

# Il blog di Emanuele Mattei in ambito Database

Programmazione database, ado.net e database mobile

## Come visualizzare ed inserire le immagini in un database Microsoft Access tramite ADO.NET

Riporto l'articolo scritto per il sito [iprogrammatori http://www.iprogrammatori.it/articoli/programmazione/art\\_adonet-inserire-e-salvare-immagine-in-un\\_131.aspx](http://www.iprogrammatori.it/articoli/programmazione/art_adonet-inserire-e-salvare-immagine-in-un_131.aspx) riguardante la possibilità di inserire e visualizzare le immagini in un database Microsoft Access.

In questo articolo, vedremo come salvare delle immagini (formato jpeg e gif) in un campo di una tabella di un database Microsoft Access. Il campo è di tipo oggetto ole. Con qualche modifica al codice, si può utilizzare anche per un database sql server con campo di tipo blob. A fine articolo trovate il sorgente di esempio utilizzato in questo articolo, per il linguaggio Visual Basic .NET e C#

Ora vediamo, come inserire e visualizzare le immagini in un database, salvando lo stream dati dell'immagine in questo campo.

### Creazione del database.

Si crea un nuovo database Microsoft Access 2003, con una tabella, una colonna di tipo contatore e l'altra di tipo oggetto ole, salvate la tabella ed il database.

### Creazione del progetto.

Si crea un nuovo progetto di tipo windows application, nella finestra principale, inseriamo tre pulsanti, una casella di testo, una label un controllo picturebox, due radiobutton ed un controllo groupbox. Il tutto come mostrato in figura 1.



Figura 1

I pulsanti, avranno il compito, di selezionare l'immagine, di salvare l'immagine nel database e di visualizzarla.

### Stesura del codice.

Ora passiamo al codice. Utilizziamo lo spazio dei nomi, oledb per la gestione dei dati per il database

Microsoft Access e system.io per la gestione dei file.

Di seguito si riporta tale dichiarazioni.

### VB.Net

```
Imports System.Data.OleDb
```

```
Imports System.IO
```

### C#

```
using System.IO;
```

```
using System.Data.OleDb;
```

Il pulsante per la ricerca delle immagini, chiamato btnImmagine, avrà il compito di instanziare un oggetto di tipo openFileDialog, per la ricerca delle immagini, ed impostare la proprietà text della casella di testo (txtimmagine) con il percorso e nome del file immagine.

Di seguito si riporta il frammento di codice di tali operazioni.

### VB.Net

```
Private Sub btnImmagine_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnImmagine.Click
```

```
Dim dlgImmagine As New OpenFileDialog()
```

```
dlgImmagine.Title = "Seleziona un immagine da importare"
```

```
dlgImmagine.Multiselect = False
```

```
dlgImmagine.Filter = "Formato JPEG(*.jpg)|*.jpg|Formato GIF (*.gif)|*.gif"
```

```
dlgImmagine.FilterIndex = 2
```

```
dlgImmagine.FileName = ""
```

```
dlgImmagine.ShowDialog()
```

```
txtimmagine.Text = dlgImmagine.FileName
```

```
End Sub
```

### C#

```
private void btnImmagine_Click(object sender, EventArgs e)
```

```
{
```

```
OpenFileDialog dlgImmagine = new OpenFileDialog();
```

```
dlgImmagine.Title = "Seleziona un immagine da importare";
```

```
dlgImmagine.Multiselect = false;dlgImmagine.Filter = "Formato JPEG(*.jpg)|*.jpg|Formato GIF (*.gif)|*.gif";
```

```
dlgImmagine.FilterIndex = 2;
```

```
dlgImmagine.FileName = "";
```

```
dlgImmagine.ShowDialog();
```

```
txtimmagine.Text = dlgImmagine.FileName;
```

```
}

```

Il pulsante per il salvataggio dell'immagine (chiamato BtnSalva) avrà il compito di salvare in formato stream dati l'immagine nel database, si utilizzerà la classe filestream, per la gestione del file, mentre la classe dataset, oledbadapter oledbcommandbuilder per la gestione dei dati, in particolare per il salvataggio dei dati. La variabile di tipo array byte, conterrà le informazioni dell'immagine selezionata.

Di seguito si riporta il codice delle suddette dichiarazioni.

