2011



СООКВООК

Standalone ASP.NET Membership Database

Web Site Factory and other projects integrate *ASP.NET Membership*, a built-in way to store and validate user credentials. You can enable *ASP.NET Membership* by selecting the membership option in the code generator project wizard. This will enable numerous membership features including a fly-over sign-in window, self-service membership enrollment, membership bar, and membership manager.

The configuration of your project will be automatically changed to support the default membership provider available in ASP.NET. This provider defines a connection string that points to a local instance of *Microsoft SQL Server Express*. The provider will automatically connect to the server and dynamically create a database to maintain users, roles, and other membership features. The database will be created under ~/*App_Data* folder of your project.

This works great on a development machine with installed *SQL Server Express*. There are many situations when you want to use a standalone membership database or store ASP.NET membership data structures directly in your own database.



Project wizard offers an option that will enable a standalone membership database configuration. Here is the screen shot of the project wizard with the standalone membership database enabled.

The connection string in the screen shot looks as follows:

Data Source=db;Initial Catalog=Membership;Integrated Security=True;

We have configured the standalone membership database with the name *aspnetdb*.

You can read more about the configuration process at <u>http://msdn.microsoft.com/en-us/library/ms229862(VS.80).aspx</u>.

These are the steps that we have taken to create the *aspnetdb* database:

1. We have started *aspnet_regsql.exe* from *Windows Explorer* as shown in picture.

rganize 🔻 📑 Open 🔻	Burn New folder			!≡ ▼ 🚺
퉬 Help 🔷	Name	Date modified	Туре	Size
🐌 IME	InstallUtil.exe	3/18/2010 1:16 PM	Application	28 KB
🍌 inf	🔲 ilasm.exe	3/18/2010 4:47 PM	Application	288 KB
🍌 L2Schemas	EdmGen.exe	3/18/2010 4:47 PM	Application	83 KB
LiveKernelReports	🔲 dfsvc.exe	3/18/2010 1:16 PM	Application	12 KB
🍌 Logs	DataSvcUtil.exe	3/18/2010 4:47 PM	Application	59 KB
🍌 Media	Cvtres.exe	3/18/2010 1:16 PM	Application	31 KB
Microsoft.NET	Csc.exe	3/18/2010 1:16 PM	Application	1,927 KB
assembly	ComSvcConfig.exe	3/18/2010 4:47 PM	Application	160 KB
🍌 authman	CasPol.exe	3/18/2010 1:16 PM	Application	93 KB
퉬 DirectX for Manager 😑	📧 aspnet_wp.exe	3/18/2010 4:47 PM	Application	32 KB
J Framework	aspnet_state.exe	3/18/2010 4:47 PM	Application	35 KB
URTInstallPath_GA	aspnet_regsql.exe	3/18/2010 4:47 PM	Application	101 KB
↓ v1.0.3705	aspnet_regiis.exe	3/18/2010 4:47 PM	Application	30 KB
July v1.1.4322	aspnet_regbrowsers.exe	3/18/2010 4:47 PM	Application	20 KB
V2.0.50727	aspnet_compiler.exe	3/18/2010 4:47 PM	Application	30 KB
🤳 v3.0	AppLaunch.exe	3/18/2010 1:16 PM	Application	104 KB
📕 v3.5	AddInUtil.exe	3/18/2010 1:16 PM	Application	29 KB
V4.0.30319	AddInProcess32.exe	3/18/2010 1:16 PM	Application	29 KB
뷁 Framework64 🖕	AddInProcess.exe	3/18/2010 1:16 PM	Application	29 KB

The path to your instance of *aspnet_regsql.exe*:

C:\%windir%\Microsoft.NET\Framework\<versionNumber>\aspnet_regsql.exe

2. We have clicked *Next* button in *ASP.NET SQL Server Setup Wizard*:

🏘 ASP.NET SQL Server Setup Wizard					
Welcome to the ASP.NET SQL Server Setup Wizard					
This wizard creates or configures a SQL Server database that stores information for ASP.NET applications services (membership, profiles, role management, personalization and SQL Web event provider).					
To configure the database for these features individually or for additional features such as session state or SQL cache dependency, run aspnet_regsql at the command line. For help with command line options, use the "-?" switch.					
Click Next to continue.					
< <u>P</u> revious <u>N</u> ext > <u>E</u> inish	Cancel				

3. We have continued to the next step to configure *SQL Server* for application services:

🗣 ASP.NET SQL Server Setup Wizard
SQL Select a Setup Option
What database task do you want to perform?
Onfigure SQL Server for application services
This option runs a script that creates a new database or configures an existing database to store information for ASP.NET membership, profiles, role management, personalization and SQL Web event provider.
\bigcirc <u>R</u> emove application services information from an existing database
This option removes information about ASP.NET membership, profiles, role management, personalization and SQL Web event provider from the database. Note: This process cannot be undone.
Note: To configure the database for additional features such as session state or SQL cache dependency, run aspnet_regsql at the command line. For help with command line options, use the "-?" switch.
< <u>Previous</u> <u>Next</u> > <u>Finish</u> Cancel

4. We have entered "." as a server name and "aspnetdb" as database name.

🗳 ASP.NET SQL	Server Setup Wizard	٢
SQL Sel	lect the Server and Database	
Specify the SQL connecting to the	Server name, database name to create or remove, and the credentials to use when the database.	
Note:	The credentials must identify a user account that has permissions to create or remove a database.	
<u>S</u> erver:		
• <u>W</u> indows aut	hentication	
⊖ S <u>Q</u> L Server a	authentication	
<u>U</u> ser name:		
P <u>a</u> ssword:		
<u>D</u> atabase:	aspnetdb 👻	
	< <u>Previous</u> <u>N</u> ext > <u>Finish</u> Cancel]

A few more clicks on the *Next* button have done the job for us. The database has been created. We have returned to the project wizard of our code generation project and configure the provider name and connection string of the newly created membership database.