

LocalDB نسخه‌ای جدید از Sql server express است که به توسعه دهندگان این اجازه را می‌دهد تا با نصب آن، از نصب کامل دیگر نسخه‌های Sql server جلوگیری نمایند. LocalDB برای برنامه‌هایی که به صورت Local و بر روی یک سیستم اجرا می‌شوند مورد استفاده قرار می‌گیرد.

## مزایای استفاده از این نسخه

فایل نصب با حجم بسیار کم. (28.2MB برای نسخه 32 بیتی و 33.7MB برای نسخه 64بیتی)

سادگی ( بدون نیاز به انجام تنظیمات خاص بر روی سیستم)

اجرا در محیط‌هایی که کاربر جاری دسترسی مدیریتی ندارد. (برای اجرای آن نیاز به Permission های مدیریتی نیست و یک کاربر سطح پایین هم می‌تواند آن را اجرا کند)

سادگی نصب

همانند Sql server Express سازگاری کاملی با T-Sql دارد. همچنین از Stored Procedure ها ، داده‌های جغرافیایی و مکانی ( geometry and geography ها) ، Triggers و View ها پشتیبانی می‌کند. سازگاری با Provider معمولی Sql server

عدم اجرای سرویس خاصی در حافظه برای مدیریت دیتابیس. پروسس‌های LocalDb هر زمان که نیاز باشد اجرا می‌شوند و هر زمان که به آنها نیاز نداشته باشیم به صورت اتوماتیک متوقف می‌شوند.

پشتیبانی از خصوصیت AttachDbFileName در کانکشن استرینگ جهت استفاده از فایل بانک اطلاعات به صورت مستقیم

سرویس پک‌های جدید جهت LocalDB به راحتی بر روی نسخه موجود نصب میشوند و نسخه قبلی را به روز رسانی میکنند.

نصب یک LocalDB برای همه کاربران یک کامپیوتر

پشتیبانی کامل از Silent Installation

امکان استفاده از آن توسط Asp.net

پشتیبانی از XQuery (XML و XPath) و BLOB

پشتیبانی از Ado.net sync framework

پشتیبانی از LINQ

پشتیبانی از Distributed transactions

کانکشن‌های نامحدود (البته به صورت Local)

### نیازمندی‌های نصب

نیاز به نصب Sql server 2012 native client . این مورد به همراه LocalDB روی سیستم نصب نمیشود

نیاز به دسترسی مدیریتی جهت نصب

140MB فضای خالی دیسک سخت

به روز رسانی دات نت فریم ورک 4 به [4.0.2](#) و یا نسخه‌های بالاتر

### محدودیت ها

عدم پشتیبانی از Windows 2000 و Windows xp ، Window server 2003

عدم امکان نصب نسخه 32 بیتی بر روی ویندوز 64 بیتی (حتما باید نسخه 64 بیتی آن را نصب کنید)

فقط می‌توان به صورت Local از آن استفاده کرد. امکان استفاده تحت شبکه وجود ندارد و فقط به کانکشن‌های Local پاسخ می‌دهد.

فقط توسط sql server 2012 management studio در دسترس می‌باشد. LocalDB را نمی‌توان از طریق Management studio های قدیمی مدیریت کرد.