### VB.Net

```
Private Sub btnSalva_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSalva.Click

```

```
If txtimmagine.Text Is String.Empty Then

```

```
Exit Sub

```

```
End If

```

```
'Creo lo stream dati byte, della immagine

```

```
Dim fs As FileStream = New FileStream(txtimmagine.Text, FileMode.OpenOrCreate, FileAccess.Read)

```

```
Dim ImgData(fs.Length) As Byte

```

```
fs.Read(ImgData, 0, System.Convert.ToInt32(fs.Length))

```

```
fs.Close()

```

```
'Mi connetto al db

```

```
Dim sql As String = "SELECT * FROM dati"

```

```
Dim condati As OleDbConnection = New OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; " & "Data Source=C:\dbimmagine.mdb")

```

```
Try

```

```
If condati.State <> ConnectionState.Open Then

```

```
condati.Open()

```

```
End If

```

```
'oggetti per il salvataggio dei dati

```

```
Dim adapterDati As OleDbDataAdapter = New OleDbDataAdapter(sql, condati)

```

```
Dim cmdBuilder As OleDbCommandBuilder = New OleDbCommandBuilder(adapterDati) Dim dtsDati As New DataSet()

```

```
adapterDati.MissingSchemaAction = MissingSchemaAction.AddWithKey

```

```
'Carico il dataset

```

```
adapterDati.Fill(dtsDati, "Dati")

```

```
'Creo un datarow che mi rappresenta la nuova riga

```

```
Dim dtrRiga As DataRow = dtsDati.Tables("Dati").NewRow()
dtrRiga("immagine") = ImgData
' aggiungo la riga al dataset

dtsDati.Tables("Dati").Rows.Add(dtrRiga)
'Salvo i dati modificati
adapterDati.Update(dtsDati, "Dati")
MessageBox.Show("Immagine salvata")
Catch ex As Exception
MessageBox.Show("Errore: " & ex.Message)
Finally
'Chiudo la connessione
If Not condati Is Nothing Then
If condati.State = ConnectionState.Open Then
condati.Close()
End If
condati.Dispose()
End If
End Try
End Sub

C#
private void btnSalva_Click(object sender, EventArgs e)
{
if (txtimmagine.Text == string.Empty)
{
return;
}
//Creo lo stream dati byte, della immagine

FileStream fs = new FileStream(txtimmagine.Text, FileMode.OpenOrCreate, FileAccess.Read);
Byte[] ImgData = new Byte[fs.Length];
fs.Read(ImgData, 0, System.Convert.ToInt32(fs.Length));
fs.Close();

//Mi connetto al db
```

```
string sql = "SELECT * FROM dati";

OleDbConnection condati = new OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; " + "Data
Source=C:\\dbimmagine.mdb");

try
{
if (condati.State != ConnectionState.Open)
{
condati.Open();
}
// oggetti per il salvataggio dei dati

OleDbDataAdapter adapterDati = new OleDbDataAdapter(sql, condati);
OleDbCommandBuilder cmdBuilder = new OleDbCommandBuilder(adapterDati);
DataSet dtsDati = new DataSet();
adapterDati.MissingSchemaAction = MissingSchemaAction.AddWithKey;
//Carico il dataset

adapterDati.Fill(dtsDati, "Dati");
//Creo un datarow che mi rappresenta la nuova riga
DataRow dtrRiga = dtsDati.Tables["Dati"].NewRow();
dtrRiga["immagine"] = ImgData;
// aggiungo la riga al dataset
dtsDati.Tables["Dati"].Rows.Add(dtrRiga);
//Salvo i dati modificati

adapterDati.Update(dtsDati, "Dati");MessageBox.Show("Immagine salvata");
}
catch (Exception ex)
{
MessageBox.Show("Errore: " + ex.Message);
}
finally
{
//Chiudo la connessione
```

```

if (condati == null)
{
if (condati.State == ConnectionState.Open)
{
condati.Close();
}
condati.Dispose();
}

}
}

```

Il pulsante per la visualizzazione dell'immagine (denominato btnVisualizza) avrà il compito di visualizzare l'immagine in un controllo picturebox, se viene scelta la radiobutton con il valore "si", ossia per il salvataggio fisico dell'immagine, verrà creato un file immagine sul computer, altrimenti, se l'opzione selezionata ha il valore no, non verrà creato nessun file immagine.

Utilizzeremo le classi filestream e binarywriter per la generazione del file, mentre le classi oledbcommand, oledbdatareader per la lettura dei dati.

Di seguito si riporta il codice per la visualizzazione dell'immagine.

#### **VB.Net**

```

Private Sub btnVisualizza_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles btnVisualizza.Click

```

```

Dim sql As String = "SELECT * FROM dati"

```

```

Dim conDati As OleDbConnection = New OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; " &
"Data Source=C:\dbimmagine.mdb")

```

```

Try

```

```

If conDati.State <> ConnectionState.Open Then

```

```

conDati.Open()

```

```

End If

```

```

Dim cmd As OleDbCommand = New OleDbCommand(sql, conDati)

```

```

Dim fs As FileStream

```

```

Dim binaryScrivi As BinaryWriter

```

```

Dim Dimensionebuffer As Integer = 300000

```

```

Dim outbyte(300000 - 1) As Byte

```

```

Dim reader As OleDbDataReader = cmd.ExecuteReader(CommandBehavior.SequentialAccess)

