

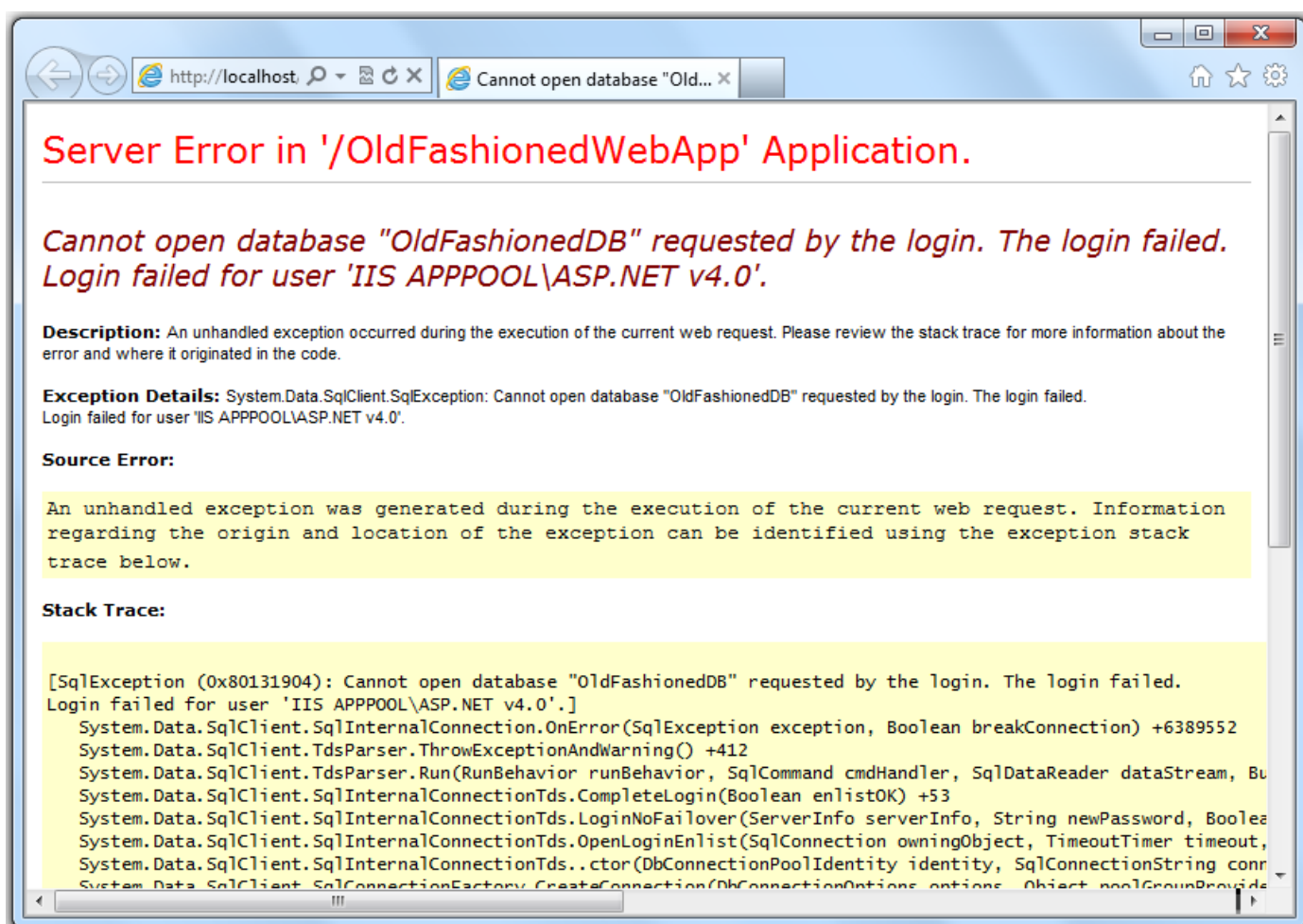
در [قسمت قبلی](#) این مقاله گفتیم که دو خاصیت از LocalDb هنگام استفاده از Full IIS باعث بروز خطا می شوند:

LocalDb نیاز دارد که پروفایل کاربر بارگذاری شده باشد
بصورت پیش فرض وهله LocalDb متعلق به یک کاربر بوده، و خصوصی است

در قسمت قبل دیدیم چگونه باید پروفایل کاربر را بدرستی بارگذاری کنیم. در این مقاله به مالکیت وهله ها (instance ownership) می پردازیم.

