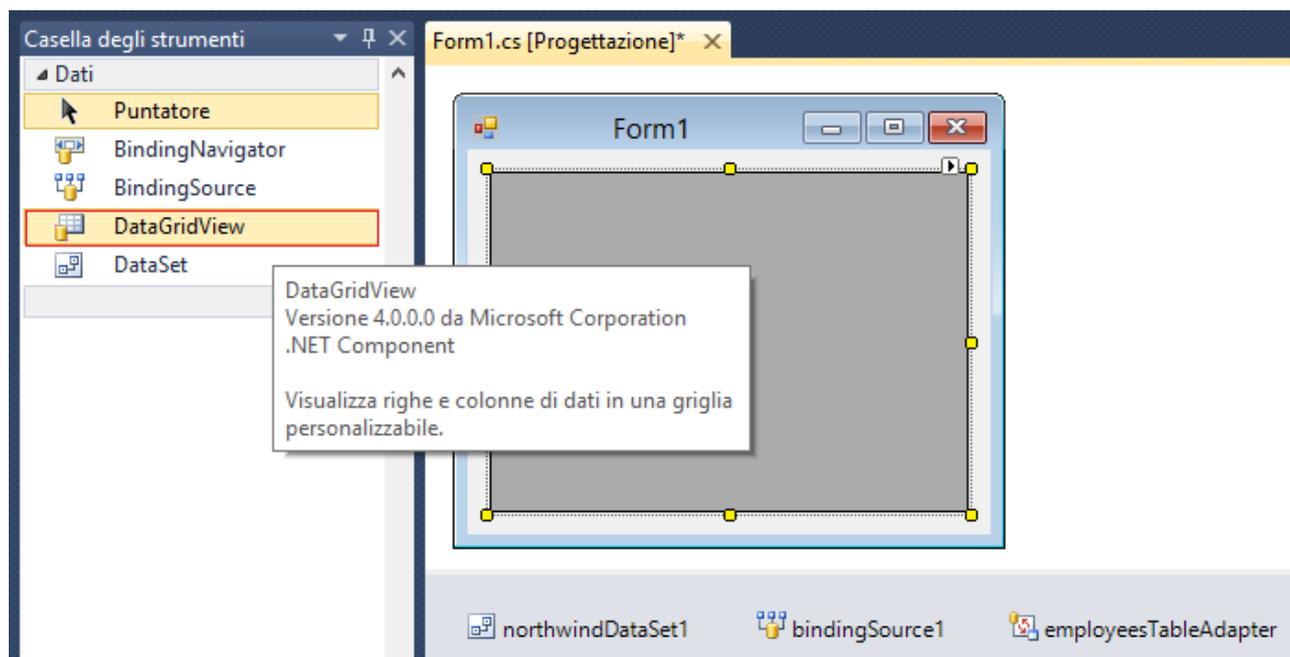


- Adesso si seleziona la proprietà `DataMember` del `bindingSource1` e si apre la casella a discesa come nella figura qui a lato
- La casella mostra tutte le tabelle e tutte le query contenute nel file del database
- Scegliamo la voce `Employees` (Impiegati) per connettere il `bindingSource1` alla tabella degli impiegati
- Il RAD aggiunge un controllo denominato `employeesTableAdapter` ai controlli non visivi proprio accanto ai primi due componenti già aggiunti (vedi figura qui sotto)
- Il `TableAdapter` è un controllo non visuale che permette la connessione tra un `BindingSource` e un controllo visuale





➤ Inserire un componente DataGridView ed adattarlo alla finestra come nella figura:



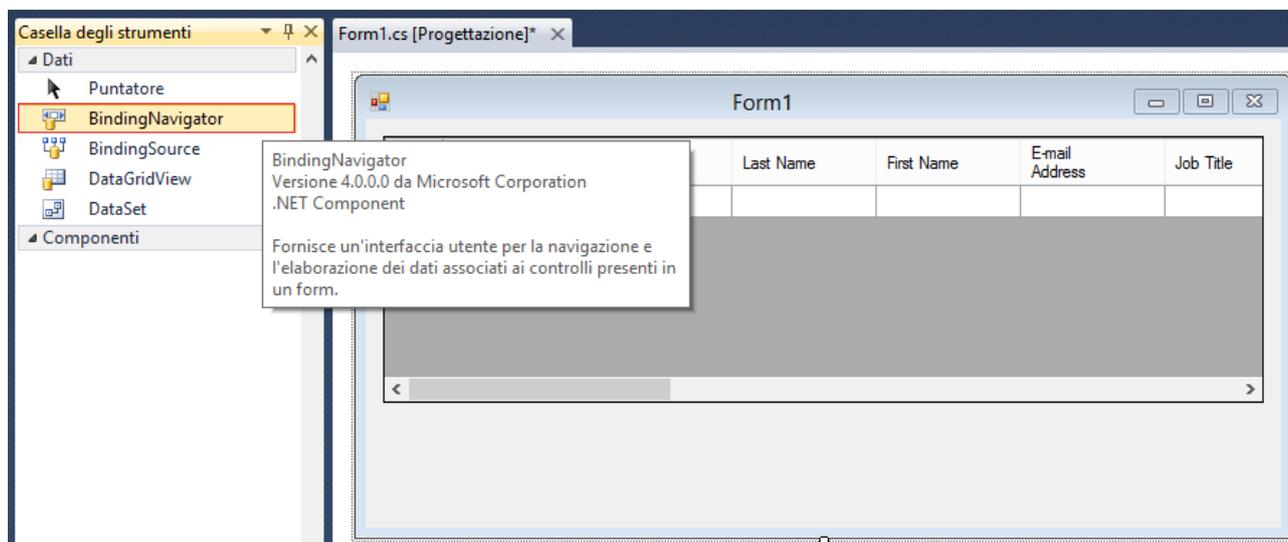
- Individuare la proprietà DataSource del DataGridView1 e aprire la sua casella a discesa
- Selezionare la voce `northwindDataSet1`
- Individuare la proprietà DataMember del DataGridView1 e aprire la sua casella a discesa
- Selezionare la voce `Employees`
- Adattare le dimensioni del Form1 e del DataGridView1
- Eseguire il progetto



