عنوان: **LocalDB چیست؟**
نویسنده: حسین مرادی نیا
تاریخ: ۱۳۹۱/۰۶/۱۶ ۱:۲۴
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, LocalDB, SQL Server 2012

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 Permissionهای 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 Procedureها ، داده‌های جغرافیایی و مکانی (geometry and geography ها) ، Triggers و Viewها 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 برروی 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 Window server 2003 ، Windows xp و Windows 2000

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 studioهای قدیمی مدیریت کرد.

عدم پشتیبانی 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 Debuging دستورات Sql در هنگام اتصال به LocalDB

نحوه نصب

ابتدا Sql server 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 IACCEPTSQLLOCALDBLICENSETERMS=YES
```

همچنین می‌توانید مراحل نصب را توسط فایل نصاب انجام دهید:

Microsoft SQL Server 2012 Express LocalDB RC0
Welcome to the Installation Wizard for SQL Server 2012 Express LocalDB RC0
Setup helps you install, modify or remove SQL Server 2012 Express LocalDB RC0. To continue, click Next.
WARNING: This program is protected by copyright law and international treaties.
< Back
Next >
Cancel

Microsoft SQL Server 2012 Express LocalDB RC0
License Agreement
Please read the following license agreement carefully.
MICROSOFT PRE-RELEASE SOFTWARE LICENSE TERMS
MICROSOFT SQL SERVER 2012 EXPRESS LOCALDB RELEASE CANDIDATE 0 (RC0)
These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the pre-release software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft
I accept the terms in the license agreement
I do not accept the terms in the license agreement
< Back
Next >
Cancel

Microsoft SQL Server 2012 Express LocalDB RC0
Ready to Install the Program
Setup is ready to begin installation.
Click Install to begin the installation.
If you want to review or change any of your installation settings, click Back. Click Cancel to exit Setup.
< Back
Install
Cancel

Microsoft SQL Server 2012 Express LocalDB RC0
Installing SQL Server 2012 Express LocalDB RC0
The program features you selected are being installed.
Please wait while the Installation Wizard installs SQL Server 2012 Express LocalDB RC0. This may take several minutes.
Status:
< Back
Next >
Cancel

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 Management Studio 2012 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 Studioهای قبلی به بانک LocalDB امکان پذیر نمی‌باشد.

برای مطالعه بیشتر

نظرات خوانندگان

نویسنده: ramin_rp
تاریخ: ۱۳۹۱/۰۶/۱۶ ۹:۳۷

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

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

هیچ بانک اطلاعاتی خاصی الزاما نیازی به VS.NET ندارد. کانکشن استرینگ را تنظیم می‌کنید و بعد می‌تونید با کد نویسی حتی جدول، فیلد و تمام تنظیمات مورد نظر را اضافه کنید.
ضمن اینکه خیلی از توانایی‌های آن با VS.NET 2010 هم کار می‌کند (^).

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

واقعاً که عالی بود. ممنون

نویسنده: کیان
تاریخ: ۱۳۹۱/۰۶/۲۸ ۱۵:۵۹

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 نمی‌توان تحت شبکه کار کرد با این نسخه؟ مثلا جایی که قراره 3 تا سیستم بصورت کلاینت/سرور کار کنن نمی‌شه فایل دیتابیس روی یک سیستم باشه و از رو 2تای دیگه بهش کانکت شد؟ سناریویی 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 می‌شه با خیال راحت از **SqlServer Express Edition** استفاده کرد؟

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

- 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 می‌شود که تک کاربر نهایی دانش آنچنانی در نصب و نگهداری بانک‌های اطلاعاتی ندارد و برنامه و سیستم بانک اطلاعاتی او یکپارچه به نظر می‌رسند. از این نمونه بانک‌های 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: ۱۳۹۱/۰۸/۰۲ ۲۱:۱۰

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

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

- میشه لطفا بگین منظور شما از تحت شبکه دقیقا چیه؟
- 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: ۱۳۹۱/۰۸/۰۶ ۲۰:۴۹

- یعنی حالت سرور داشته باشد و بتوان به آن مستقیما 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: ۱۳۹۲/۰۱/۱۱ ۱۰:۲۴

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 کانکشن استرینگ زیر استفاده میکنم دیتابیس رو توی C:\Users\... قرار میده. 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: ۱۳۹۲/۰۱/۱۱ ۱۰:۳۸

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