مشکل وهله خصوصی

در پایان قسمت قبلی، اپلیکیشن وب را در این حالت رها کردیم:

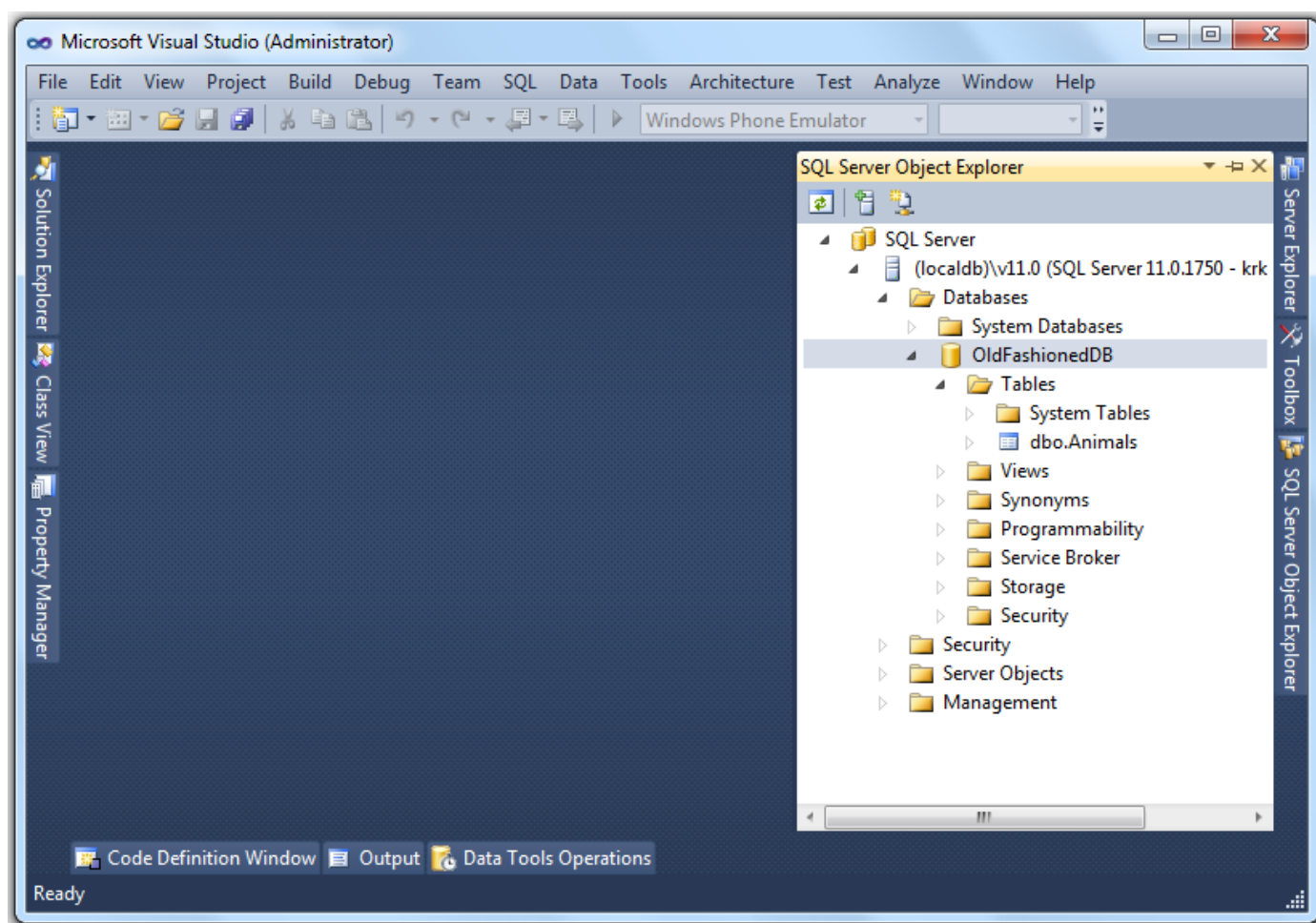


همانطور که مشاهده می کنید با خطای زیر مواجه هستیم:

System.Data.SqlClient.SqlException: Cannot open database "OldFashionedDB" requested by the login. The login failed.

'Login failed for user 'IIS APPPOOL\ASP.NET v4.0

این بار پیغام خطا واضح و روشن است. LocalDb با موفقیت اجرا شده و اپلیکیشن وب هم توانسته به آن وصل شود، اما این کانکشن سپس قطع شده چرا که دسترسی به هله جاری وجود نداشته است. اکانت ApplicationPoolIdentity (در اینجا IIS APPPOOL\ASP.NET v4.0) نتوانسته به دیتابیس LocalDb وارد شود، چرا که دیتابیس مورد نظر در رشته اتصال اپلیکیشن (OldFashionedDB) وجود ندارد. عجیب است، چرا که وصل شدن به همین دیتابیس با رشته اتصال جاری در ویژوال استودیو با موفقیت انجام می‌شود.