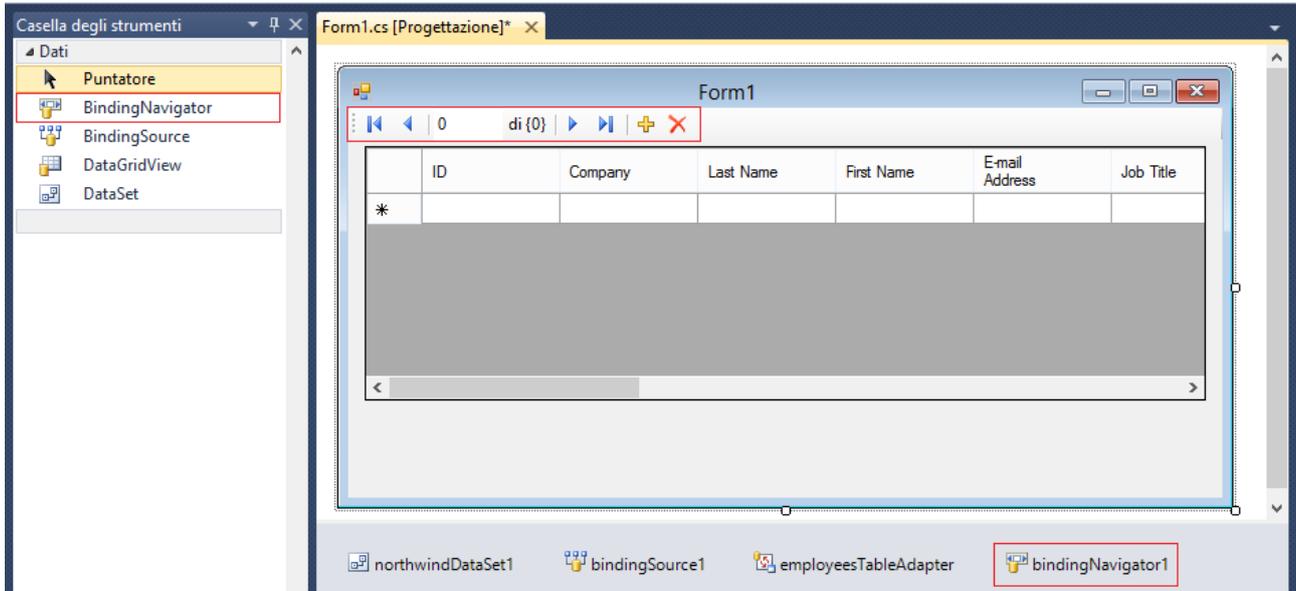
	ID	Company	Last Name	First Name	E-mail Address	Job Title	Business Phone
	1	Northwind Traders	Freehafer	Nancy	nancy@northwin...	Sales Represent...	(123)555-01
	2	Northwind Traders	Cencini	Andrew	andrew@northwi...	Vice President, S...	(123)555-01
	3	Northwind Traders	Kotas	Jan	jan@northwindtra...	Sales Represent...	(123)555-01
▶	4	Northwind Traders	Sergienko	Mariya	mariya@northwin...	Sales Represent...	(123)555-01
	5	Northwind Traders	Thorpe	Steven	steven@northwin...	Sales Manager	(123)555-01
	6	Northwind Traders	Neipper	Michael	michael@northwi...	Sales Represent...	(123)555-01

PROGETTO GUIDATO

- Aprire il precedente progetto con Visual Studio in linguaggio C#
- Individuare la sezione dei controlli intitolata Dati

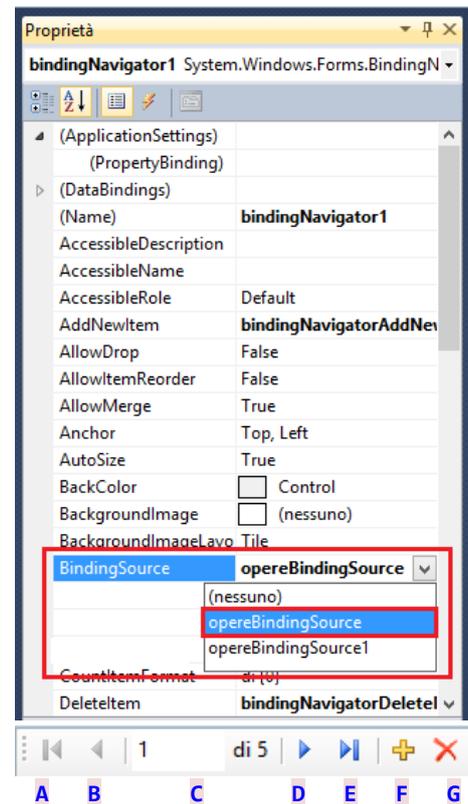
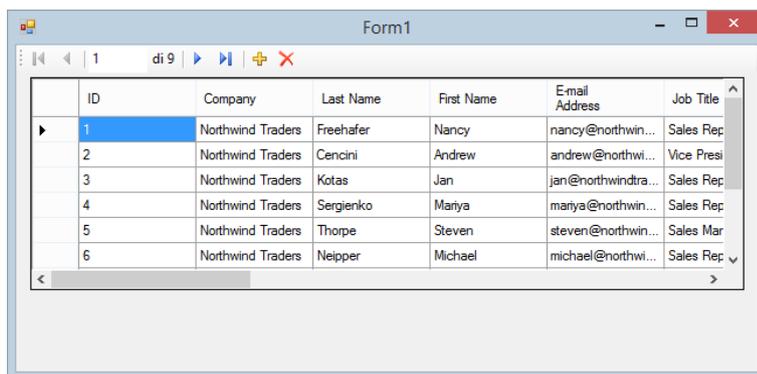