عدم پشتیبانی از Visual Studio 2010 از LocalDB

عدم اجرا بر روی موبایل‌های هوشمند

محدودیت سایز بانک اطلاعات : 10GB

عدم پشتیبانی از قابلیت FileStream

محدودیت استفاده از فقط یک CPU

عدم امکان Debugging دستورات Sql در هنگام اتصال به LocalDB

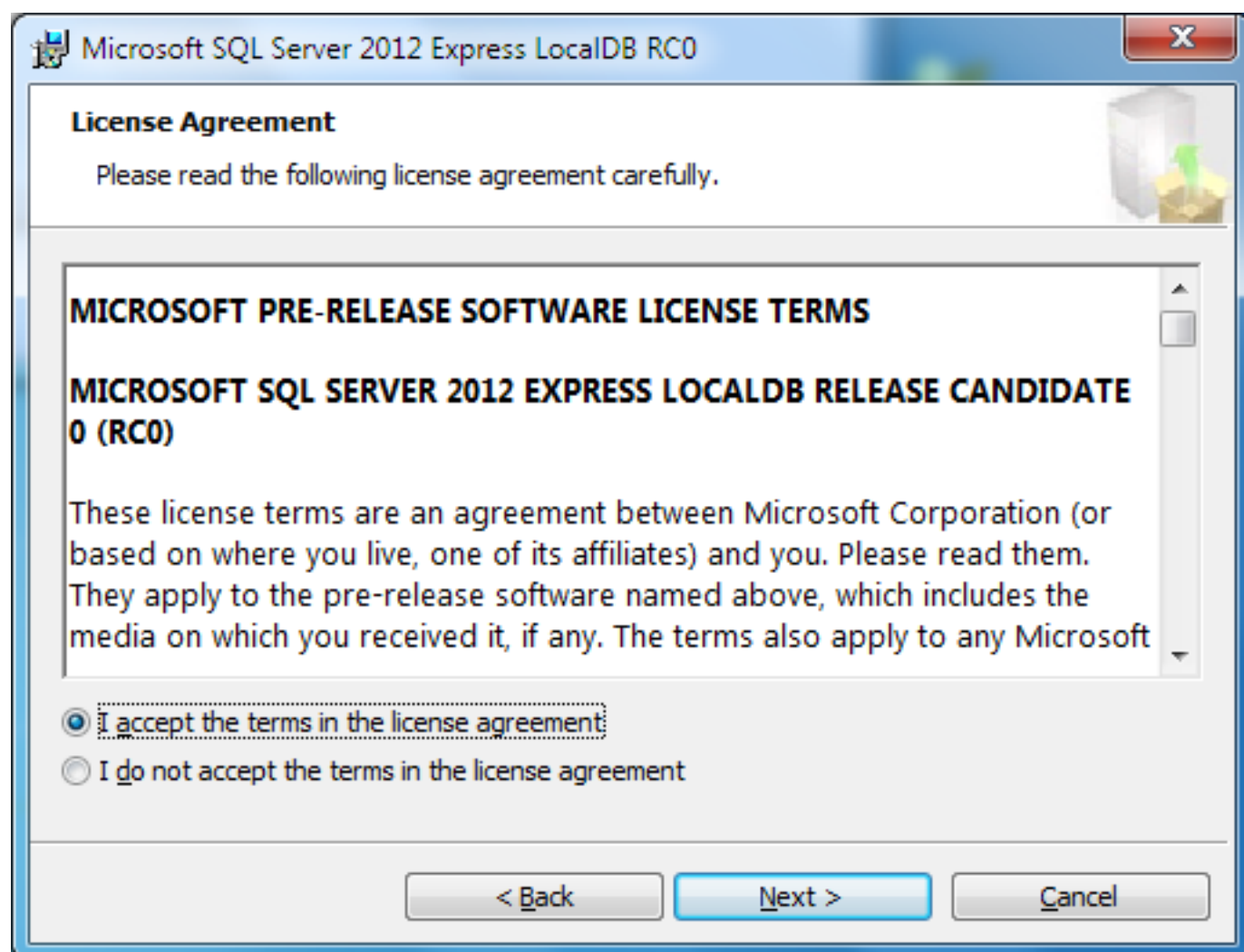
### نحوه نصب

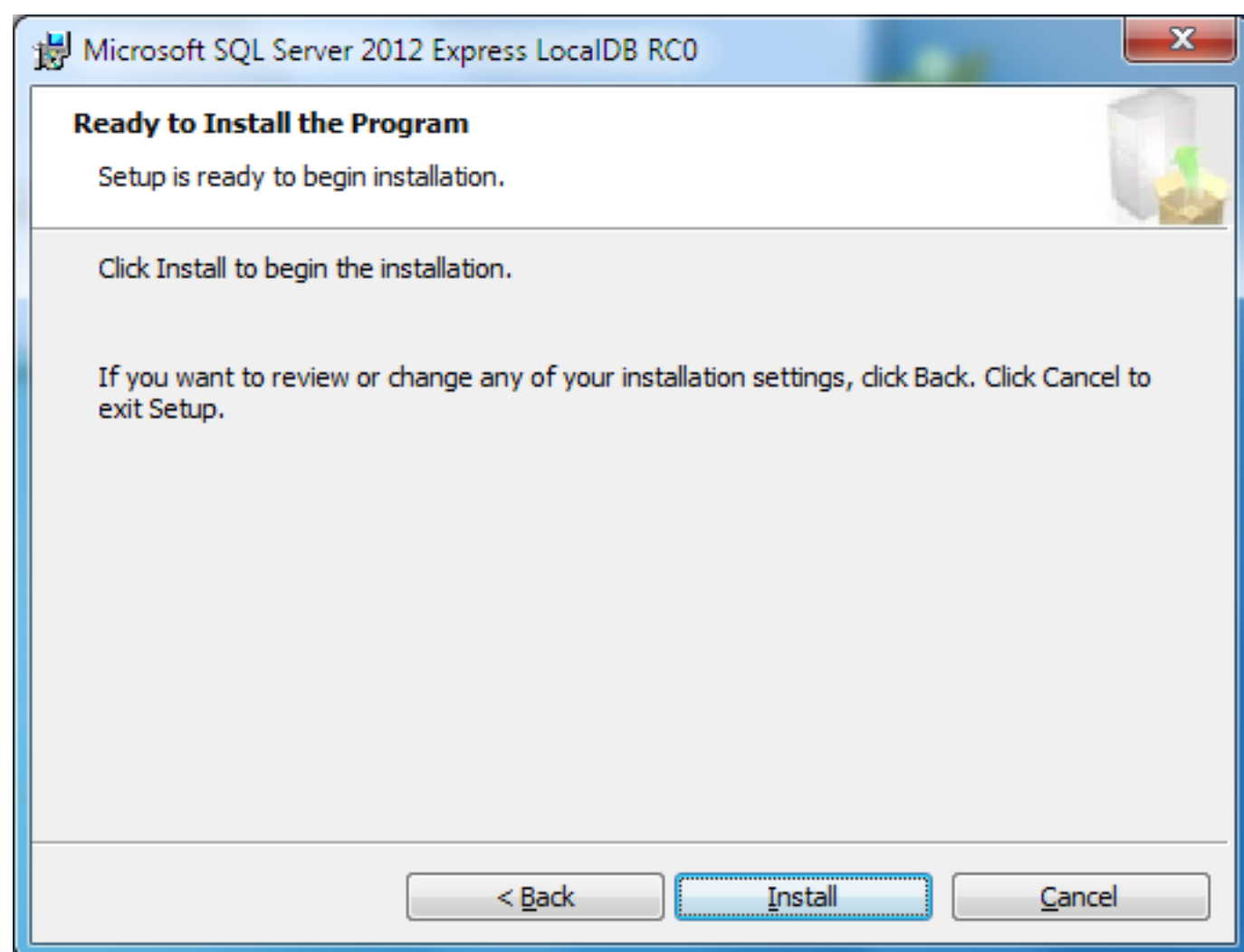
ابتدا LocalDB را دانلود نمایید. سپس برای نصب آن بر روی سیستم فقط کافی است که فایل نصاب برنامه را اجرا نموده و License مربوطه را قبول نمایید. همچنین در صورت نیاز به Silent Installation کافی است که از دستور زیر در خط فرمان استفاده نمایید:

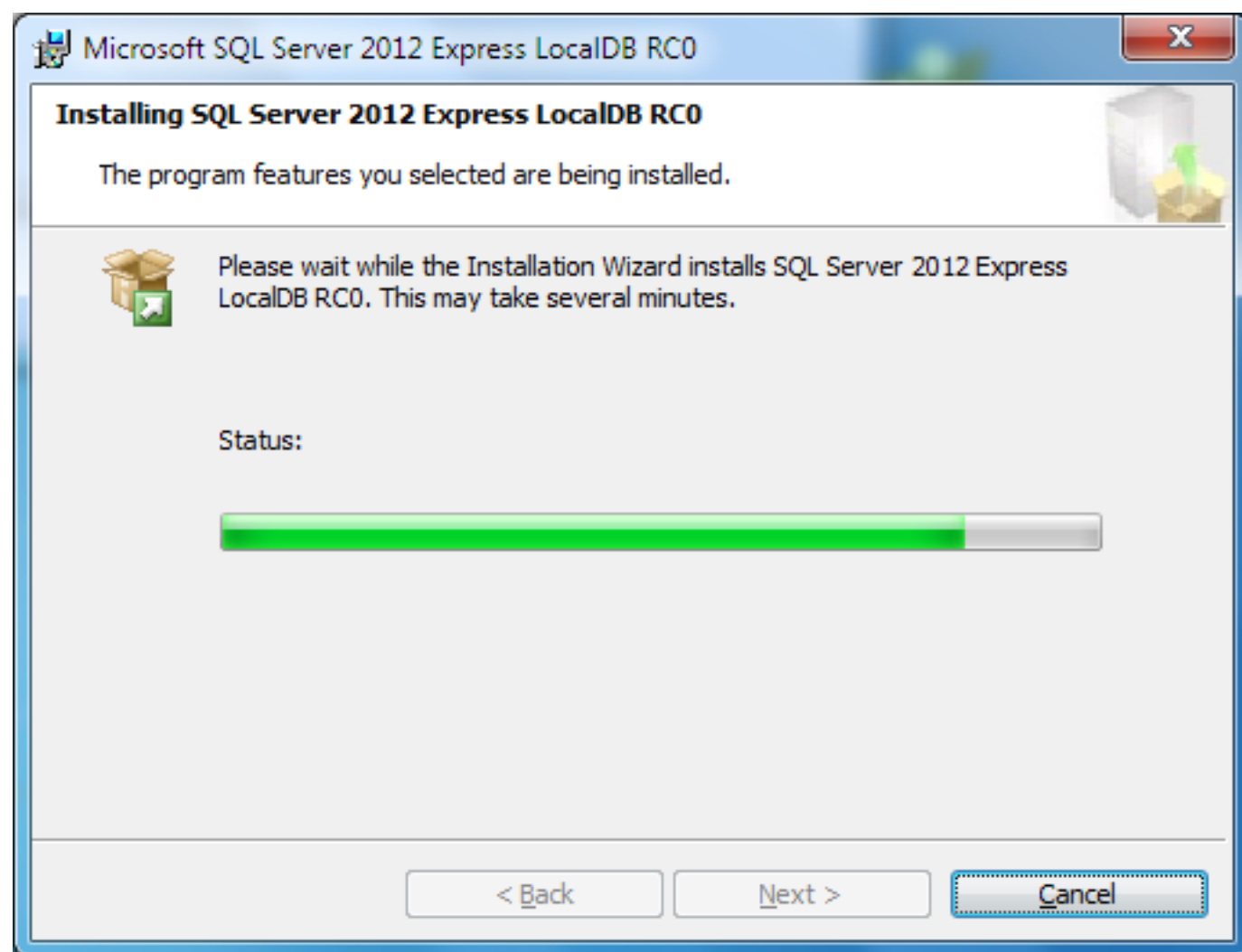
```
msiexec /i SqlLocalDB.msi /qn IACCEPTSQQLOCALDBLICENSETERMS=YES
```

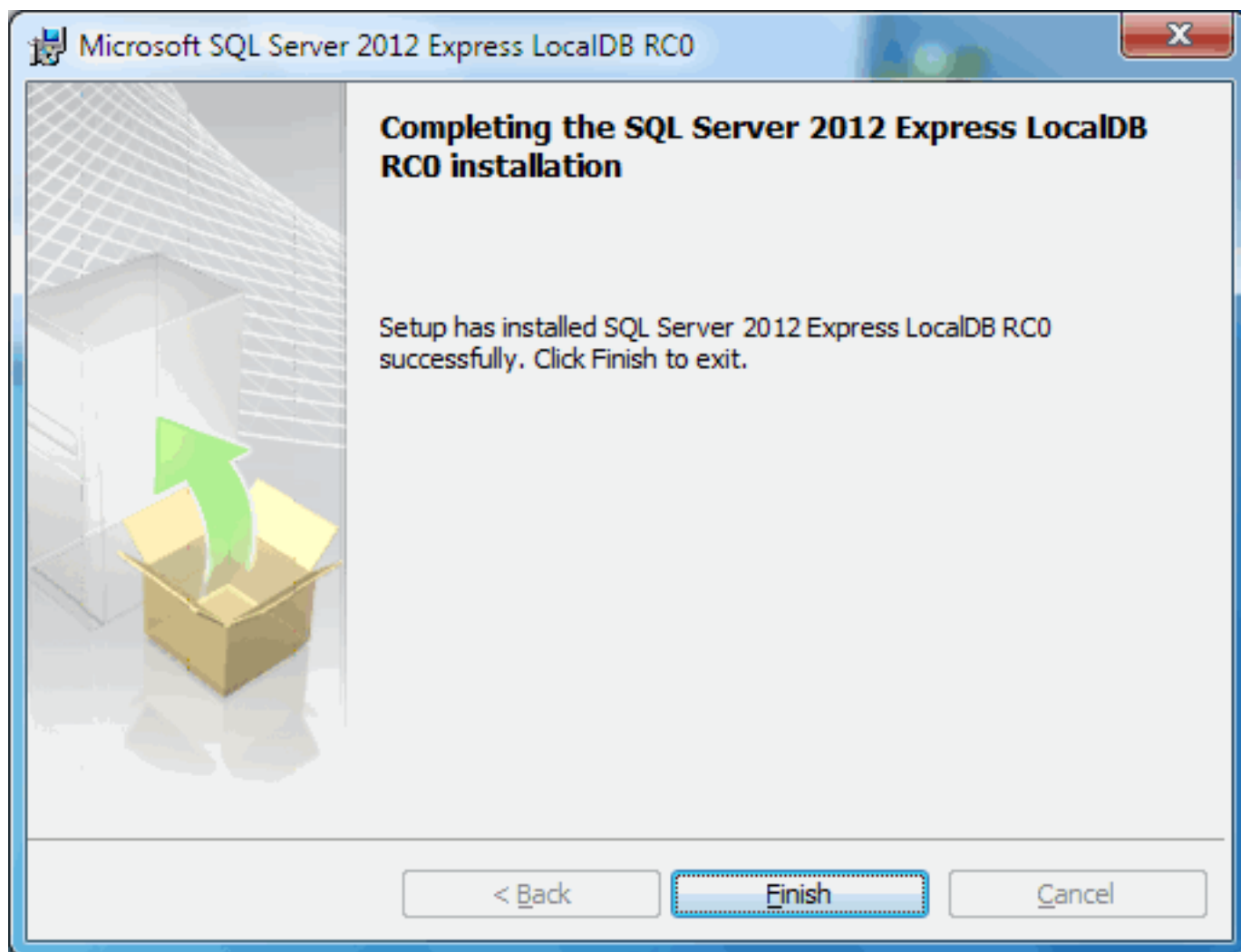
همچنین می‌توانید مراحل نصب را توسط فایل نصاب انجام دهید:





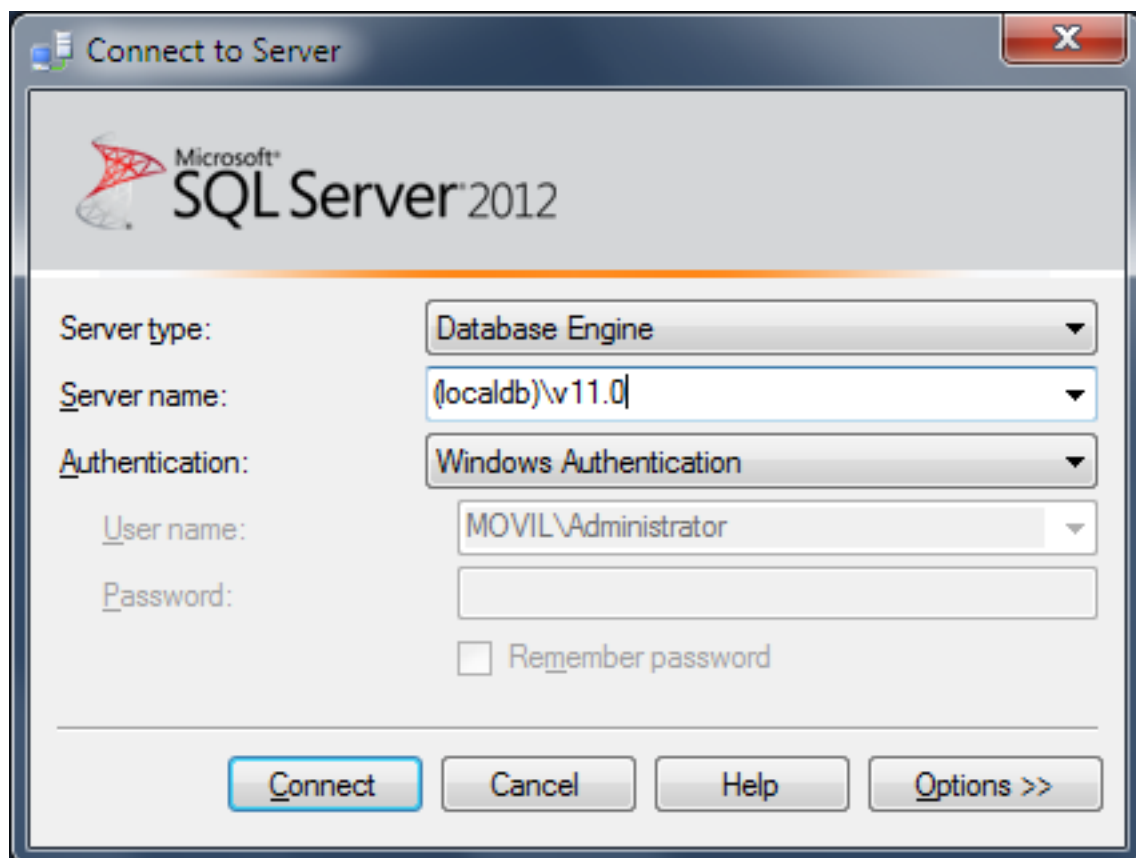






#### نحوه اتصال به LocalDB توسط Sql Server Management Studio

اگر .net framework خود را از نسخه 4 به 4.0.2 و یا نسخه‌های بعد از آن به روز رسانی کرده باشید می‌توان توسط Sql Server 2012 Management Studio به Sql server LocalDB وصل شد. عبارت (local)\v11.0 را به عنوان نام سرور وارد نمایید.



مجددا لازم به ذکر است که امکان اتصال توسط Management Studio های قبلی به بانک LocalDB امکان پذیر نمی باشد.

[برای مطالعه بیشتر](#)



## نظرات خوانندگان

نویسنده: **ramin\_rp**  
تاریخ: ۹:۳۷ ۱۳۹۱/۰۶/۱۶

سلام  
تا جایی که من اطلاع دارم نسخه rtm از 2010 vs ساپورت نمیشه

نویسنده: **وحید نصیری**  
تاریخ: ۱۲:۷ ۱۳۹۱/۰۶/۱۶

هیچ بانک اطلاعاتی خاصی الزاما نیازی به VS.NET ندارد. کانکشن استرینگ را تنظیم می کنید و بعد می تونید [با کد نویسی](#) حتی جدول، فیلد و تمام تنظیمات مورد نظر را اضافه کنید. ضمن اینکه خیلی از توانایی های آن با VS.NET 2010 هم کار می کند ( ^ ) .

نویسنده: **علی یزدی**  
تاریخ: ۲:۲۲ ۱۳۹۱/۰۶/۱۷

واقعا که عالی بود. ممنون

نویسنده: **کیان**  
تاریخ: ۱۵:۵۹ ۱۳۹۱/۰۶/۲۸

ایا LocalDB جایگزینی برای SqlExpress خواهد بود در ادامه؟ ( [اینجا](#) انتهای پاراگراف اول منظورش چیه دقیقا؟ *SQL Server Express LocalDB should be used in place of the SQL Server Express user instance feature which is deprecated* )

هیچ شکلی نمی توان تحت شبکه کار کرد با این نسخه؟ مثلا جایی که قراره 3 تا سیستم بصورت کلاینت/سرور کار کنن نمی شه فایل دیتابیس روی یک سیستم باشه و از رو 2 تای دیگه بهش کانکت شد؟ سنارویی که خیلی وقتا ممکنه استفاده بشه بهترین راه برای سناریوهایی مثل مورد 2 که مثال زدم چیه؟ (یک محیط و یک برنامه معمولی که قرار نیست sql server بصورت full نصب بشه رو سرور) آیا می شه با خیال راحت از **SqlServer Express Edition** استفاده کرد؟

نویسنده: **وحید نصیری**  
تاریخ: ۱۶:۱۸ ۱۳۹۱/۰۶/۲۸

- خیر. نام کامل این محصول « [SQL Server Express LocalDB](#) » است + نسخه «Microsoft® SQL Server® 2012 Express» [از اینجا](#) قابل دریافت است. بنابراین جایگزین یا حذف نشده.  
- هدف اصلی از LocalDb ارائه یک « [embedded database](#) » جدید از طرف مایکروسافت است.  
بنابراین [هدف آن](#) استفاده تحت شبکه نیست. جاهایی استفاده می شود که تک کاربر نهایی دانش آنچنانی در نصب و نگهداری بانک های اطلاعاتی ندارد و برنامه و سیستم بانک اطلاعاتی او یکپارچه به نظر می رسند. از این نمونه بانک های اطلاعاتی embedded باز هم هستند. مانند SQL CE، SQLite، Firebird Embedded و غیره.  
- نسخه Express تحت شبکه قابل استفاده است؛ البته [نیاز به تنظیم](#) دارد.

نویسنده: **حسین**  
تاریخ: ۲۱:۱۰ ۱۳۹۱/۰۸/۰۲

من با Management Studio ی SQL Server 2008 همین (localdb\v11.0) رو وارد کردم و متصل شد. مطمئن هستيد که فقط با SQL 2012 میشه بهش وصل شد؟

نویسنده: رضا بزرگی  
تاریخ: ۲۰:۳۵ ۱۳۹۱/۰۸/۰۶

- همیشه لطفاً بگین منظور شما از تحت شبکه دقیقاً چیست؟  
- برنامه MVC4 که با localdb هست را چگونه میشود روی هاست‌های اشتراکی استفاده کرد؟ آیا نیاز به تنظیم خاص در کانکشن‌استرینگ دارد؟

نویسنده: وحید نصیری  
تاریخ: ۲۰:۴۹ ۱۳۹۱/۰۸/۰۶

- یعنی حالت سرور داشته باشد و بتوان به آن مستقیماً از راه دور متصل شد. مثل SQL Server کامل.  
- هاست اشتراکی باید این بانک اطلاعاتی را پیشتر نصب کرده باشد.  
مانند SQL Server CE نیست که با توزیع چند DLL مشکل آن برطرف شود و نیازی به نصب نداشته باشد.