همانطور که در تصویر بالا مشاهده می‌کنید از ابزار SQL Server Object Explorer استفاده شده است. این ابزار توسط [SQL Server Data Tools](#) معرفی شد و در نسخه‌های بعدی ویژوال استودیو هم وجود دارد و توسعه یافته است. چطور ممکن است ویژوال استودیو براحتمی بتواند به دیتابیس وصل شود، اما اپلیکیشن وب ما با همان رشته اتصال نمی‌تواند دیتابیس را باز کند؟ در هر دو صورت رشته اتصال ما بدین شکل است:

```
Data Source=(localdb)\v11.0;Initial Catalog=OldFashionedDB;Integrated Security=True
```

پاسخ این است که در اینجا، دو هله از LocalDb وجود دارد. برخلاف هله‌های SQL Server Express که بعنوان سرویس‌های ویندوزی اجرا می‌شوند، هله‌های LocalDb بصورت پروسس‌های کاربری (user processes) اجرا می‌شوند. هنگامی که کاربران مختلفی سعی می‌کنند به LocalDb متصل شوند، برای هر کدام از آنها پروسس‌های مجزایی اجرا خواهد شد. هنگامی که در ویژوال استودیو به (localdb)\v11.0 وصل می‌شویم، هله ای از LocalDb ساخته شده و در حساب کاربری ویندوز جاری اجرا می‌شود.

اما هنگامی که اپلیکیشن وب ما در IIS می‌خواهد به همین دیتابیس وصل شود، وهله دیگری ساخته شده و در **ApplicationPoolIdentity** اجرا می‌شود. گرچه ویژوال استودیو و اپلیکیشن ما هر دو از یک رشته اتصال استفاده می‌کنند، اما در عمل هر کدام به وهله‌های متفاوتی از LocalDb دسترسی پیدا خواهند کرد. پس مسلماً دیتابیزی که توسط وهله ای در ویژوال استودیو ساخته شده است، برای اپلیکیشن وب ما در IIS در دسترس نخواهد بود.

یک مقایسه خوب از این وضعیت، پوشه **My Documents** در ویندوز است. فرض کنید در ویژوال استودیو کدی بنویسیم که در این پوشه یک فایل جدید می‌سازد. حال اگر با حساب کاربری دیگری وارد ویندوز شویم و به پوشه My Documents برویم این فایل را نخواهیم یافت. چرا که پوشه My Documents برای هر کاربر متفاوت است. بهمین شکل، وهله‌های LocalDb برای هر کاربر متفاوت است و به پروسس‌ها و دیتابیس‌های مختلفی اشاره می‌کنند.

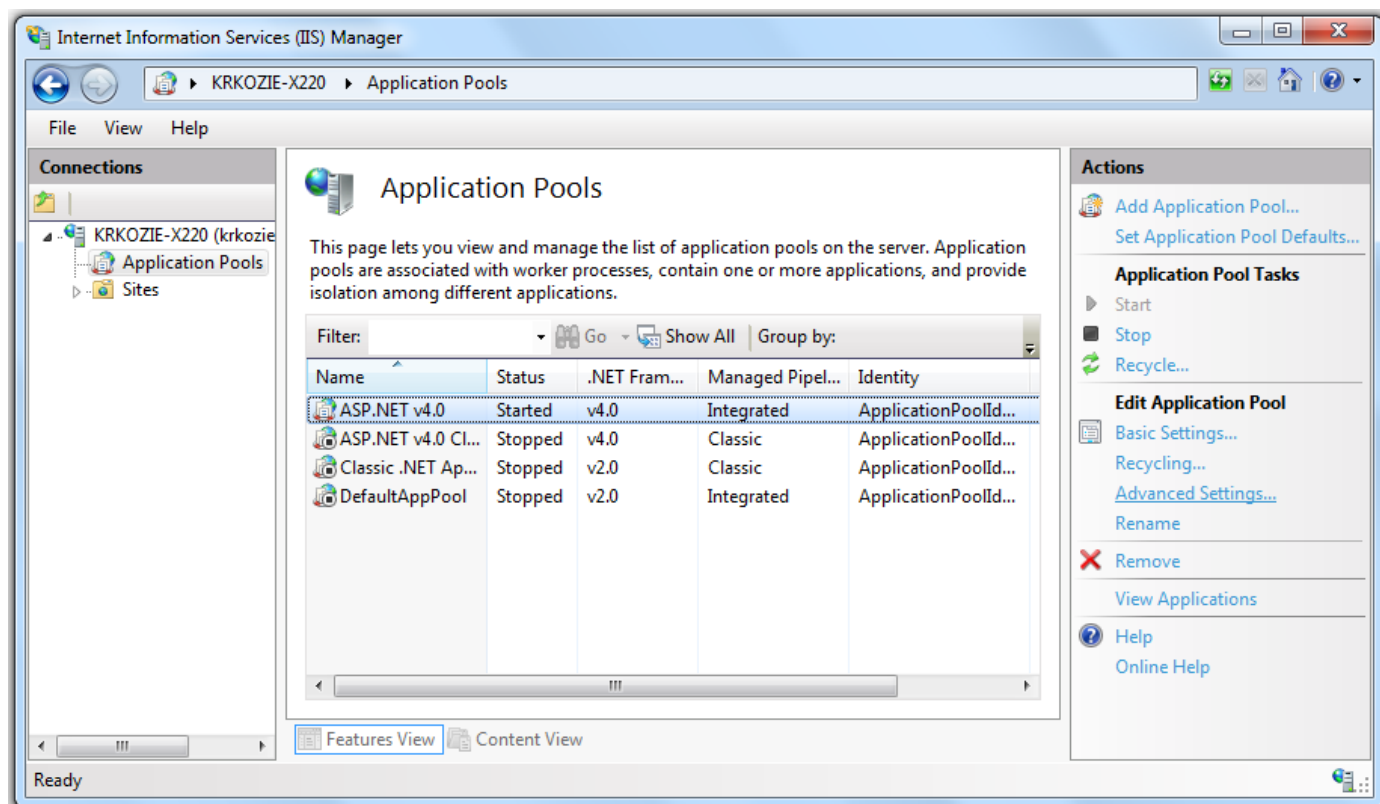
به همین دلیل است که اپلیکیشن وب ما می‌تواند بدون هیچ مشکلی روی IIS Express اجرا شود و دیتابیس را باز کند. چرا که IIS Express درست مانند LocalDb یک پروسس کاربری است. IIS Express توسط ویژوال استودیو راه اندازی می‌شود و روی حساب کاربری جاری اجرا می‌گردد، پس پروسس آن با پروسس خود ویژوال استودیو یکسان خواهد بود و هر دو زیر یک اکانت کاربری اجرا خواهند شد.