- Selezionare il controllo **BindingNavigator** e inserirlo nel Form1
- La finestra di progettazione dovrebbe apparire come nella seguente figura:



- Nel Form1 è comparsa (in alto) una barra che è composta da diversi pulsanti ed elementi
- Sotto il Form1 è comparso un componente non visuale denominato `bindingNavigator1`
- Selezionare componente `bindingNavigator1` e consultare le sue proprietà

- Individuare tra le sue proprietà quella denominata `BindingSource` e selezionare l'elemento denominato `bindingSource1`
- Questo collegamento vincola il navigatore a legarsi al componente di tipo `BindingSource` che dovrebbe essere lo stesso del `DataGridView1`; in pratica i due controlli (navigatore e griglia) saranno legate allo stessa sorgente di collegamento



- Quando si avvia il progetto nella griglia appariranno i record esistenti nella tabella di database, ma anche il navigatore si predispose mostrando informazioni relative alla tabella
- Provare il progetto



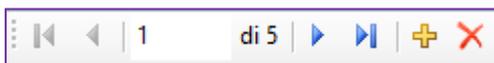
IL NAVIGATORE PER BASI DI DATI



[https://msdn.microsoft.com/it-it/library/8zhc8d2f\(v=vs.110\).aspx](https://msdn.microsoft.com/it-it/library/8zhc8d2f(v=vs.110).aspx)

Il controllo `BindingNavigator` è composto da una barra (ToolStrip) con una serie di elementi (oggetti ToolStripItem) per la maggior parte delle azioni correlate ai dati: aggiunta, eliminazione ed esplorazione. Per impostazione predefinita, il controllo `BindingNavigator` contiene questi pulsanti standard.

Nell'immagine riportata di seguito è illustrato il controllo `BindingNavigator` in un form.



È possibile utilizzare il controllo `BindingNavigator` per fornire agli utenti un metodo standard per la ricerca e la modifica dei dati in un Windows Form. `BindingNavigator` viene spesso utilizzato con il componente `BindingSource` per consentire agli utenti di spostarsi tra i vari record di dati in un form e apportare modifiche ai record.

- La sezione centrale del navigatore indica il numero del record corrente; il numero non coincide con il valore della chiave, ma solo col numero di record ottenuto dal database
- La sezione indica anche quanti record totali vi sono in tabella
- I pulsanti con icone di freccia servono per spostarsi tra i record della tabella
- I pulsanti a sinistra servono rispettivamente per posizionarsi sul record primo o precedente in elenco
- I pulsanti a destra servono rispettivamente per posizionarsi sul record successivo o ultimo in elenco
- Il pulsante con un simbolo + serve invece per inserire un nuovo record nella tabella
- Il pulsante con un simbolo X serve invece per eliminare un record esistente nella tabella



PROCEDURE SUL DATASET

SALVARE LE MODIFICHE DI UN DATASET IN UN DATABASE



<https://msdn.microsoft.com/it-it/library/xzb1zw3x.aspx>

Le modifiche operate sul `DataGridView` non sono automaticamente salvate sul database.

In effetti ogni modifica è effettuata su una memoria locale che, in attesa di conferma di salvataggio sul database, resta congelata senza avere effetti reali.



Dopo aver modificato e convalidato i dati del **dataset** in uso, è quindi necessario inviare i dati aggiornati a un database. Per eseguire questa operazione, occorre invocare il metodo **Update** di un oggetto **TableAdapter**. Il metodo Update dell'adattatore consente di aggiornare una tabella di dati singola e di eseguire il comando corretto (INSERT, UPDATE o DELETE) in base alla proprietà RowState di ciascuna riga di dati della tabella.

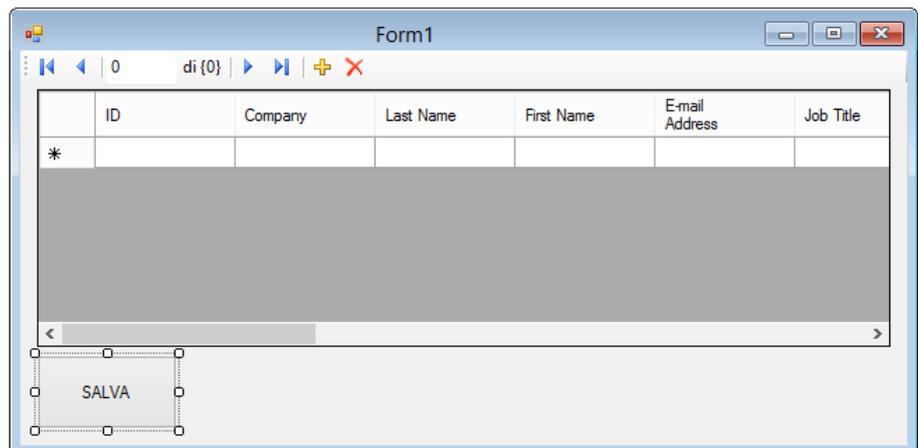
Quando si salvano i dati nelle tabelle correlate, Visual Studio fornisce un nuovo componente **TableAdapterManager** che consente di eseguire i salvataggi nell'ordine corretto in base ai vincoli di chiave esterna definiti nel database.