نویسنده: میثم  
تاریخ: ۱۰:۲۴ ۱۳۹۲/۰۱/۱۱

سلام. دیتابیس برنامه من LocalDB هستش و از روش EF Code First هم برای برقراری ارتباط با دیتابیس استفاده میکنم. میخوام فایل‌های مرتبط با دیتابیس‌م .mdf و .ldf. توی فولدر Bin باشه. ولی وقتی از کانکشن‌استرینگ زیر استفاده میکنم دیتابیس رو توی ...\\C:\\Users در حالی که با SQL Compact چنین مشکلی نداشتم و فایل .sdf رو در فولدر Bin قرار میداد.

```
<entityFramework>
  <defaultConnectionFactory type="System.Data.Entity.Infrastructure.SqlConnectionFactory,
EntityFramework">
    <parameters>
      <parameter value="Data Source=(localdb)\\v11.0; Integrated Security=True;
MultipleActiveResultSets=True;" />
    </parameters>
  </defaultConnectionFactory>
</entityFramework>
```

```
<connectionStrings>
  <clear/>
  <add name="MyContext"
    connectionString="Data Source=(localdb)\\v11.0; Integrated Security=True;
MultipleActiveResultSets=True; AttachDBFilename=MyDatabase.mdf"
    providerName="System.Data.SqlClient"
  />
</connectionStrings>
```

ممنون میشم راهنماییم کنید.

نویسنده: وحید نصیری  
تاریخ: ۱۰:۳۸ ۱۳۹۲/۰۱/۱۱

این یک روش عمومی است و در تمام رشته‌های اتصالی دات نیتی کار می‌کند:  
از DataDirectory استفاده کنید. مثلاً:

```
AttachDBFilename=|DataDirectory|\\database.mdf
```

مقدار آن در برنامه‌های ASP.NET به صورت خودکار به پوشه استاندارد App\_Data مپ می‌شود. برای سایر حالات می‌تونید اون رو در زمان آغاز برنامه دستی مقدار دهی کنید:

```
AppDomain.CurrentDomain.SetData("DataDirectory", "C:\\myDB");
```

ضمناً روش مسيردهی کامل هم همیشه کار می‌کند

```
AttachDbFilename='Full\Path\To.MDF'
```

نویسنده: یزدان  
تاریخ: ۱۳۹۲/۰۱/۱۱ ۱۲:۱۳

سلام؛ AppDomain چیه ؟  
اینکه میگن توی یک برنامه ممکنه چند AppDomain داشته باشیم چه معنی میده ؟

نویسنده: وحید نصیری  
تاریخ: ۱۳۹۲/۰۱/۱۱ ۱۲:۲۷

[AppDomain](#) روشی است برای ایزوله سازی اطلاعات برنامه‌ها از یکدیگر. به این ترتیب در یک پروسه هم می‌شود چندین برنامه دیگر را به صورت کاملاً ایزوله از هم بارگذاری و اجرا کرد.

نویسنده: رحیم  
تاریخ: ۱۳۹۲/۰۱/۱۵ ۰:۳۰

اگر از ویندوز XP رو هم پشتیبانی می‌کرد خیلی خوب می‌شد، بدون این گزینه نمی‌شه رفت سراغش مخصوصاً برای برنامه‌های شرکتی که باید روی اکثر سیستم عامل‌ها اجرا بشه (و فعلاً xp استفاده زیادی داره)

نویسنده: حسین  
تاریخ: ۱۳۹۲/۱۱/۱۸ ۱:۰۰

با تشکر  
من 2 تا مشکلی که در هنگام استفاده از local db در ویژوال استدیو 2012 برخورد کردم این بود که  
1-هر از گاهی پیغام کم آوردن خطای حافظه رو میداد (مثلاً هنگامی که می‌خواستم محتویات جداول رو ببینم) و تقریباً توی اکثر اوقات چاره‌ای به جز بستن ویژوال استدیو و باز کردن مجدد آن نداشتم  
2-اسکرپت تهیه کردن از جدول و اطلاعات داخل آن واقعا عذاب آورده و هر جدول رو باید جداگانه اسکرپت تبدیل میکردم خواستم ببینم راهی برای این مشکل وجود دارد؟

نویسنده: محسن درپرستی  
تاریخ: ۱۳۹۳/۰۱/۰۹ ۲۰:۳۷

امروز هرچقدر سعی می‌کنم یک دیتابیس Sql server ce رو روی هاست اشتراکی منتقل کنم مدام با خطای زیر مواجه می‌شم. راه حل‌های مختلفی رو هم امتحان کردم مثل تغییر سطح تراست یا تغییر نسخه کامپایل به x86 یا x64 [این](#) و [این](#) نهایتاً به جایی نرسیدم.

Unable to load the native components of SQL Server Compact corresponding to the ADO.NET provider of version 8854. Install the correct version of SQL Server Compact. Refer to KB article 974247 for more details.

نویسنده: وحید نصیری  
تاریخ: ۱۳۹۳/۰۱/۰۹ ۲۰:۵۵

روی این هاست قبلا یک نسخه‌ی SQL Ce نصب شده. الان شماره درایور دات نت SQL Ce که در برنامه‌ی شما در حال استفاده است، تطابقی با نگارش نصب شده‌ی روی سرور ندارد. بسته کامل آن‌را از نیوگت دریافت کنید. [از اینجا](#) علاوه بر ارائه درایور دات نت، حاوی دو پوشه amd64 و x86 کپی شده در پوشه bin برنامه هم خواهد بود (فایل‌های native اصل بانک اطلاعاتی). همین پوشه‌ها و فایل‌ها را باید به همراه برنامه نیز ارائه کنید.

نویسنده: علی پناهی  
تاریخ: ۱۳۹۳/۰۴/۱۱ ۸:۵۰

در کد زیر اگر کانکشن استرینگ را کامنت کنم بانک ساخته می‌شود و مشکلی ندارد در غیر این صورت پیغام خطای زیر را می‌دهد:

A file activation error occurred. The physical file name 'tax.mdf' may be incorrect. Diagnose and correct additional errors, and retry the operation.

CREATE DATABASE failed. Some file names listed could not be created. Check related errors.

فایل app.config :

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <configSections>

    <section name="entityFramework"
type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework, Version=6.0.0.0,
Culture=neutral, PublicKeyToken=b77a5c561934e089" requirePermission="false" />
    <!-- For more information on Entity Framework configuration, visit
http://go.microsoft.com/fwlink/?LinkID=237468 --></configSections>

    <connectionStrings>
      <clear/>
      <add name="Context"
        connectionString="Data Source=(localdb)\v11.0; Integrated Security=True;
MultipleActiveResultSets=True; AttachDBFilename=tax.mdf"
        providerName="System.Data.SqlClient"
      />
    </connectionStrings>

    <entityFramework>

      <defaultConnectionFactory type="System.Data.Entity.Infrastructure.LocalDbConnectionFactory,
EntityFramework">
        <parameters>
          <parameter value="v11.0"/>
        </parameters>
      </defaultConnectionFactory>
    </entityFramework>