راه حل ها

درک ماهیت مشکل جاری، راه حال‌های مختلفی را برای رفع آن بدست می‌دهد. از آنجا که هر راه حل مزایا و معایب خود را دارد، بجای معرفی یک راه حال واحد چند راهکار را بررسی می‌کنیم.

رویکرد 1: اجرای IIS روی کاربر جاری ویندوز

اگر مشکل، حساب‌های کاربری مختلف است، چرا خود IIS را روی کاربر جاری اجرا نکنیم؟ در این صورت ویژوال استودیو و اپلیکیشن ما هر دو به یک وهله از LocalDb وصل خواهند شد و همه چیز بدرستی کار خواهد کرد. ایجاد تغییرات لازم نسبتاً ساده است. IIS را اجرا کنید و Application Pool مناسب را انتخاب کنید، یعنی همان گزینه که برای اپلیکیشن شما استفاده می‌شود.



قسمت Advanced Settings را باز کنید:

Advanced Settings

(General)

.NET Framework Version	v4.0
Enable 32-Bit Applications	False
Managed Pipeline Mode	Integrated
Name	ASP.NET v4.0
Queue Length	1000
Start Automatically	True

CPU

Limit	0
Limit Action	NoAction
Limit Interval (minutes)	5
Processor Affinity Enabled	False
Processor Affinity Mask	4294967295