Per effettuare il salvataggio occorre quindi chiamare il metodo **Update** di un oggetto **TableAdapter** passando il nome della tabella di dati contenente i valori da scrivere nel database. L'uso del metodo è simile al seguente frammento di codice:

```
Try
{
    this.Validate();
    this.miatabellaBindingSource1.EndEdit();
    this.miatabellaTableAdapter.Update(this.miodatabaseDataSet.Opere);
    MessageBox.Show("Modifiche salvate!");
}
catch (System.Exception ex)
{
    MessageBox.Show("Operazione fallita!\n" + ex.GetType());
}
```

PROGETTO GUIDATO (SALVATAGGIO MODIFICHE)

- Aprire il precedente progetto con Visual Studio in linguaggio C#
- Aggiungere un pulsante al Form1 come nella figura a lato
- Modificare il suo testo in SALVA
- Associare al pulsante il codice seguente:



ID	Company	Last Name	First Name	E-mail Address	Job Title
*					

SALVA



Visual C#

```
private void button1_Click(object sender, EventArgs e)
{
    try
    {
        this.Validate();
        this.bindingSource1.EndEdit();

        this.employeesTableAdapter.Update(this.northwindDataSet1.Employees);
        MessageBox.Show("Modifiche salvate!");
    }
    catch (System.Exception ex)
    {
        MessageBox.Show("Operazione fallita!\n" + ex.GetType());
    }
}
```

- Eseguire il progetto e modificare un qualsiasi record esistente nel DataGridView (vedi illustrazione qui sotto)
- Ripetere anche per altri record e infine confermare la modifica col pulsante Salva

	EmployeeID	LastName	FirstName	Title	TitleOfCourtesy	BirthDate
	1	Carta	Bianca	Sales Represent...	Ms.	08/12/19
	2	Carta	Nerina	Vice President, S...	Dr.	19/02/19
▶	3	Piano	Guido	Sales Represent...	Ms.	30/08/19
	4	Piano	Remo	Sales Represent...	Mrs.	19/09/19
	5	Corrente	Daria	Sales Manager	Mr.	04/03/19
	6	Piu	Mino	Sales Represent...	Mr.	02/07/19
	7	Vento	Clemente	Sales Represent	Mr	29/05/19

- Chiudere l'esecuzione dell'applicazione
- Se si riavvia l'applicazione i dati dovrebbero essere cambiati

PROGETTO GUIDATO (RINFRESCARE I DATI)

- Aprire il precedente progetto con Visual Studio in linguaggio C#



- Aggiungere un altro pulsante al Form1 come nella figura a lato
- Modificare il suo testo in AGGIORNA
- Associare al pulsante il codice seguente:

	EmployeeID	LastName	FirstName	Title	TitleOfCourtesy	BirthDate ^
	1	Carta	Bianca	Rappresentante ...	Ms.	08/12/19
▶	2	Carta	Nerina	Vice Presidente	Dr.	19/02/19
	3	Piano	Guido	Rappresentante ...	Mr.	30/08/19
	4	Piano	Remo	Rappresentante ...	Dr.	19/09/19
	5	Corente	Daria	Manager Vendite	Ms.	04/03/19
	6	Piu	Mino	Direttore Vendite	Mr.	02/07/19
	7	Vento	Clemente	Mananer Vendite	Mr.	29/05/19

Visual C#

```
private void button2_Click(object sender, EventArgs e)
{
    try
    {
        this.employeesTableAdapter.Fill(this.northwindDataSet1.Employees);
        MessageBox.Show("Dati aggiornati!");
    }
    catch (System.Exception ex)
    {
        MessageBox.Show("Operazione fallita!\n" + ex.GetType());
    }
}
```

- Eseguire il progetto e inserire un nuovo record alla fine del DataGridView e poi confermarlo col pulsante Salva
- Poi premere il pulsante AGGIORNA per verificare l'aggiornamento dati
- Chiudere l'esecuzione dell'applicazione e poi riavviarla nuovamente per vedere gli effetti

Procedura: inserire nuovi record in un database



<https://msdn.microsoft.com/it-it/library/ms233812.aspx>

Per inserire nuovi record in un database, è possibile utilizzare il metodo `TableAdapter.Update` oppure uno dei metodi `DBDirect` del `TableAdapter` (in modo specifico il metodo `TableAdapter.Insert`).

Se nell'applicazione non sono utilizzati `TableAdapter`, sarà possibile utilizzare oggetti `Command` per interagire e inserire nuovi record nel database in uso (ad esempio, `SqlCommand`) che vedremo in seguito.

Utilizzare il metodo `TableAdapter.Insert` quando nell'applicazione vengono utilizzati gli oggetti per la memorizzazione dei dati oppure se si desidera un maggiore controllo sulla creazione di nuovi record nel database.