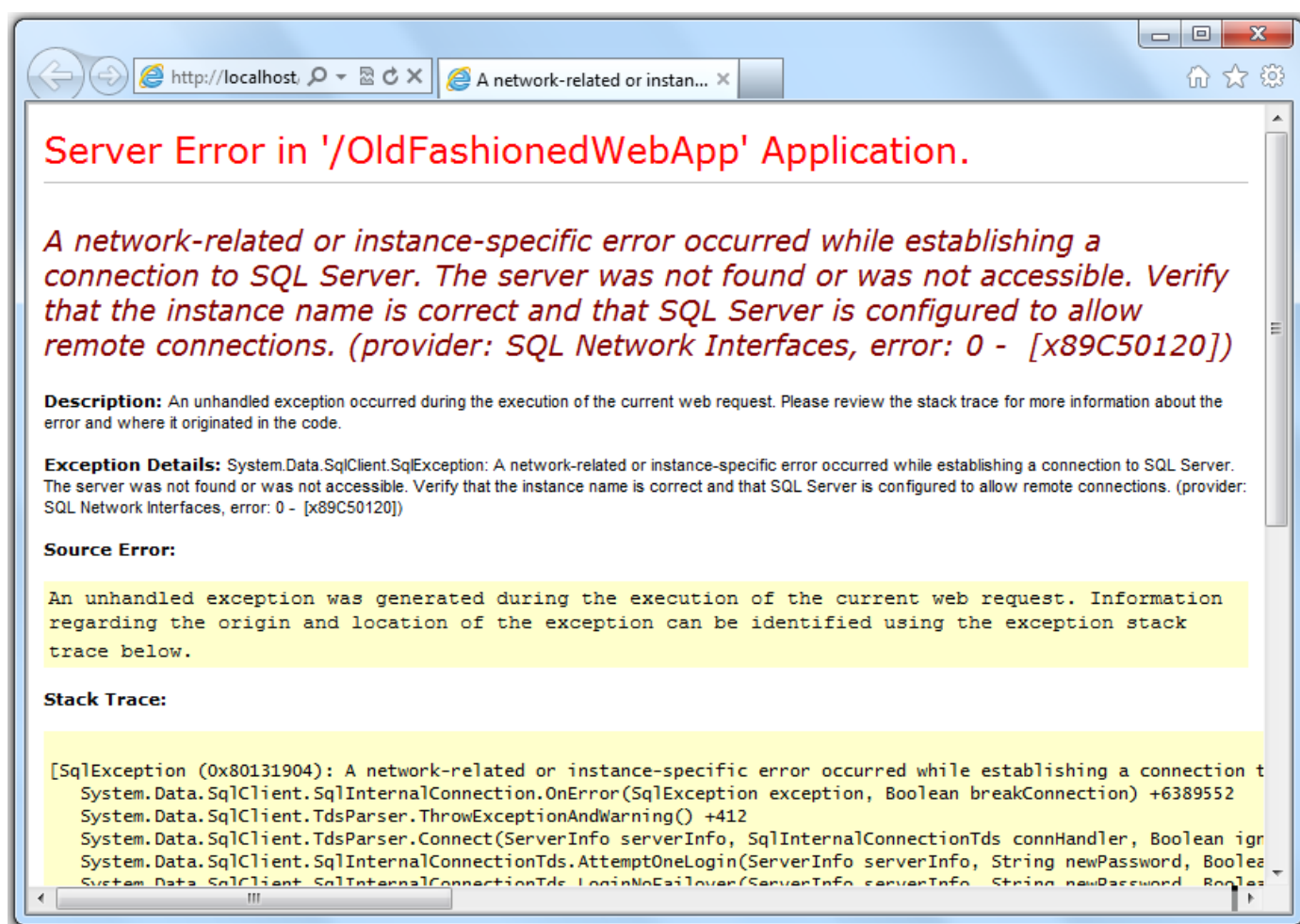
  </configuration>
</startup><supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" /></startup></configuration>
```

نویسنده: وحید نصیری  
تاریخ: ۱۳۹۳/۰۴/۱۱ ۹:۵۰

کمی بالاتر در مورد تنظیمات [data directory](#) بحث شده.

این مقاله قسمت اول یک سری دو قسمتی است، که در آن به نحوه استفاده از LocalDB در IIS می‌پردازیم.

LocalDb دیتابیس توصیه شده برای ویژوال استودیو است و برای انواع پروژه‌ها مانند اپلیکیشن‌های وب می‌تواند استفاده شود. هنگام استفاده از این دیتابیس در IIS Express یا Cassini همه چیز طبق انتظار کار میکند. اما به محض آنکه بخواهید از آن در Full IIS استفاده کنید با خطاهایی مواجه می‌شوید. مقصود از Full IIS همان نسخه ای است که به همراه ویندوز عرضه می‌شود و در قالب یک Windows Service اجرا می‌گردد.



هنگام استفاده از Full IIS دو خاصیت از LocalDb باعث بروز مشکل می‌شوند:

LocalDb نیاز دارد پروفایل کاربر بارگذاری شده باشد

بصورت پیش فرض، وهله LocalDb متعلق به یک کاربر بوده و خصوصی است

در ادامه این مقاله روی بارگذاری پروفایل کاربر تمرکز می‌کنیم، و در قسمت بعدی به مالکیت وهله LocalDb می‌پردازیم.

بارگذاری پروفایل کاربر

بگذارید دوباره به خطای موجود نگاهی بیاندازیم:

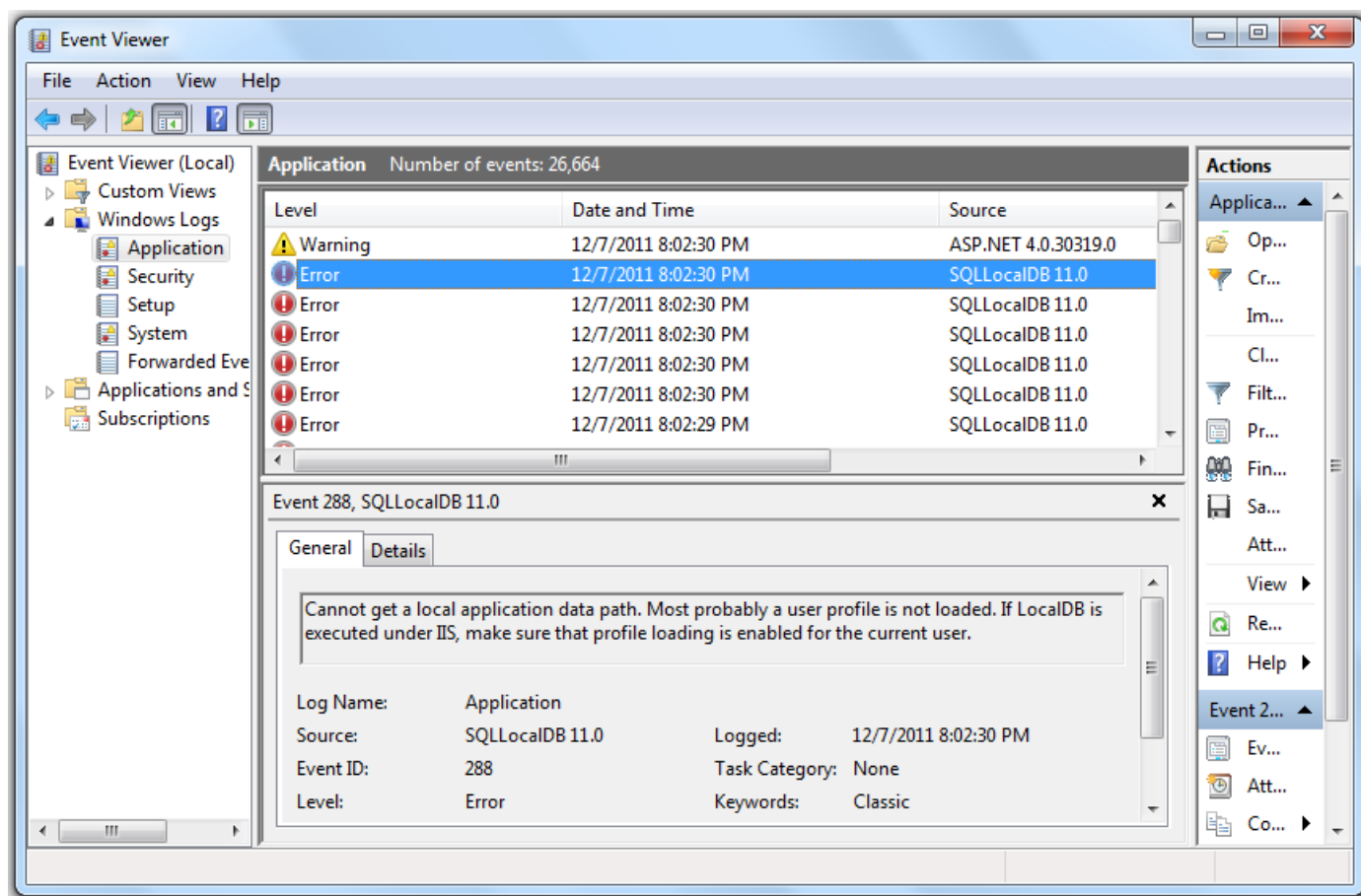
*System.Data.SqlClient.SqlException: A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not accessible. Verify that the instance name is correct and that SQL Server is configured to allow remote connections. (provider: SQL Network Interfaces, error: (0 - [x89C50120*

این پیغام خطا زیاد مفید نیست، اما LocalDb اطلاعات بیشتری در Event Log ویندوز ذخیره می‌کند. اگر **Windows Logs** را باز کنید و به قسمت **Application** بروید پیغام زیر را مشاهده خواهید کرد.

*Windows API call SHGetKnownFolderPath returned error code: 5. Windows system error message is: Access is denied  
Reported at line: 400*  
بعلاوه این پیام خطا:

*Cannot get a local application data path. Most probably a user profile is not loaded. If LocalDB is executed under IIS, make sure that profile loading is enabled for the current user*

به احتمال زیاد تعداد بیشتری از این دو خطا در تاریخچه وقایع وجود خواهد داشت، چرا که منطق کانکشن ADO.NET چند بار سعی می‌کند در بازه‌های مختلف به دیتابیس وصل شود.

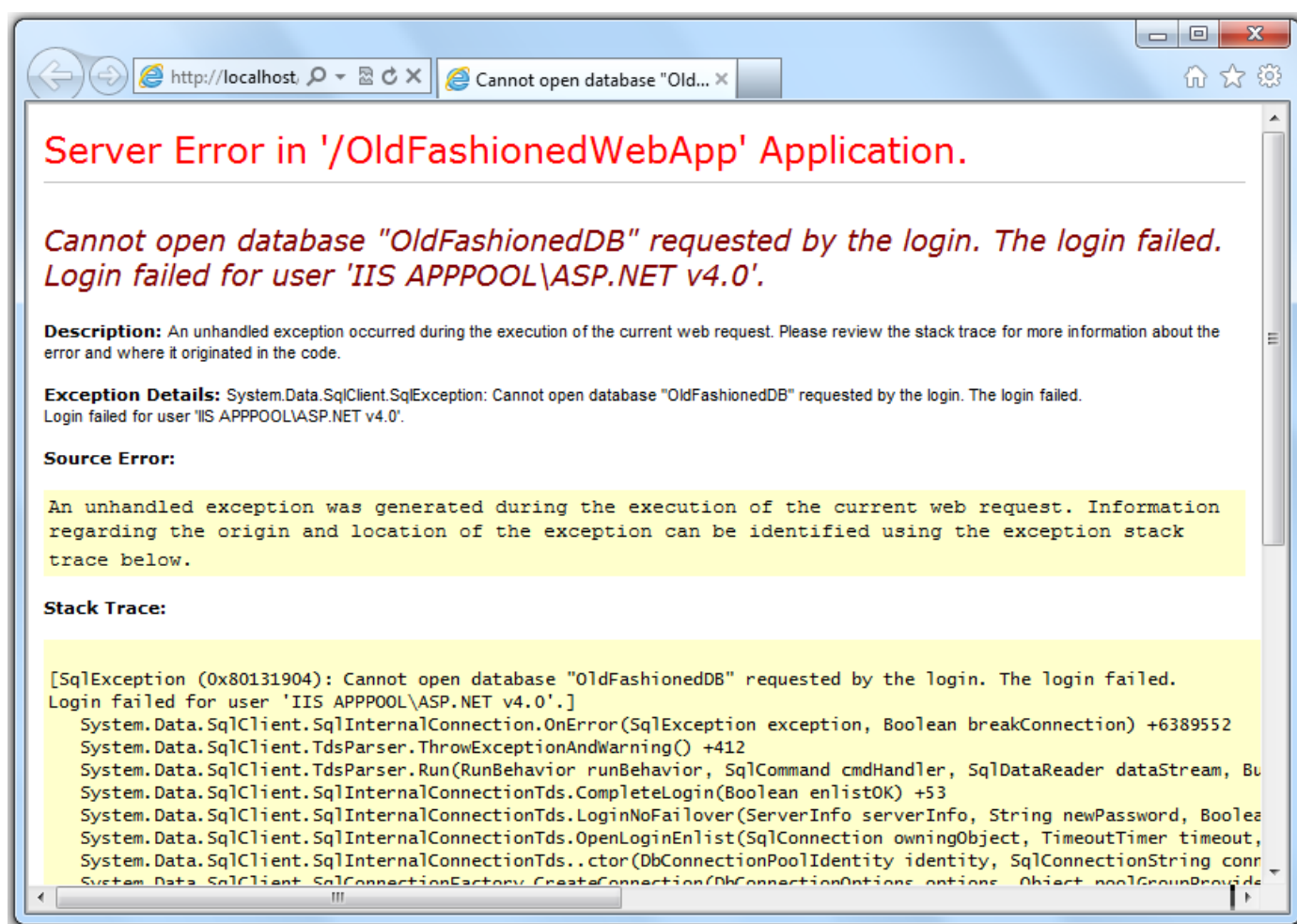


پیغام خطای دوم واضح است، نیاز است پروفایل کاربر را بارگذاری کنیم. انجام اینکار زیاد مشکل نیست، هر Application Pool در IIS تنظیماتی برای بارگذاری پروفایل کاربر دارد که از قسمت Advanced Settings قابل دسترسی است. متأسفانه پس از انتشار سرویس پک 1 برای Windows 7 مسائل کمی پیچیده‌تر شد. در حال حاضر فعال کردن **loadUserProfile** برای بارگذاری

کامل پروفایل کاربر به تنهایی کافی نیست، و باید **setProfileEnvironment** را هم فعال کنیم. برای اطلاعات بیشتر در این باره به مستندات [KB 2547655](https://msdn.microsoft.com/en-us/library/aa479616.aspx) مراجعه کنید. بدین منظور باید فایل **applicationHost.config** را ویرایش کنید. فایل مذکور در مسیر `C:\Windows\System32\inetsrv\config` قرار دارد. همانطور که در مستندات KB 2547655 توضیح داده شده، باید پرچم هر دو تنظیمات را برای **ASP.NET 4.0** فعال کنیم:

```
<add name="ASP.NET v4.0" autoStart="true" managedRuntimeVersion="v4.0"
managedPipelineMode="Integrated">
  <processModel identityType="ApplicationPoolIdentity" loadUserProfile="true"
setProfileEnvironment="true" />
</add>
```

پس از بروز رسانی این تنظیمات، Application Pool را Restart کنید و اپلیکیشن را مجدداً اجرا نمایید. اگر همه چیز طبق انتظار پیش برود، با خطای جدیدی مواجه خواهیم شد:



جای هیچ نگرانی نیست، چرا که این پیغام خطا انتظار می‌رود. همانطور که در ابتدا گفته شد، دو خاصیت LocalDb باعث بروز این خطاها می‌شوند و ما هنوز به خاصیت دوم نپرداخته ایم. بصورت پیش فرض وهله‌های LocalDb خصوصی (private) هستند و در Windows account جاری اجرا می‌شوند. بنابراین ApplicationPoolIdentity در IIS به وهله‌های دیتابیس دسترسی نخواهد داشت. در قسمت دوم این مقاله، راه‌های مختلفی را برای رفع این مشکل بررسی می‌کنیم.



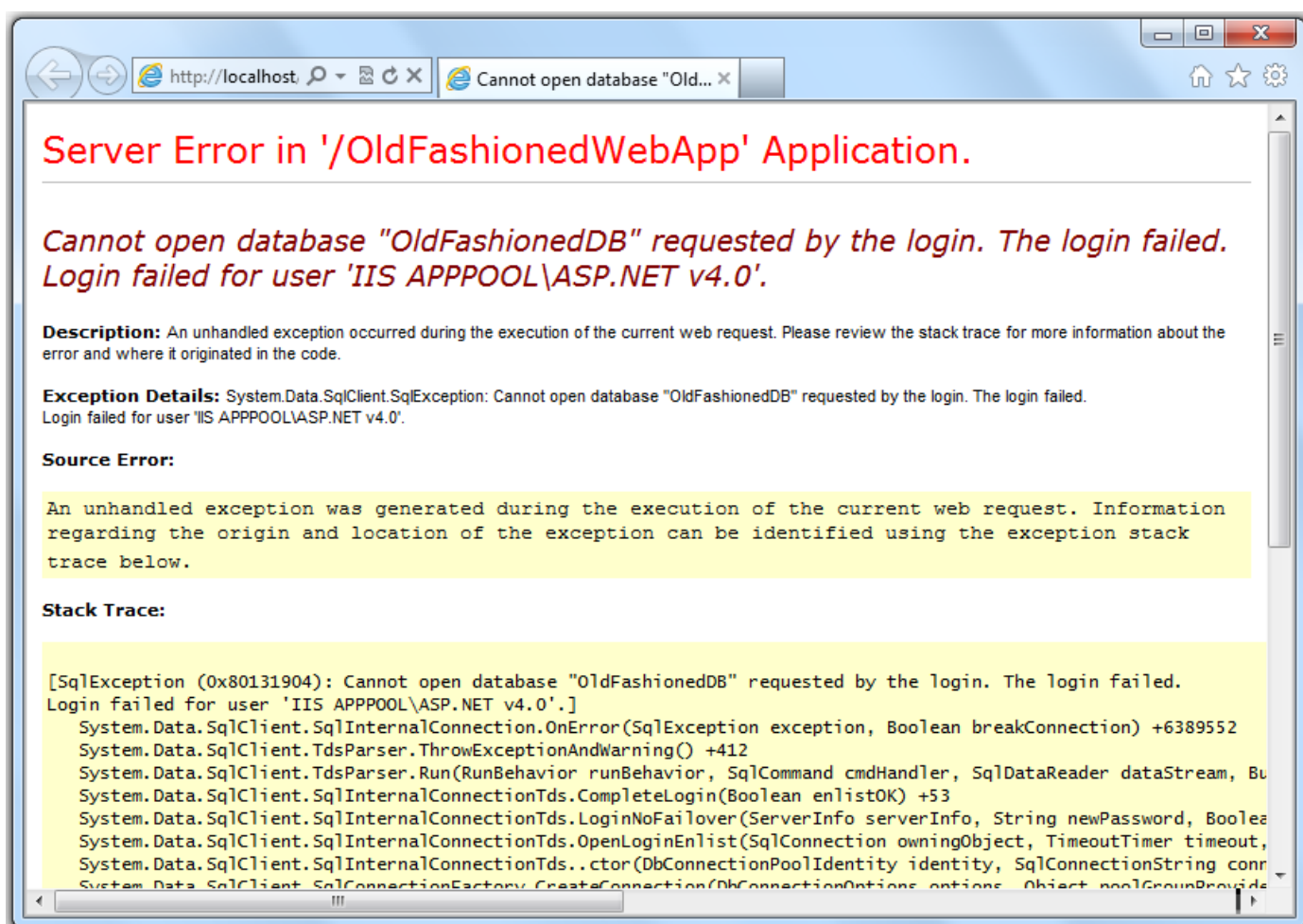
در [قسمت قبلی](#) این مقاله گفتیم که دو خاصیت از LocalDb هنگام استفاده از Full IIS باعث بروز خطا می شوند:

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:



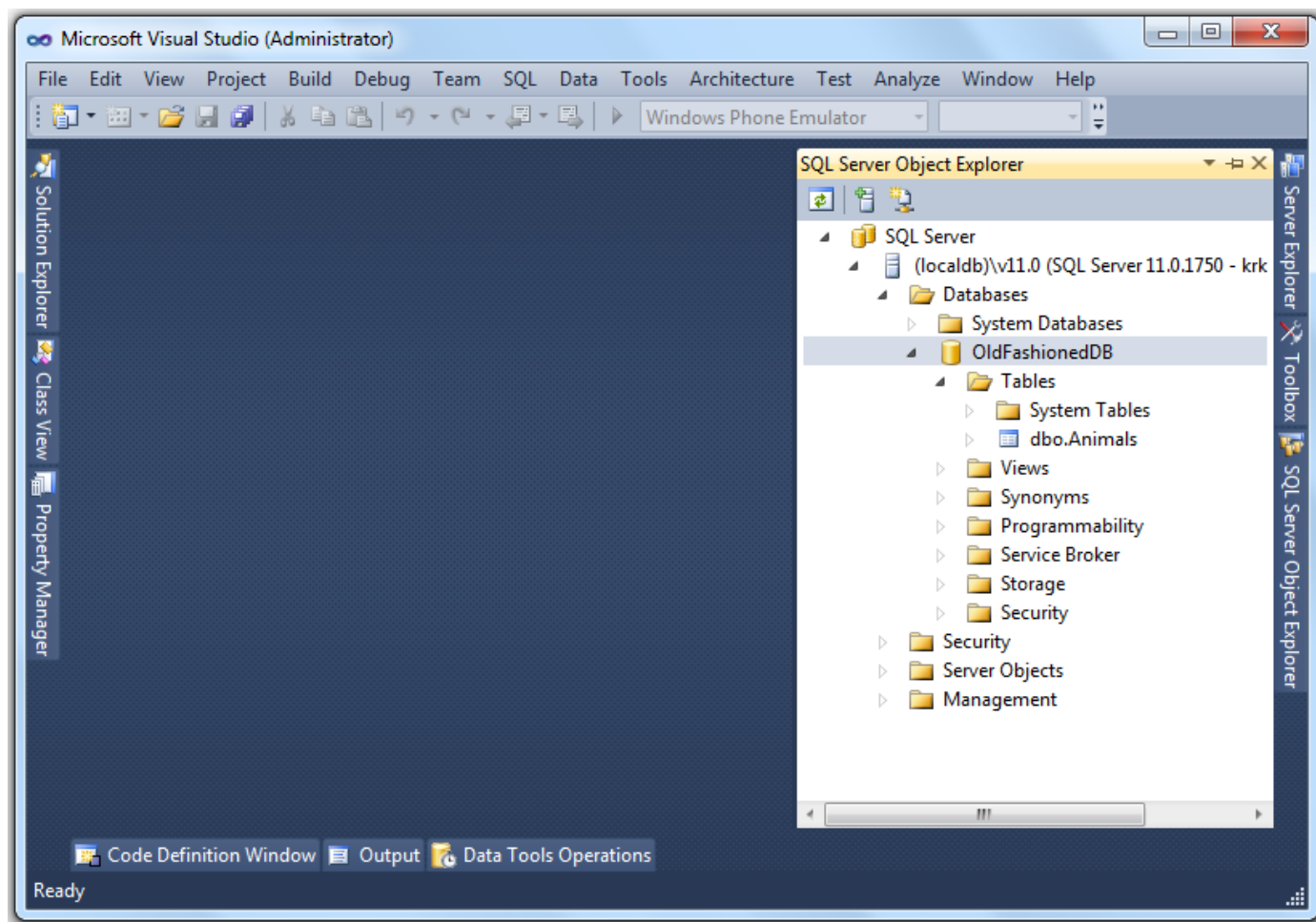
همانطور که مشاهده می کنید با خطای زیر مواجه هستیم:

*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 وهله 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 وهله از LocalDb وجود دارد. برخلاف وهله‌های SQL Server Express که بعنوان سرویس‌های ویندوزی اجرا می‌شوند، وهله‌های 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، وهله 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.

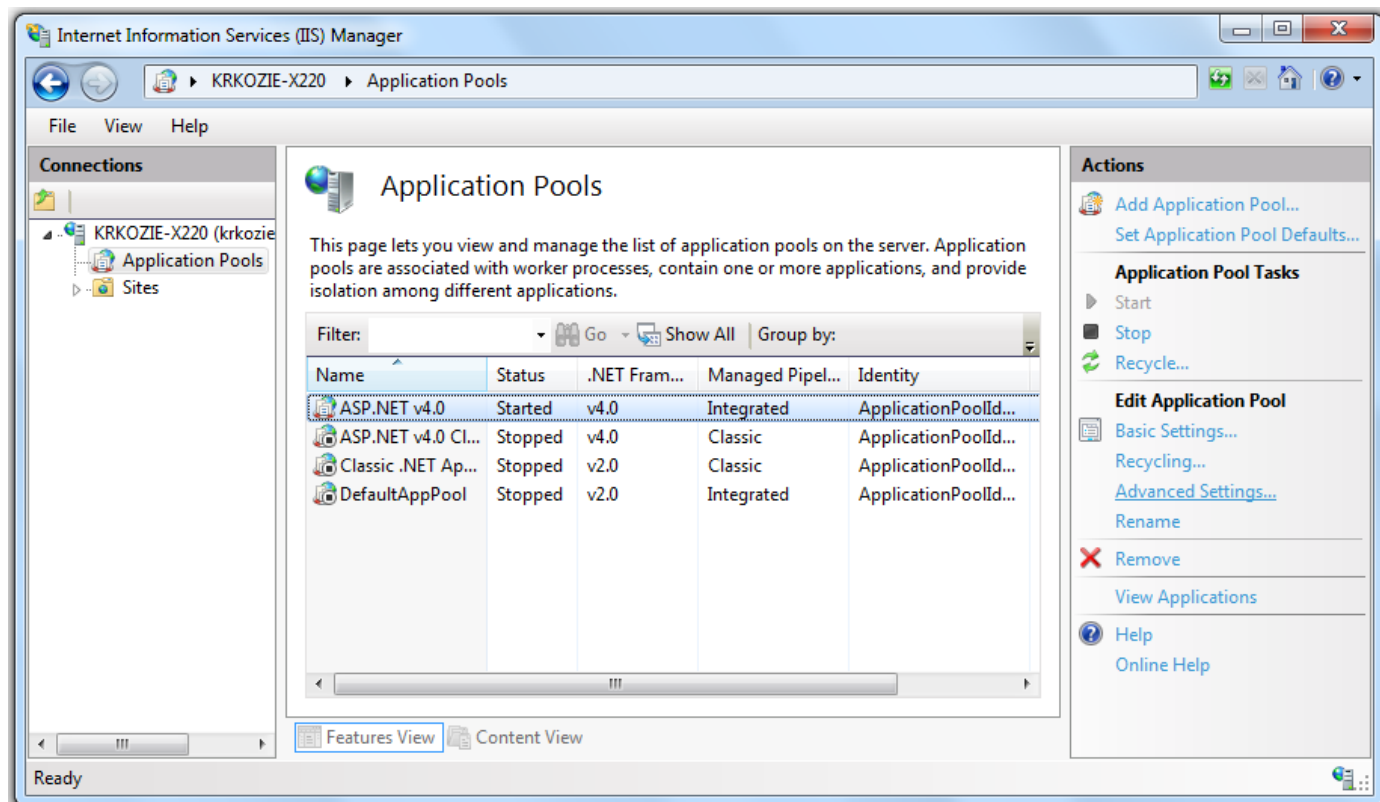
به همین دلیل است که اپلیکیشن وب ما می‌تواند بدون هیچ مشکلی روی IIS Express اجرا شود و دیتابیس را باز کند. چرا که IIS Express درست مانند LocalDb یک پروسس کاربری است. IIS Express توسط ویژوال استودیو راه اندازی می‌شود و روی حساب کاربری جاری اجرا می‌گردد، پس پروسس آن با پروسس خود ویژوال استودیو یکسان خواهد 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	<b>v4.0</b>
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

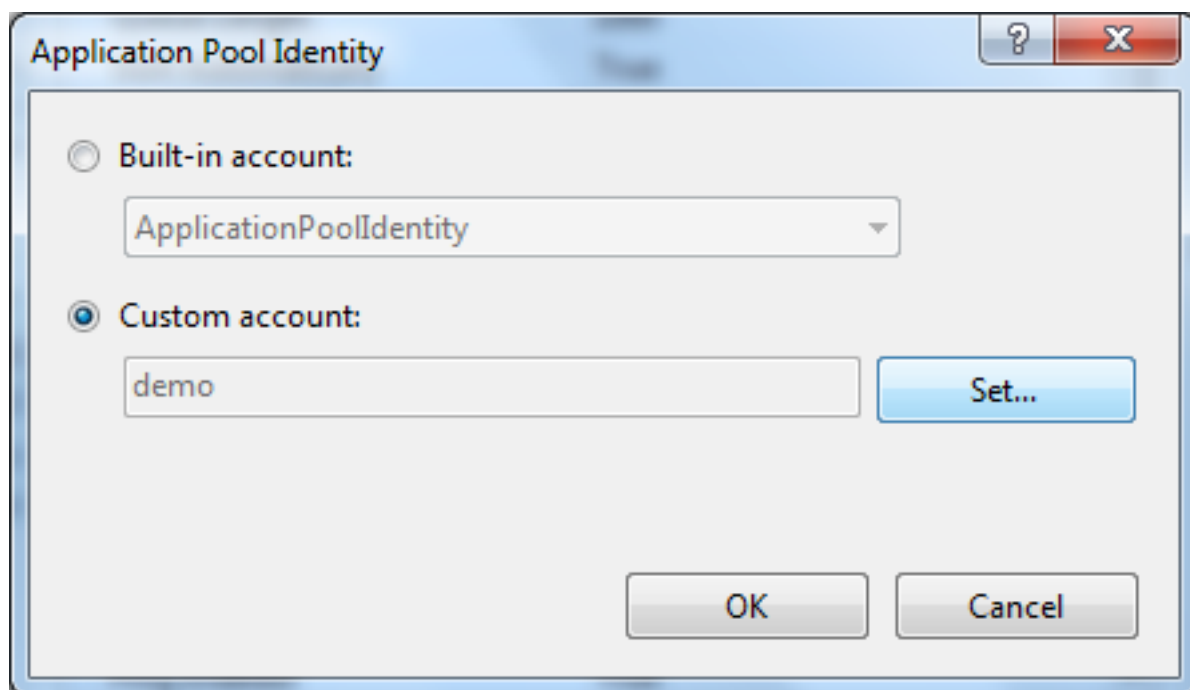
**Process Model**

<b>Identity</b>	<b>ApplicationPoolIdentity</b> <span>...</span>
Idle Time-out (minutes)	20
Load User Profile	True
Maximum Worker Processes	1
Ping Enabled	True

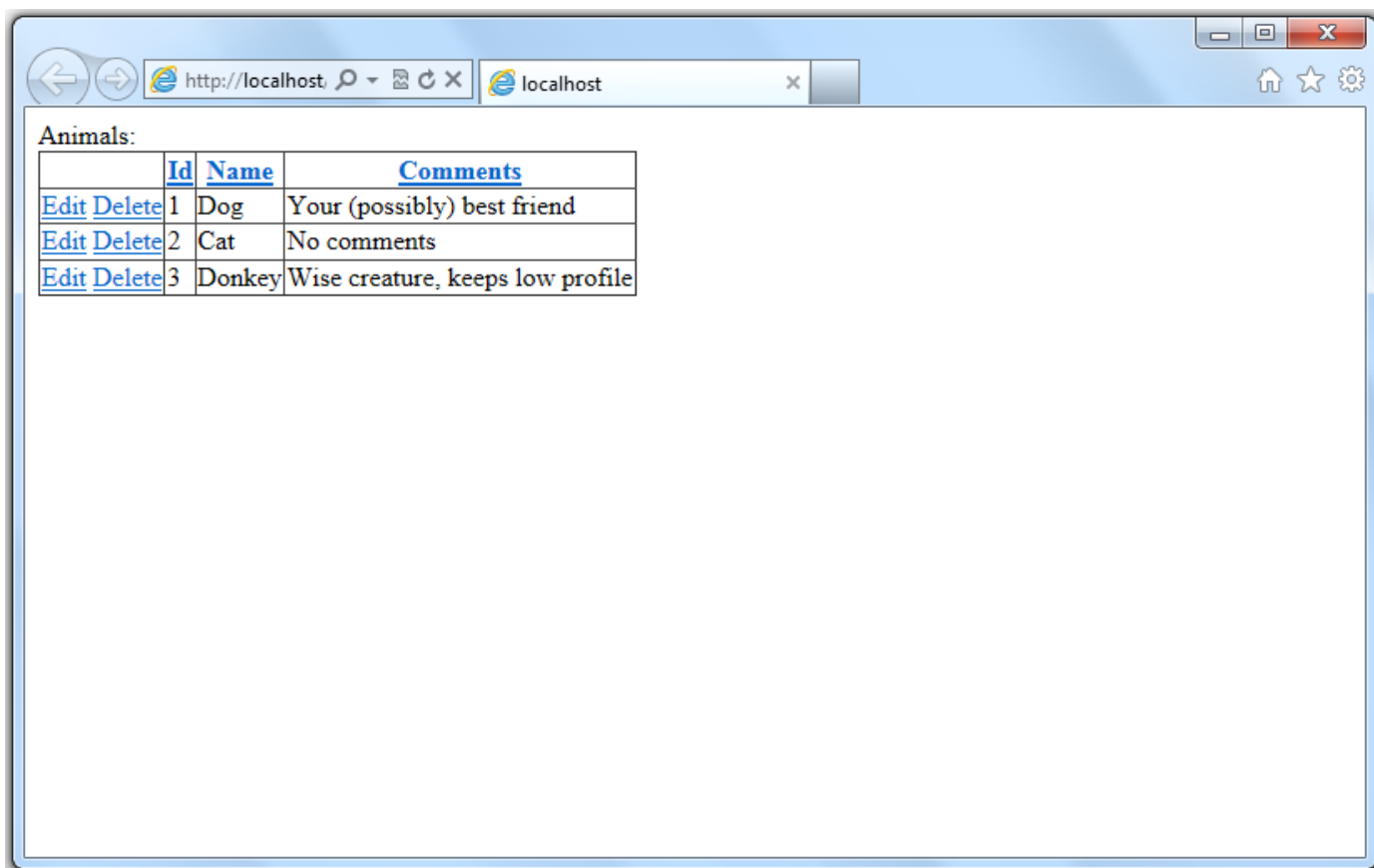
**Identity**  
 [identityType, username, password] Configures the application pool to run as built-in account, i.e. Application Pool Identity (recommended), Network Service, Local System, Local Service, or as a specific user identity.

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
```