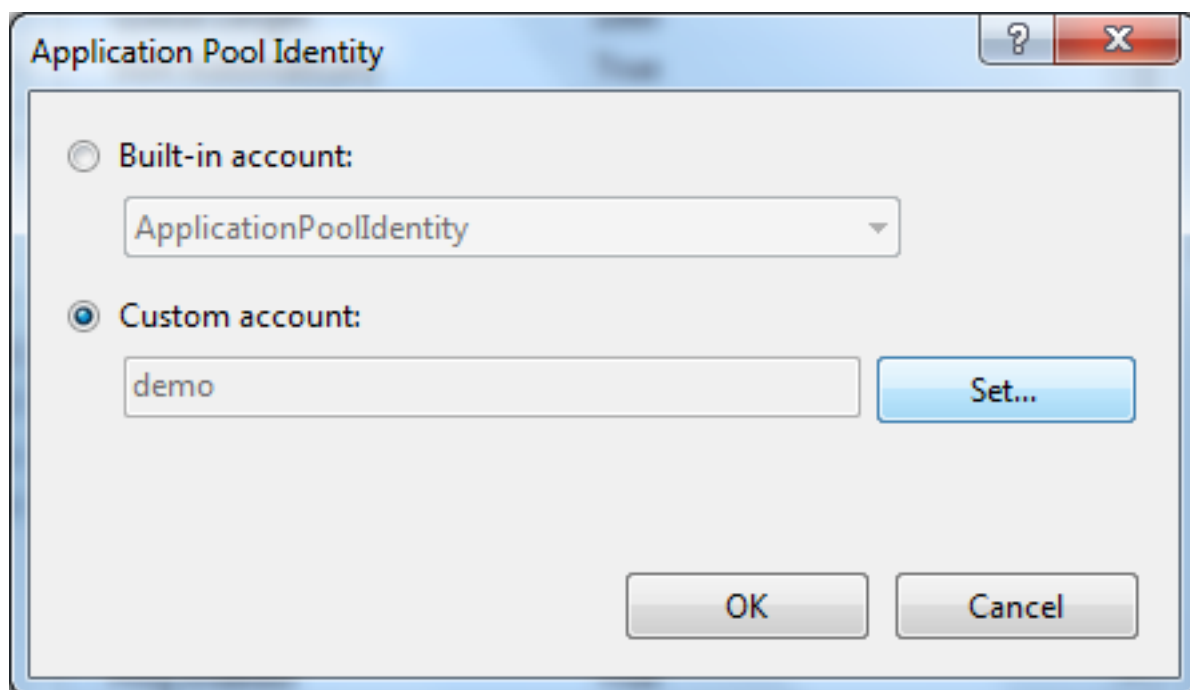
Process Model

Identity	ApplicationPoolIdentity
Idle Time-out (minutes)	20
Load User Profile	True
Maximum Worker Processes	1
Ping Enabled	True

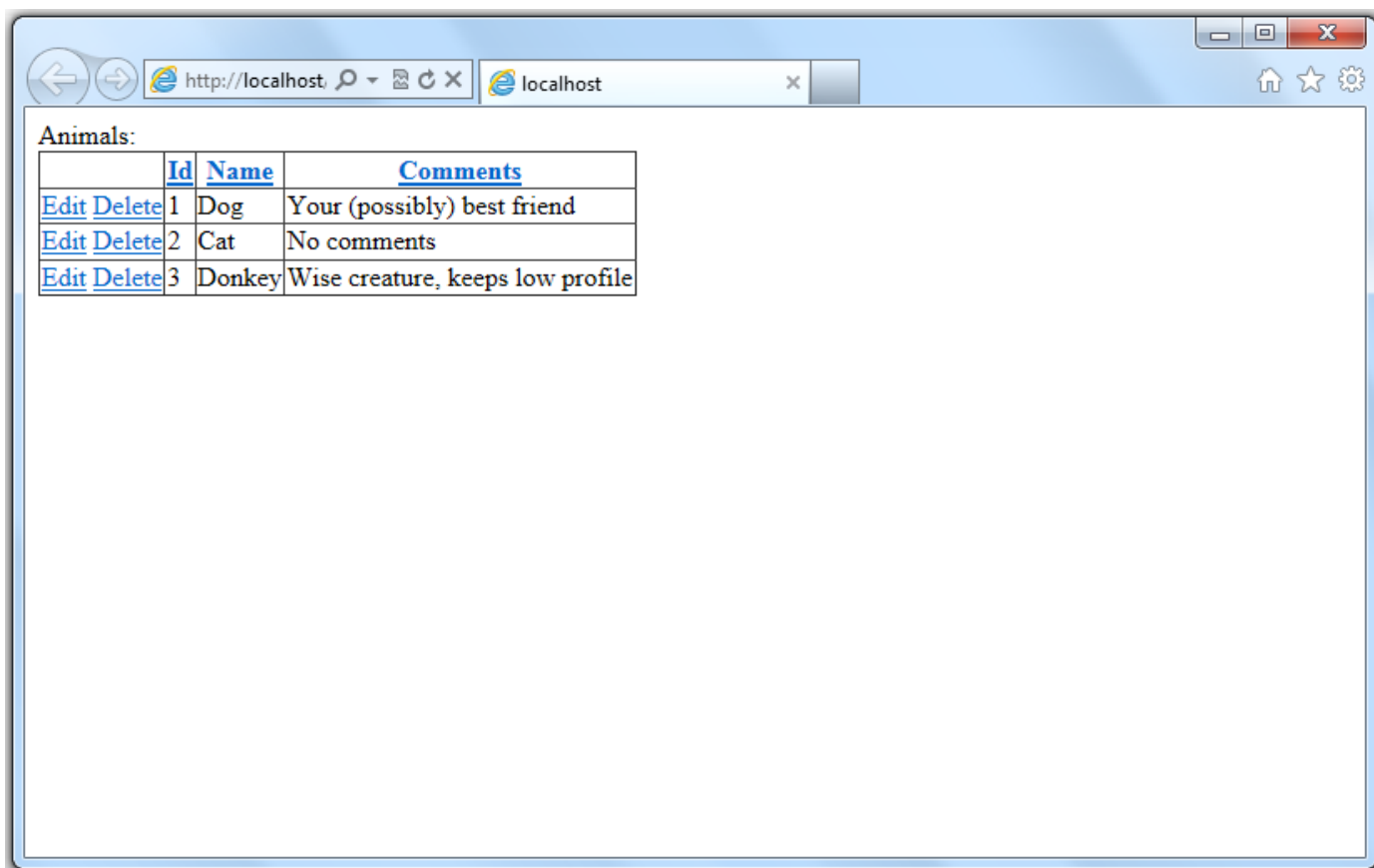
Identity
 [identityType, username, password] Configures the application pool to run as built-in account, i.e. Application Pool Identity (recommended), Network Service, Local System, Local Service, or as a specific user identity.