Per inserire nuovi record in un database utilizzando il metodo `TableAdapter.Insert` si deve:

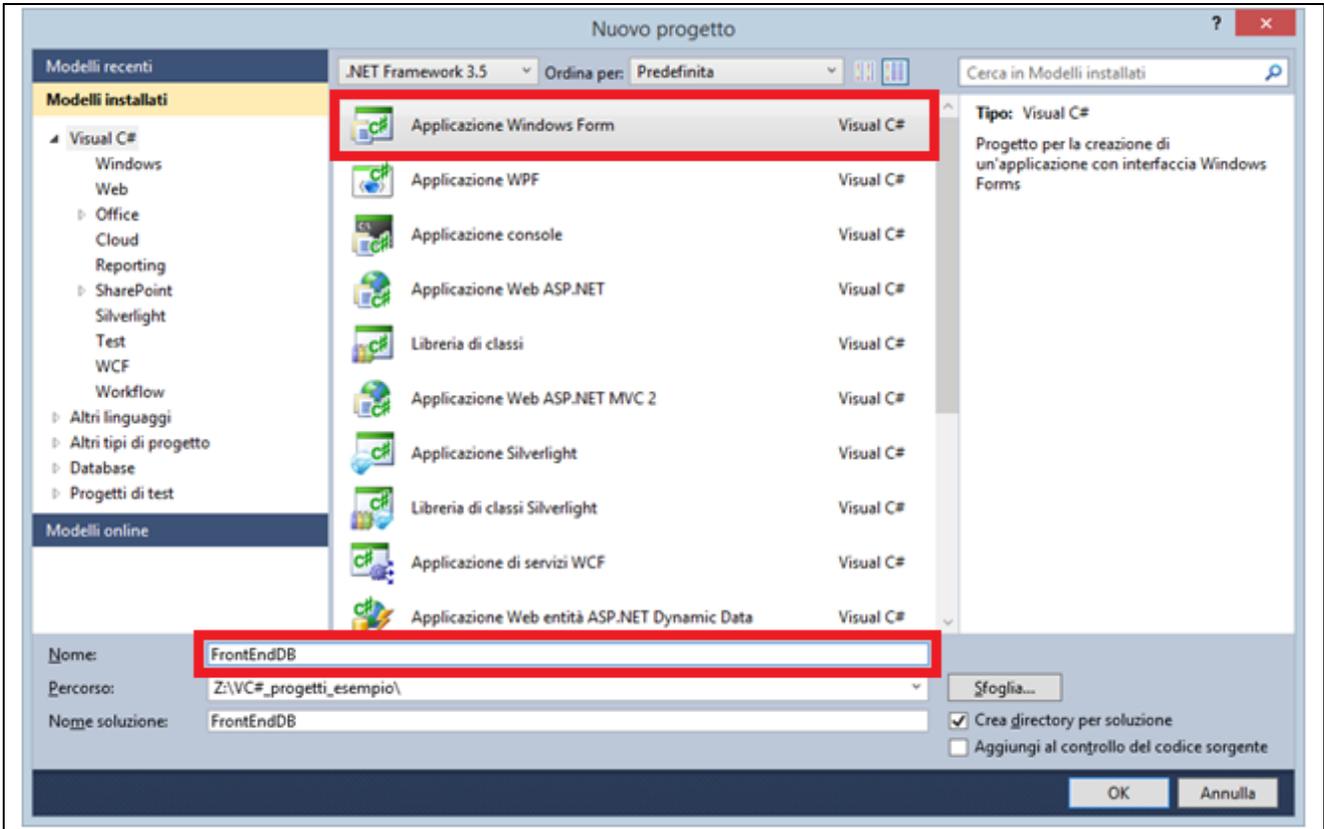
- 1) dichiarare una variabile del tipo `TableAdapter` opportuno (es. `NorthwindDataSetTableAdapters`) e istanziarlo con una `new`.



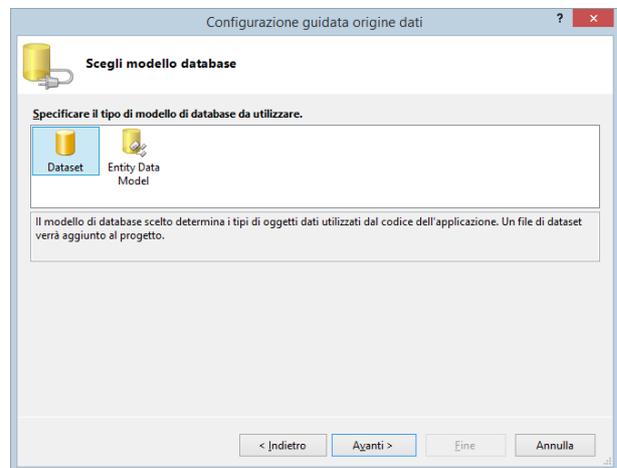
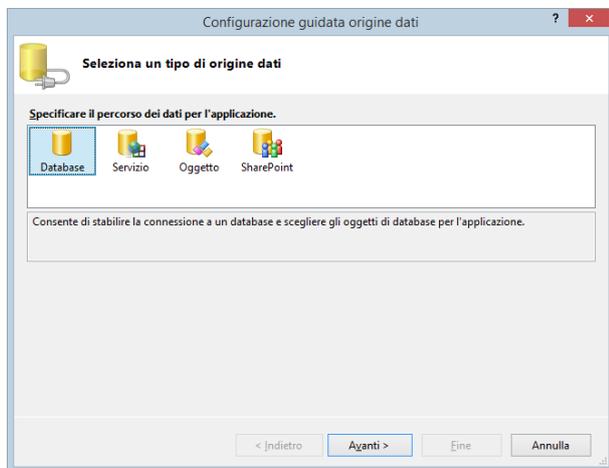
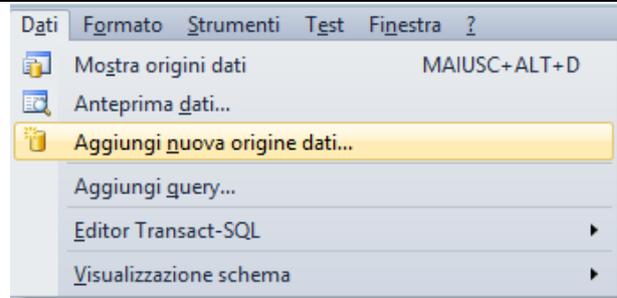
2) poi eseguire un metodo Insert specificando gli argomenti richiesti dalla tabella.