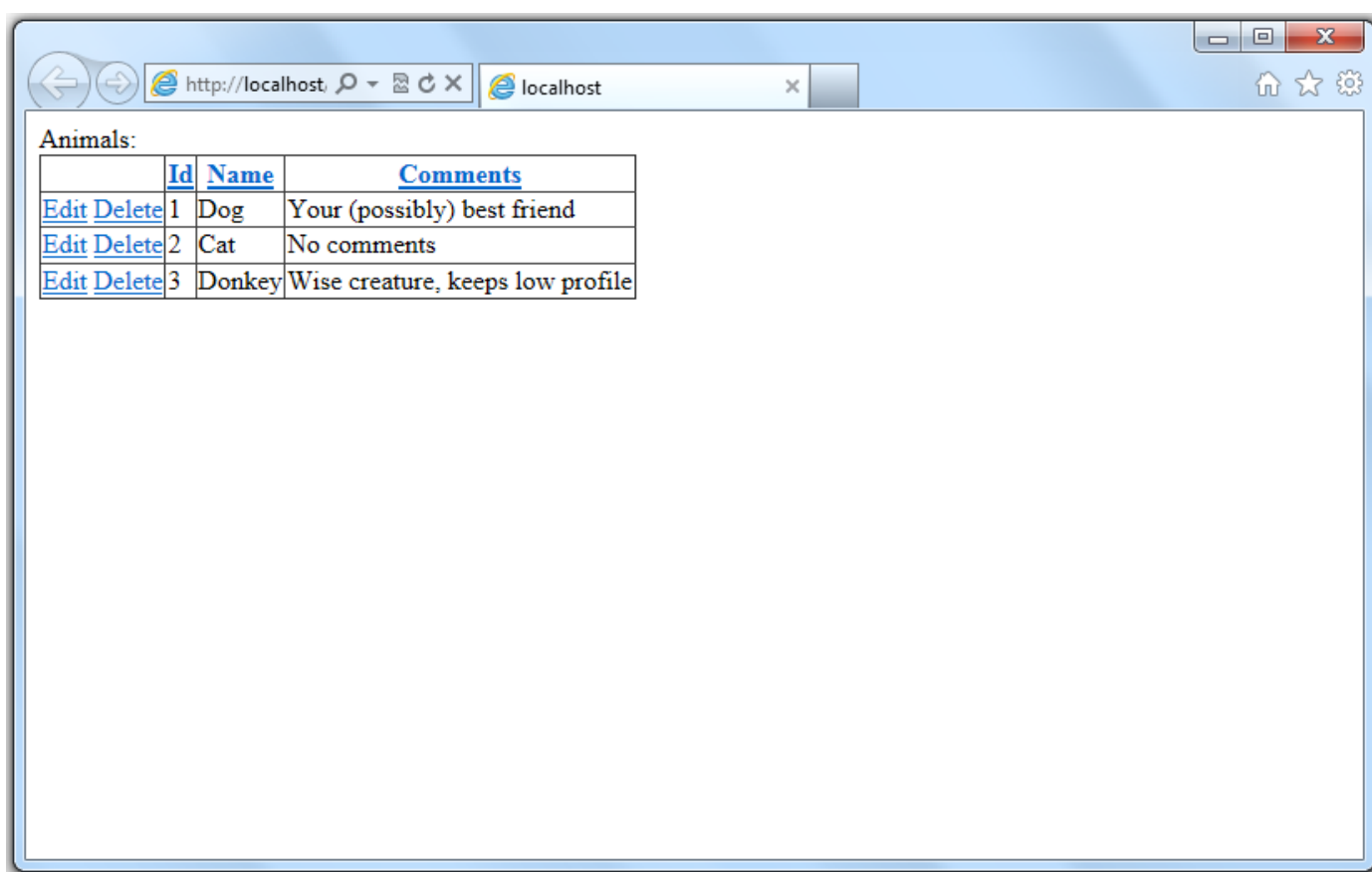
این فرمان و هله خصوصی LocalDb را با نام عمومی **IIS\_DB** به اشتراک می‌گذارد. حال تمام کاربران سیستم می‌توانند با آدرس **(localdb)\.IIS\_DB** به این و هله وصل شوند. این فرمت آدرس دهی سرور دیتابیس، مشخص می‌کند که از یک و هله shared استفاده می‌کنیم. رشته اتصال جدید مانند لیست زیر خواهد بود:

```
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.