OK Cancel

روی دکمه سه نقطه کنار خاصیت **Identity** کلیک کنید تا پنجره Application Pool Identity باز شود:



در این قسمت می‌توانید از حساب کاربری جاری استفاده کنید. روی دکمه Set کلیک کنید و نام کاربری و رمز عبور خود را وارد نمایید. حال اگر اپلیکیشن را مجدداً اجرا کنید، همه چیز باید بدرستی اجرا شود.



خوب، معایب این رویکرد چیست؟ مسلماً اجرای اپلیکیشن وب روی اکانت کاربری جاری، ریسک‌های امنیتی متعددی را معرفی می‌کند. اگر کسی بتواند اپلیکیشن وب ما را هک کند، به تمام منابع سیستم که اکانت کاربری جاری به آنها دسترسی دارد، دسترسی خواهد داشت. اما اجرای اپلیکیشن مورد نظر روی ApplicationPoolIdentity امنیت بیشتری را ارائه می‌کند، چرا که اکانت‌های ApplicationPoolIdentity دسترسی بسیار محدودتری به منابع سیستم محلی دارند. بنابراین استفاده از این روش بطور کلی توصیه نمی‌شود، اما در سناریوهای خاصی با در نظر داشتن ریسک‌های امنیتی می‌تواند رویکرد خوبی باشد.

رویکرد 2: استفاده از و هله مشترک

یک راه حال دیگر استفاده از قابلیت instance sharing است. این قابلیت به ما این امکان را می‌دهد تا یک و هله LocalDb را بین کاربران یک سیستم به اشتراک بگذاریم. و هله به اشتراک گذاشته شده، توسط یک نام عمومی (public name) قابل دسترسی خواهد بود.

ساده‌ترین راه برای به اشتراک گذاشتن و هله‌های LocalDb استفاده از ابزار [SqlLocalDB.exe](#) است. بدین منظور Command Prompt را بعنوان مدیر سیستم باز کنید و فرمان زیر را اجرا نمایید:

```
sqllocaldb share v11.0 IIS_DB
```