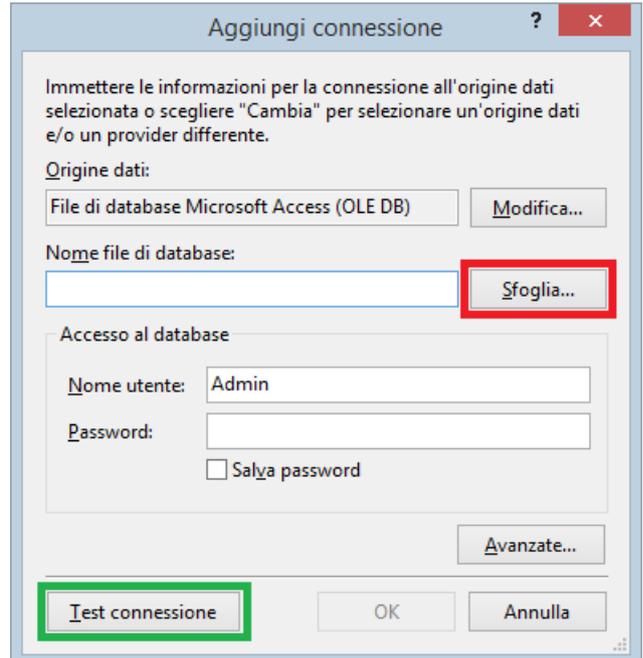
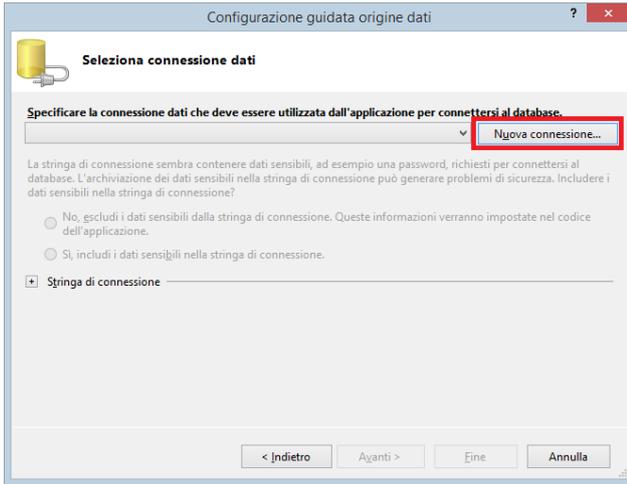
PROGETTO GUIDATO (OLTRE IL DATAGRIDVIEW)

Aprire un nuovo progetto con Visual Studio in linguaggio C#

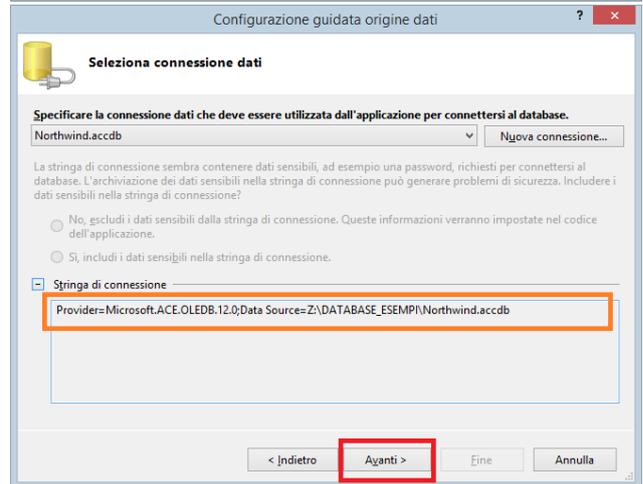
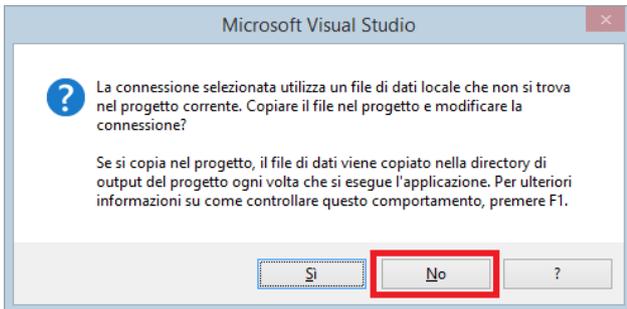


- Selezionare il menu Dati e la voce Aggiungi nuova origine dati ...
- Selezionare Database come origine dati
- Selezionare Dataset come modello
- Selezionare Nuova connessione
- Selezionare Sfogliare e scegliere il DB

