

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Foros](#) | [Tutoriales](#) | [Servicios Gratuitos](#) | [Contacte](#)

Tutorial desarrollado por: Isaac Gutierrez Gómez.

Isaac está actualmente trabajando como responsable de arquitectura de una importante empresa perteneciente a una gran entidad financiera. Ha liderado varios proyectos de aplicaciones para Internet, así como proyectos de telecomunicaciones y de IT para distintas industrias.

Contacta en: isaac@adictosaltrabajo.com

Descargar este documento en formato PDF [ado.pdf](#)

DB Ghost for SQL Server

Create the perfect db release every time - build, compare & synchronize

Sql Server Classes

Need job skills for SQL Server? Try our free videos before you buy!

Universal Data Provider

ADO.NET provider to access MS SQL, MS Access, Oracle, MySQL, DB2,

SQLXML Editor - XMLSpy

Edit XML Schemas for SQL Server .2K Build .NET Apps, D/L a Free Trial

Anuncios Goooooogole

Anunciarse en este sitio

TUTORIAL DE ADO.NET

INTRODUCCIÓN

ADO.NET es una evolución del modelo de datos ADO, de hecho utiliza algunos objetos ADO como Connection y Command, pero también agrega algunos objetos nuevos como DataSet, DataReader y DataAdapter.

Una característica importante de ADO.NET es el objeto DataSet que funciona de forma independiente. Está basado en XML y es un objeto que está desconectado y no tiene conciencia de cual es el origen y el destino de los datos. Es el objeto DataAdapter el que se conecta a la base de datos para llenar el objeto DataSet. Por lo tanto el objeto DataAdapter es la interfaz de conexión con los orígenes y destino de los datos, esto hace que ADO.NET sea mucho más consistente y que tenga una mayor eficacia.

Para trabajar con conexiones de bases de datos podemos escoger entre un proveedor de datos de SQL Server (System.Data.SqlClient) y un proveedor de datos OLE DB (System.Data.OleDb).

Existen cinco objetos básicos en ADO.NET:

- Connection: Conectar con la base de datos y administrar transacciones.
- Command: Para ejecutar comandos SQL.
- DataReader: Sirve para leer una secuencia de registros de datos sólo hacia delante.
- DataSet: Es similar al Recordset de ADO, pero más eficaz ya que representa a una memoria caché de datos, con estructuras análogas a las de una base de datos y siempre está desconectado.
- DataAdapter: Sirve para insertar datos en un objeto DataSet y reconciliar datos de la base de datos desde un objeto DataSet.

Creo mejor para seguir con el tutorial hacer algún ejemplo y comentarle, pienso que es la forma más rápida de empezar a trabajar con esta tecnología. Las operaciones básicas cuando se trabaja con bases de datos son listados, actualizaciones, inserciones y borrados de datos. Si lo que se quiere hacer es mostrar los resultados de una base de datos como una secuencia de datos de sólo lectura y sólo hacia delante, se puede ejecutar un comando y recuperar los resultados mediante el objeto DataReader. Pero si queremos realizar operaciones como enlazar datos, recorrer datos o incluso la configuración remota de los resultados de una consulta lo mejor es utilizar un objeto DataSet.

Llenar un objeto DataSet

Lo primero que tenemos que hacer es crear el objeto sqlDataAdapter .

```
Dim myConnection as SqlConnection
```

```
Dim mySqlDataAdapter as SqlDataAdapter
```

```
myConnection = new SqlConnection("server=Miservidor;Trusted_Connection=yes;database=nombredb")
```

```
mySqlDataAdapter = new SqlDataAdapter("select * from mitabla", myConnection)
```

También se puede crear a través de un comando.

```
Dim miComando as SqlCommand = New SqlCommand ("select * from mitabla", myConnection)
```

```
Dim mySqlDataAdapter as SqlDataAdapter = new SqlDataAdapter (miComando)
```

El objeto SqlDataAdapter tiene dos métodos **Fill** y **Update**. El método Fill llena un objeto DataSet y el método Update recoge los cambios realizados en un objeto DataSet y los hace efectivos en el destino.