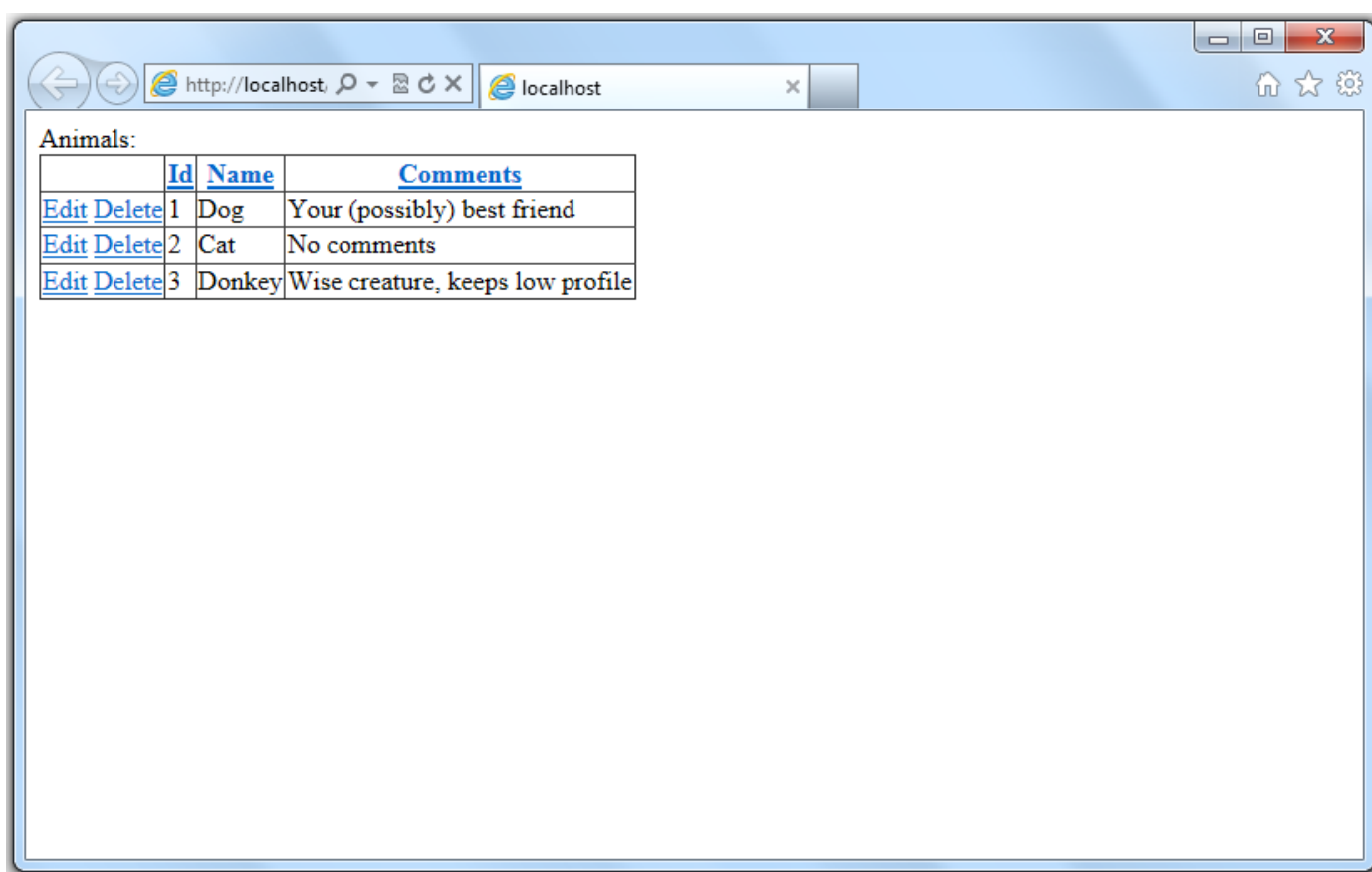
این فرمان و هله خصوصی LocalDb را با نام عمومی **IIS_DB** به اشتراک می‌گذارد. حال تمام کاربران سیستم می‌توانند با آدرس **(localdb)\.IIS_DB** به این و هله وصل شوند. این فرمت آدرس دهی سرور دیتابیس، مشخص می‌کند که از یک و هله shared استفاده می‌کنیم. رشته اتصال جدید مانند لیست زیر خواهد بود:

```
Data Source=(localdb)\.IIS_DB;Initial Catalog=OldFashionedDB;Integrated Security=True
```

پیش از آنکه اپلیکیشن وب ما بتواند به این وهله متصل شود، باید لاگین‌های مورد نیاز برای ApplicationPoolIdentity را ایجاد کنیم. راه اندازی وهله ساده است، کافی است دیتابیس را در SQL Server Object Explorer باز کنید. این کار اتصالی به دیتابیس برقرار می‌کند و آن را زنده نگاه می‌دارد. برای ایجاد لاگین مورد نظر، می‌توانیم در SQL Server Object Explorer یک کوئری اجرا کنیم:

```
create login [IIS APPPOOL\ASP.NET v4.0] from windows;  
exec sp_addsrvrolemember N'IIS APPPOOL\ASP.NET v4.0', sysadmin
```

اسکرپت بالا به اکانت ApplicationPoolIdentity سطح دسترسی کامل می‌دهد. در صورت امکان بهتر است از سطوح دسترسی محدودتری استفاده کنید، مثلاً دسترسی به دیتابیس یا جداولی مشخص. حالا می‌توانید اپلیکیشن را مجدداً اجرا کنید و همه چیز بدون خطا باید کار کند.



معایب این روش چیست؟ مشکل اصلی در این رویکرد این است که پیش از آنکه اپلیکیشن ما بتواند به وهله مشترک دسترسی داشته باشد، باید وهله مورد نظر را راه اندازی و اجرا کنیم. بدین منظور، حساب کاربری ویندوزی که مالکیت وهله را دارد باید به آن وصل شود و کانکشن را زنده نگه دارد، در غیر اینصورت وهله LocalDb قابل دسترسی نخواهد بود.