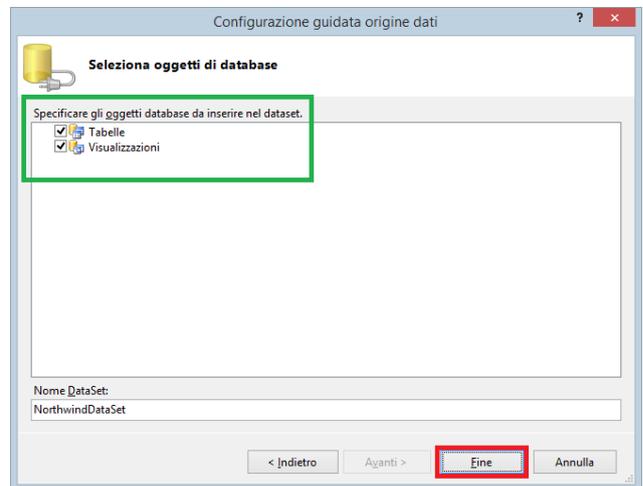
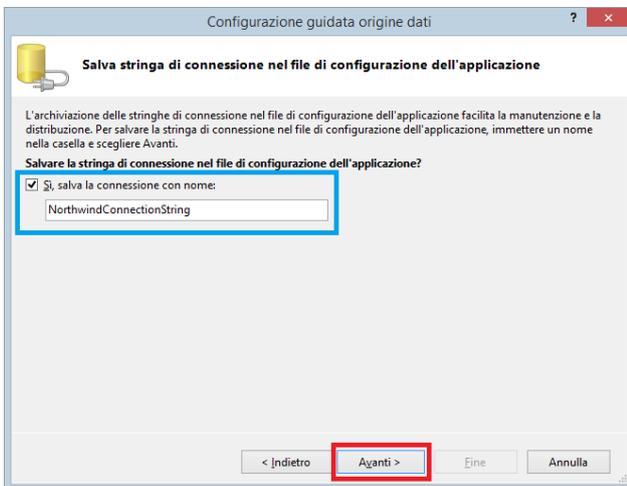




- Individuare Northwind e confermare con OK (eventualmente fare il test)
- Individuare Northwind e confermare con OK (eventualmente fare il test)
- Controllare la stringa di connessione e proseguire con Avanti
- Per poter lavorare sui dati scegliere NO nella proposta di fare una copia dati

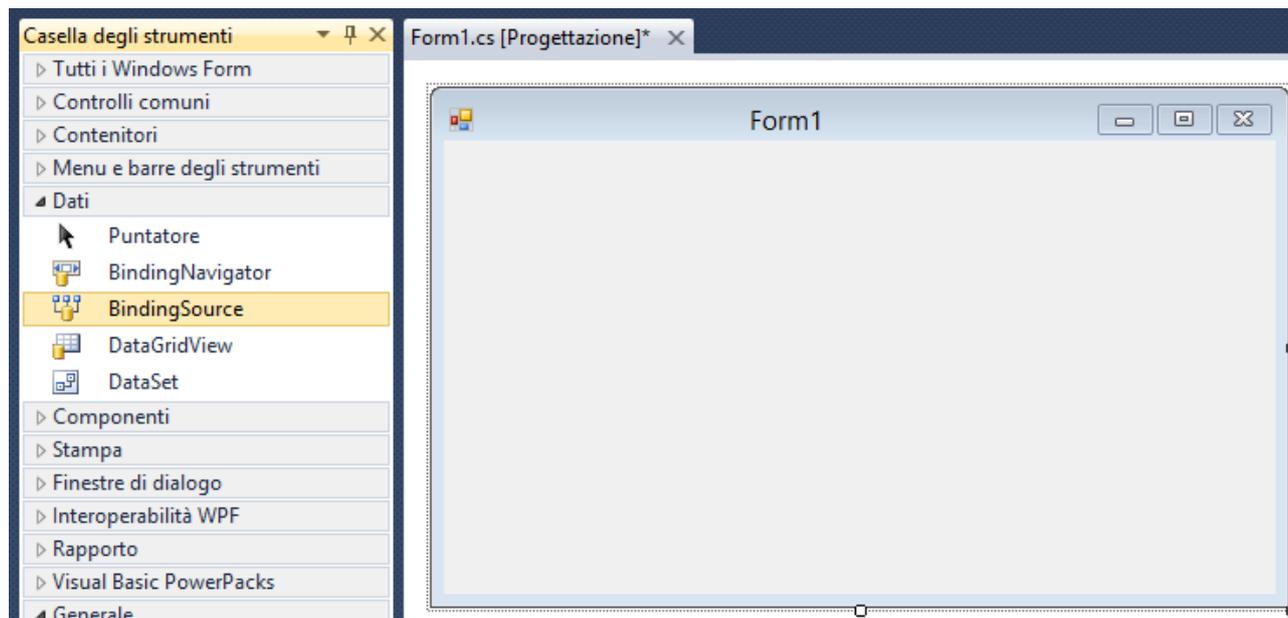


- Controllare la stringa di connessione e proseguire con Avanti
- Confermare il salvataggio con nome della connessione e proseguire con Avanti
- Selezionare tutto per importare quanto più possibile dal database e concludere con Fine





- Dalla casella degli strumenti (vedi figura illustrata di seguito) inserisci un componente BindingSource nel progetto



Visual C#

```
private void button2_Click(object sender, EventArgs e)
{
    try
    {
        this.employeesTableAdapter.Fill(this.northwindDataSet1.Employees);
        MessageBox.Show("Dati aggiornati!");
    }
    catch (System.Exception ex)
    {
        MessageBox.Show("Operazione fallita!\n" + ex.GetType());
    }
}
```

- Eseguire il progetto e inserire un nuovo record alla fine del DataGridView e poi confermarlo col pulsante Salva
- Poi premere il pulsante AGGIORNA per verificare l'aggiornamento dati
- Chiudere l'esecuzione dell'applicazione e poi riavviarla nuovamente per vedere gli effetti

