

Tema 11; Acceso conectado a BBDD

ActiveX Data Object

- ADO.NET es la tecnología que las aplicaciones asp.net utilizan para comunicarse con la BD.
- Optimizada para aplicaciones distribuidas (como aplicaciones web).
- Basada en XML
- Entorno conectado vs desconectado

Entorno conectado

- los usuarios están conectados continuamente a una fuente de datos
- es más fácil de mantener
- La concurrencia se controla más fácilmente
- datos estén más actualizados
- Debe existir una conexión de red constante
- Escalabilidad limitada
- Connection: se utilizan para establecer las conexiones al proveedor de datos adecuado (método Open).
- Command: sirven para ejecutar sentencias SQL y procedimientos almacenados.
- System.Data.OleDb y System.Data.SqlClient: clases responsables del acceso a datos desde fuentes SQL Server y OLE DB. Al trabajar con SQL llevarán el prefijo Sql y al emplear Ole DB llevarán OleDb
- Para recuperar datos de la BD necesitas: Una sentencia SQL, Un Command que ejecute la SQL y Un DataReader que capture los registros recuperados
- En caso de insertar, actualizar o borrar datos de la BD necesitas: Un Command para ejecutar la SQL
- ExecuteNonQuery obtiene el número de registros afectados. Ejecuta un comando y no devuelve ningún resultado
- try/catch para el manejo de excepciones

DataReader

- proporciona un flujo de datos firme
- un cursor de sólo lectura que avanza por los registros sólo hacia delante.
- Mantienen una conexión viva con el origen de datos, pero no permiten realizar ningún cambio.
- ExecuteReader Ejecuta un comando y devuelve un comando que implementa DataReader (Permite iterar a partir de los registros recibidos)
- .Read() recupera una fila. True si se ha recuperado una fila de información correctamente, False si hemos intentado leer después del final del conjunto de resultados
- Podemos acceder a los valores de una fila con el nombre del campo
- .Close() cierra el DataReader
- Más ligero y veloz que DataSet. Mejor elección para acceso a datos simple
- Recomendado para consultas de sólo lectura utilizadas una única vez (listar todos los artículos de un almacén)

Espacios de nombres

System.Data	
System.Data.Common	
System.Data.OleDb	Ms access, Oracle.. DB
System.Data.SqlClient	Ms SQL Server 7.0 DB
System.Data.SqlTypes	contiene clases para trabajar con tipos de datos nativos de SQL Server
System.Configuration	proporciona clases para trabajar con información de configuración almacenada en archivos de configuración.

Parámetros de conexión

Connection Timeout	Define el tiempo de espera máximo que debe esperar una conexión para intentar conectar con éxito con el servidor de base de datos. En caso de superar este tiempo se genera una excepción. El tiempo por defecto definido es de 15 segundos.
Data Source	Recibe el nombre del servidor SQL Server utilizado en la conexión, o en caso de utilizar bases de datos de usuario Access el nombre del archivo utilizado.
Initial Catalog / Database	Nombre de la base de datos con la que vamos a trabajar.
Integrated Security	Configura nuestra conexión de un modo seguro o no. Recibe como valores True, False y SSPI, siendo True y SSPI el mismo modo de seguridad.
AttachDBFilename	Si se utiliza un nombre de archivo para conectar con la base de datos, se simplifica la implementación de la base de datos con la aplicación (especificamos el archivo mdf)
Persist Security Info	Si recibe el valor True, se devuelve la contraseña junto con la conexión si ha sido abierta o permanece abierta, esto supone un riesgo de seguridad, por lo que se suele dejar como está configurada por defecto, False.
Password	Contraseña para la identificación de inicio de sesión en SQL Server.
Provider	Utilizada únicamente para conexiones OleDbConnection, establece o devuelve el nombre del proveedor.
User ID	Nombre de usuario para el inicio de sesión en SQL Server 2005, login.

Proveedores

- SQLOLEDB: proveedor OLEDB de SQL
- MSDAORA: proveedor OLEDB para una bd Oracle
- Microsoft.Jet.OLEDB.4.0: proveedor OLEDB de Access

Comandos de conexión

- Crear una conexión

```
string s =
    "datasource=.\SQLEXPRESS;IntegratedSecurity=SSPI;AttachDBFilename=|DataDirectory|\\Database1.mdf;Use
    r Instance=true";
```

```
SqlConnection c=new SqlConnection(s);
```

- .Open() / .Close() abren / cierran la conexión con la BD

DataDirectory

- es una cadena de sustitución que indica la ruta de acceso de la base de datos.
- elimina la necesidad de definir la ruta de acceso completa. |DataDirectory|
- se establece en AppDomain llamando a AppDomain.SetData. AppDomain.CurrentDomain.SetData("DataDirectory", newpath);
- Si no se establece DataDirectory Para las aplicaciones que se coloquen en un directorio en el equipo cliente, la ruta de acceso de la base de datos será la carpeta en la que se coloque la aplicación. Para aplicaciones Web, se tendrá acceso a la carpeta App_Data.

web.config

- Archivo de configuración de la aplicación ASP.NET basado en XML
- Incluye las opciones del proyecto
- Para evitar almacenar cadenas en el código, se puede almacenar en el archivo web.config.
- La cadena de conexión se puede almacenar en el archivo de configuración en el elemento <connectionStrings> donde el nombre se puede utilizar para buscar el valor almacenado en el atributo connectionString en tiempo de ejecución