رویکرد 3: استفاده از SQL Server Express

از آنجا که نسخه کامل SQL Server Express بعنوان یک سرویس ویندوزی اجرا می‌شود، شاید بهترین راه استفاده از همین روش باشد. کافی است یک نسخه از SQL Server Express را نصب کنیم، دیتابیس مورد نظر را در آن بسازیم و سپس به آن متصل شویم. برای این کار حتی می‌توانید از ابزار جدید SQL Server Data Tools استفاده کنید، چرا که با تمام نسخه‌های SQL Server سازگار است. در صورت استفاده از نسخه‌های کامل تر، رشته اتصال ما بدین شکل تغییر خواهد کرد:

```
Data Source=.\SQLEXPRESS;Initial Catalog=OldFashionedDB;Integrated Security=True
```

مسلما در این صورت نیز، لازم است اطمینان حاصل کنیم که ApplicationPoolIdentity به وهله SQL Server Express دسترسی کافی دارد. برای این کار می‌توانیم از اسکریپت قبلی استفاده کنیم:

```
create login [IIS APPPOOL\ASP.NET v4.0] from windows;  
exec sp_addsrvrolemember N'IIS APPPOOL\ASP.NET v4.0', sysadmin
```

حال اجرای مجدد اپلیکیشن باید با موفقیت انجام شود. استفاده از این روش مسلما امکان استفاده از LocalDb را از ما می‌گیرد. ناگفته نماند که وهله‌های SQL Server Express همیشه در حال اجرا خواهند بود چرا که بصورت سرویس‌های ویندوزی اجرا می‌شوند. همچنین استفاده از این روش ممکن است شما را با مشکلاتی هم مواجه کند. مثلا خرابی رجیستری ویندوز می‌تواند SQL Server Express را از کار بیاندازد و مواردی از این دست. راهکارهای دیگری هم وجود دارند که در این مقاله به آنها نپرداختیم. مثلا می‌توانید از AttachDbFilename استفاده کنید یا از اسکریپت‌های T-SQL برای استفاده از وهله خصوصی ASP.NET کمک بگیرید. اما این روش‌ها دردسرهای زیادی دارند، بهمین دلیل از آنها صرفنظر کردیم.

مطالعه بیشتر درباره LocalDb

[Introducing LocalDB and LocalDB Q&A](#)

[Upgrading .NET Framework 4 to support LocalDB connections and using SQL Server Management Studio to work with](#)

[LocalDB](#)

[Using LocalDB in Visual Studio 2010](#)

[Where are LocalDB database files located](#)

نظرات خوانندگان

نویسنده: رضا گرمارودی
تاریخ: ۲۰:۳۱۳۹۲/۱۰/۲۹

سلام؛ ممنون از مقاله جالبتون. یک سوال داشتم. مقالات شما رو در خصوص Identity و Localdb دنبال کردم. شما تجربه کاری با WindowsAzor داریو و اینجا هم از Localdb صحبت کردین. می‌خواستم برای هر کاربر یک دیتابیس مجزا داشته باشم اما هاست‌ها نهایتاً دو یا سه تا دیتابیس sqlserver در اختیار شما می‌گذارند. LocalDb همان طور که از اسمش برمیاد Local هست یعنی همیشه روی هاست ولو این که از این دیتابیس تنها یک نفر آنلاین استفاده کنه؟ استفاده از windowsAzor چطور؟ مشکل من و حل می‌کنه یا شما راه حل بهتری معرفی می‌کنید؟

نویسنده: محسن خان
تاریخ: ۲۲:۳۱۳۹۲/۱۰/۲۹

[SQL Server CE](#) داخل پروسه برنامه اجرا میشه (و پروسه مجزایی نداره). [نسخه LocalDB](#) خارج از پروسه برنامه اما به صورت یک [child process](#) اجرا میشه. برنامه وب شما می‌تونه با هر تعداد فایل sdf مربوط به SQL CE یا فایل mdf بانک اطلاعاتی LocalDB کار کنه (در رشته اتصالی آن [AttachDbFileName](#) قابل تعریف هست).

نویسنده: رضا گرمارودی
تاریخ: ۱۶:۵۷۱۳۹۲/۱۱/۰۱

Attachdb در رشته اتصالی موجود است و بر روی لوکال مشکلی ندارم. اما بر روی هاست بانک را Attach نمی‌کند. با سه تا هاست مختلف تست گرفتم ولی نشد! ممکنه LocalDb و Sql Server CE تنها بر روی لوکال کار کنند و نه بر روی هاست؟ یا ارائه دهنده سرویس می‌بایست این سرویس‌ها را بر روی سرور خود نصب کنه؟

نویسنده: محسن خان
تاریخ: ۱۷:۵۱۳۹۲/۱۱/۰۱

SQL CE با bin deploy کار می‌کنه و نیازی به نصب نداره (البته نصاب داره؛ ولی مهم و الزامی نیست). فایل‌هاش رو بسته نیوگت زیر به پروژه اضافه می‌کنه:

<http://www.nuget.org/packages/Microsoft.SqlServer.Compact>

اما LocalDB رو باید سرور دار حتما خودش یکبار نصب کنه.