PROGETTO GUIDATO (INSERIRE NUOVI RECORD)

- Aprire il precedente progetto con Visual Studio in linguaggio C#



- Aggiungere un terzo pulsante al Form1 come nella figura a lato
- Modificare il suo testo in INSERISCI
- Se vuoi puoi settare delle immagini
- Associare al pulsante il codice seguente:

EmployeeID	LastName	FirstName	Title	TitleOfCo	BirthDate	HireDate
1	CARTA	Bianca	Rappresentante Vendite	Ms.	08/12/1948	01/05/1995
2	CARTA	Nerina	Vice Presidente	Ms.	19/02/1952	14/08/1995
3	PIANO	Guido	Rappresentante Vendite	Dr.	30/08/1963	01/04/1995
4	PIANO	Remo	Rappresentante Vendite	Dr.	19/09/1937	03/05/1995
5	CORRENTE	Daria	Manager Vendite	Mrs.	04/03/1955	17/10/1995
6	PIU	Mino	Rappresentante Vendite	Mr.	02/07/1963	17/10/1995
7	VFENTO	Clemente	Rappresentante Vendite	Mr.	29/05/1960	02/01/1995

Visual C#

```
private void button3_Click(object sender, EventArgs e)
{
    NorthwindDataSetTableAdapters.EmployeesTableAdapter poldo;
    poldo = new NorthwindDataSetTableAdapters.EmployeesTableAdapter();
    string cognome = "PARTO";
    string nomebattesimo = "Domenica";
    string mansione = "Capo Vendite";
    string titolo = "Mrs.";
    System.DateTime nascita = System.DateTime.Parse("28/02/1964");
    System.DateTime assunzione = System.DateTime.Parse("01/04/1984");
    string indirizzo = "Via Con Vento";
    string citta = "Sassari";
    string regione = "Sardegna";
    string paese = "Italy";
    string cap = "07100";
    string telefono = "+39079123456789";
    string est = "5467";
    try
    {
        poldo.Insert(cognome, nomebattesimo, mansione, titolo
            , nascita, assunzione
            , indirizzo, citta, regione
            , cap, paese, telefono
            , est
            , null, null, null); //fine inserimento
        this.employeesTableAdapter.Fill(this.northwindDataSet1.Employees);
    }
    catch (System.Exception ex)
    {
        MessageBox.Show("Operazione fallita!\n" + ex.GetBaseException());
    }
}
```



- Eseguire il progetto e inserire un nuovo record col pulsante Inserisci
- Chiudere l'esecuzione dell'applicazione e poi riavviarla nuovamente per vedere gli effetti

COMMENTO

Come si deduce dal codice dell'esempio l'istruzione `poldo.Insert(...)`

serve per inserire un record nella tabella legata alla variabile `poldo`.

Ovviamente è possibile preparare i campi del record basandosi su textBox o altri controlli.



ESERCIZI

ESERCIZI SU INTERFACCE ELEMENTARI

Esercizio 1) NORTHWIND ORDERS

BASE DI DATI

- Ripetere i progetti guidati Fase 1, 2 e 3 ma legando il DataGridView alla tabella Orders.

Esercizio 2) NORTHWIND PRODUCTS

BASE DI DATI

- Ripetere i progetti guidati Fase 1, 2 e 3 ma legando il DataGridView alla tabella Products.

Esercizio 3) NORTHWIND SUPPLIERS

BASE DI DATI

- Ripetere i progetti guidati Fase 1, 2 e 3 ma legando il DataGridView alla tabella Suppliers.

Esercizio 4) NORTHWIND ORDER DETAILS

BASE DI DATI

- Ripetere i progetti guidati Fase 1, 2 e 3 ma legando il DataGridView alla tabella [Order Details].

Esercizio 5) DATABASE BIBLIOTECA

BASE DI DATI

- Creare una nuova base di dati Biblioteca con tre tabelle Iscritti, Testi e Prestiti.
- Poi creare una interfaccia in Visual Studio che mostri in tre diverse DataGridView rispettivamente ciascuna delle tre tabelle



SOMMARIO

BASI DI DATI	2
DATABASE E ACCESS.....	2
Concetto di base di dati	2
Database Access	2
Tabelle in Access	2
DATABASE DI RIFERIMENTO.....	3
Un esempio di base di dati	3
UNA PRIMA INTERFACCIA	4
 Progetto Guidato	4
ANALISI DEI COMPONENTI.....	10
Componenti e database	10
DataSet.....	10
BindingSource.....	11
TableAdapter	11
Schema di collegamento	11
 Progetto Guidato (fase 1)	12
Cosa è successo?	15
 Progetto Guidato (fase 2)	16
Cosa è successo?	16
 Progetto Guidato (fase 3)	16
 Progetto Guidato	19
Il Navigatore per Basi di Dati.....	21
PROCEDURE SUL DATASET	21
salvare le modifiche di un dataset in un database	21
 Progetto Guidato (Salvataggio modifiche).....	22
 Progetto Guidato (Rinfrescare i dati).....	23
 Progetto Guidato (oltre il datagridview).....	25
 Progetto Guidato (Inserire nuovi record)	27
ESERCIZI SU INTERFACCE ELEMENTARI	30
Esercizio 1) NorthWind Orders	30
Esercizio 2) NorthWind Products	30
Esercizio 3) NorthWind Suppliers.....	30
Esercizio 4) NorthWind Order Details	30
Esercizio 5) Database Biblioteca	30