Cuando se haya establecido el objeto SqlDataAdapter, se le podrá pasar un objeto DataSet para llenarlo:

```
Dim myDataSet as DataSet = new DataSet()
```

```
mySqlDataAdapter.Fill(myDataSet,"MiColumna")
```

Como los datos se almacenan en una colección de filas de la tabla, es posible utilizar fácilmente una instrucción **foreach** para recorrer las filas:

```
Dim myDataRow as DataRow
```

```
For Each myDataRow In myDataSet.Tables("MiTabla").Rows
```

```
    Console.WriteLine(myDataRow("MiColumnaId").ToString())
```

```
Next
```

El código completo es el siguiente:

```
Imports System
```

```
Imports System.Data
```

```
Imports System.Data.SqlClient
```

```
namespace LlenarDataset
```

```
public class gettingdata
```

```
public shared sub Main()
```

```
    Dim mygettingdata as gettingdata
```

```
    mygettingdata = new gettingdata()
```

```
    mygettingdata.Run()
```

```
end sub
```

```
public sub Run()
```

```
    Dim myConnection as SqlConnection
```

```
    Dim mySqlDataAdapter as SqlDataAdapter
```

```

try

    Dim myDataSet as DataSet = new DataSet()

    myConnection = new SqlConnection("server=MiServidor;Trusted_Connection=yes;database=nombrebddd")

    mySqlDataAdapter = new SqlDataAdapter("select * from mitabla", myConnection)

    mySqlDataAdapter.Fill(myDataSet,"MiColumna")

    Dim myDataRow as DataRow

    For Each myDataRow In myDataSet.Tables("MiTabla").Rows

        Console.WriteLine(myDataRow("MiColumnaId").ToString())

    Next

    catch e as Exception

        Console.WriteLine(e.ToString())

    end try

end sub

end class

end namespace

```

Convertir una aplicación ADO en una aplicación ADO.NET

Tenemos que tener en cuenta que un objeto Recordset de ADO sólo contiene datos para una tabla, a diferencia de un objeto DataSet de ADO.NET que puede contener varias tablas y es bastante más flexible.

La secuencia básica de una aplicación que utilice ADO es una conexión a la base de datos, se ejecuta una consulta SQL que devuelve un objeto Recordset de ADO y si es una consulta pues se recorre este objeto para mostrar los datos recuperados al usuario.

`Conexión con la base de datos

```
cn.Open("server=nombre_servidor;Trusted_Connection=yes;database=mibaseDatos;provider=sqloledb")
```

`Abrimos el RecordSet

```
Set rst = New ADODB.Recordset
```

```
Rst.Open "select * from MiTabla", cn, adOpenKeyset, adLockPessimistic
```

`Nos situamos en el primer registro para recorrer los datos

```
rst.MoveFirst
```

Los pasos a seguir en .NET son muy parecidos, establecemos una conexión con la base de datos mediante un objeto SqlConnection, se utiliza el objeto Reader para recorrer los datos devueltos cuando se ejecuta la consulta y este objeto DataReader de sólo lectura y sólo hacia delante ofrece mejor rendimiento el de ADO. Al igual que se llenaba antes el objeto RecordSet aquí se tiene que llenar el objeto DataSet mediante SqlDataAdapter y el método Fill de este último objeto.

`Establecemos la conexión

```
Dim myConnection as SqlConnection = new SqlConnection("server=MiServidor;Trusted_Connection=yes;database=nombrebdd")
```

`Utilizamos el objeto SqlDataAdapter

```
Dim mySqlDataAdapter as SqlDataAdapter = new SqlDataAdapter("select * from mitabla", myConnection)
```

`Llenamos el DataSet

```
Dim myDataSet as DataSet = new DataSet()
```

```
mySqlDataAdapter.Fill(myDataSet, "MiColumna")
```

En una aplicación ADO si ejecutamos una consulta existirá alguna función que se desplaza por el objeto Recordset para obtener datos de cada registro y cada combinación de campos. Se utilizan los métodos **MoveFirst**, **MoveLast**, **MovePrevious** y **MoveNext** del objeto **Recordset**.

Para recorrer un objeto DataSet en ADO.NET se asigna a un objeto **DataView** y se recorre el objeto DataView resultante para extraer los datos. Utilizando el objeto DataView es posible convertir cualquier tabla de un objeto DataSet en un objeto que funcione de forma similar al antiguo objeto Recordset de ADO.

```
Dim myDataView as DataView = new DataView (myDataSet.Tables("MiColumna"))
```

Y se recorrería con un bucle como se hace con el antiguo Recordset.

```
Dim cont as integer
```

```
For cont = 0 to myDataView.Count -1
```

```
    Console.WriteLine(MyDataView(cont)("campo"))
```

```
    .....
```

```
next
```

[Sobre el Autor ...](#)

Si desea contratar formación, consultoría o desarrollo de piezas a medida puede contactar con

J2EE, EJBs, Struts...

[Autentia S.L.](#) Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto

Descripción

[MySql en Windows](#)

MySql es una de las principales bases de datos "gratuitas" que podemos encontrar en Internet. En este tutorial aprenderéis a instalarlo en Windows

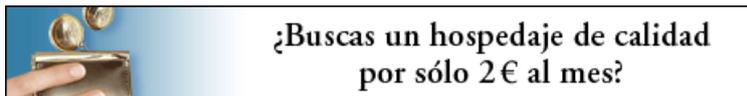
Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)



www.AdictosAlTrabajo.com Optimizado 800X600