```

```

'leggo il primo record

```

```
reader.Read()
```

```
'senza salvataggio della immagine
```

```
If rdbSi.Checked = False Then
```

```
Dim arrayStreamDati() As Byte = reader.Item("immagine")
```

```
If arrayStreamDati.Length > 0 Then
```

```
'Carico in uno memoryStream l'array stream dati
```

```
Dim Stream As MemoryStream = New MemoryStream(arrayStreamDati, True)
```

```
Stream.Write(arrayStreamDati, 0, arrayStreamDati.Length)
```

```
'Creo un oggetto bitmap contenente l'immagine
```

```
Dim immagine As Bitmap = New Bitmap(Stream)
```

```
'carico nella picture box
```

```
PictureBox1.Image = immagine
```

```
PictureBox1.SizeMode = PictureBoxSizeMode.StretchImage
```

```
'Chiudo lo streamadati
```

```
Stream.Close()
```

```
End If
```

```
Else
```

```
'Salvo l'immagine sul computer
```

```
fs = New FileStream("C:\pippo.bmp", FileMode.OpenOrCreate, FileAccess.Write)
```

```
binaryScrivi = New BinaryWriter(fs)
```

```
reader.GetBytes(1, 0, outbyte, 0, Dimensionebuffer)
```

```
binaryScrivi.Write(outbyte)
```

```
binaryScrivi.Flush()
```

```
'Chiudo gli oggetti
```

```
binaryScrivi.Close()
```

```
fs.Close()
```

```
reader.Close()
```

```
'Visualizzo l'immagine
```

```
Dim immagine As Image = Nothing
```

```
immagine = Image.FromFile("C:\pippo.bmp")
```

```
PictureBox1.Image = immagine
```

```
PictureBox1.SizeMode = PictureBoxSizeMode.StretchImage
```

```
End If
```

```
Catch ex As Exception
```

```
MessageBox.Show("Errore: " & ex.Message)
```

```
Finally
```

```
'chiusura connessione
```

```
If Not conDati Is Nothing Then
```

```
If conDati.State = ConnectionState.Open Then
```

```
conDati.Close()
```

```
End If
```

```
conDati.Dispose()
```

```
End If
```

```
End Try
```

```
End Sub
```

```
C#
```

```
private void btnVisualizza_Click(object sender, EventArgs e)
```

```
{
```

```
string sql = "SELECT * FROM dati";
```

```
OleDbConnection conDati = new OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; " + "Data Source=C:\\dbimmagine.mdb");
```

```
try
```

```
{
```

```
if (conDati.State != ConnectionState.Open)
```

```
{
```

```
conDati.Open();
```

```
}
```

```
OleDbCommand cmd = new OleDbCommand(sql, conDati);
```

```
FileStream fs;
```



```
BinaryWriter binaryScrivi;  
  
Int32 Dimensionebuffer = 300000;  
  
Byte[] outbyte = new Byte[300000 - 1];  
  
OleDbDataReader reader = cmd.ExecuteReader(CommandBehavior.SequentialAccess);  
  
//leggo il primo record  
reader.Read();  
  
//senza salvataggio della immagine  
if (rdbSi.Checked == false)  
{  
Byte[] arrayStreamDati = (Byte[])reader["immagine"]; // Item["immagine"];  
if (arrayStreamDati.Length > 0)  
{  
//Carico in uno memoryStream l'array stream dati  
MemoryStream Stream = new MemoryStream(arrayStreamDati, true);  
Stream.Write(arrayStreamDati, 0, arrayStreamDati.Length);  
  
//Creo un oggetto bitmap contenente l'immagine  
Bitmap immagine = new Bitmap(Stream);  
  
//carico nella picture box  
PictureBox1.Image = immagine;  
PictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;  
  
//Chiudo lo streamadati  
Stream.Close();  
}  
}  
else  
{  
//Salvo l'immagine sul computer  
fs = new FileStream("C:\\\pippo.bmp", FileMode.OpenOrCreate, FileAccess.Write);  
binaryScrivi = new BinaryWriter(fs);  
reader.GetBytes(1, 0, outbyte, 0, Dimensionebuffer);  
binaryScrivi.Write(outbyte);  
binaryScrivi.Flush();  
  
//Chiudo gli oggetti  
binaryScrivi.Close();  
fs.Close();  
}
```

```
reader.Close();  
  
//Visualizzo l'immagine  
  
Image immagine = null;immagine = Image.FromFile("C:\\pippo.bmp");  
PictureBox1.Image = immagine;  
PictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;  
}  
}  
catch (Exception ex)  
{  
    MessageBox.Show("Errore: " + ex.Message);  
}  
finally  
{  
    //chiusura connessione  
    if (conDati != null)  
    {  
        if (conDati.State == ConnectionState.Open)  
        {  
            conDati.Close();  
        }  
        conDati.Dispose();  
    }  
}  
}
```

### Conclusioni:

In questo articolo, abbiamo visto come tramite le classi del namespace system.io e system.oledb, possiamo salvare un immagine in un database Microsoft Access e visualizzarle in un controllo picturebox, sia con e senza creazione del file immagine. Una soluzione, per non dipendere da file locali.

Tramite la parola download si può scaricare il file di esempio utilizzato nell'articolo.

[Download.](#)

## Comments

No Comments

(C) 2007 User Group Italiano di SQL Server