```
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");
```

ضمنا روش مسیردهی 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 کاملا 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 مخصوصا برای برنامه‌های شرکتی که باید روی اکثر سیستم عامل‌ها اجرا بشه (و فعلا xp استفاده زیادی داره)

نویسنده: حسین
تاریخ: ۱۳۹۲/۱۱/۱۸ ۱:۰

با تشکر
من 2 تا مشکلی که در هنگام استفاده از local db در ویژوال استدیو 2012 برخورد کردم این بود که
-1هر از گاهی پیغام کم آوردن خطای حافظه رو میداد (مثلا هنگامی که میخواستم محتویات جداول رو ببینم) و تقریبا توی اکثر اوقات چاره ای به جز بستن ویژوال استدیو و باز کردن مجدد آن نداشتم
-2اسکریپت تهیه کردن از جدول و اطلاعات داخل آن واقعا عذاب آوره و هر جدول رو باید جداگانه اسکریپت 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 یا این و این
نهایتا 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 نتی، 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>

<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.

عنوان: **نحوه ایجاد Sequence و استفاده آن در Sql Server 2012**
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۷/۲۲ ۱۵:۸
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server 2012, SQL Server

قبل از ایجاد Sequence در Sql Server 2012، توضیح مختصری را درباره آن می‌دهم.
در واقع Sequence روشی برای تولید اعداد ترتیبی با قابلیت افزایش یا کاهش عددهای دلخواه می‌باشد که توسط کاربر یا برنامه نویس ایجاد می‌شود. بنابراین Sequenceها User-Defined می‌باشند.
در اینجا ممکن است سئوالی پیش بیاید که اینکار توسط Identity هم قابل انجام است، اما چرا استفاده از Sequence توسط مایکروسافت پیشنهاد می‌شود.
بدلایل زیر استفاده از Sequence بهتر می‌باشد:

ممکن است Application شما قبل از درج رکورد، درون یک جدول نیاز به عدد منحصربفردی داشته باشد.
عدد تولید شده بوسیله Sequence را می‌توانید بین جداول یا ستونهای مختلف یک جدول به اشتراک بگذارید.
می‌توانید روند تولید اعداد ترتیبی را Restart نمایید. به عبارت دیگر قابلیت Restart نمودن Sequence وجود دارد.
می‌توانید Sequence خود را براساس Sort یک یا چند فیلد، تنظیم نمایید.

Syntax آن به شرح ذیل می‌باشد:

```
CREATE SEQUENCE [schema_name . ] sequence_name
    [ AS [ built_in_integer_type | user-defined_integer_type ] ]
    [ START WITH <constant> ]
    [ INCREMENT BY <constant> ]
    [ { MINVALUE [ <constant> ] } | { NO MINVALUE } ]
    [ { MAXVALUE [ <constant> ] } | { NO MAXVALUE } ]
    [ CYCLE | { NO CYCLE } ]
    [ { CACHE [ <constant> ] } | { NO CACHE } ]
    [ ; ]
```

شرح Syntax :
در زمان ایجاد Sequence، نوع آن می‌بایست عددی باشد، چنانچه Type، آن را مشخص ننمایید، SQL Server، نوع آن را bigint در نظر می‌گیرد.
Start With: بدین مفهوم می‌باشد، که Sequence ایجاد شده از چه عددی آغاز شود.
INCREMENT BY: مفهومش این است که Sequence به چه مقداری افزایش یا کاهش یابد. به عبارت دیگری عدد تولید شده براساس مقدار Increment by تولید می‌شود.
Minvalue: کمترین مقداری که Sequence می‌تواند ایجاد نماید.
Maxvalue: بیشترین مقداری که Sequence می‌تواند ایجاد نماید.
Cycle: مقداری را که برای Cycle تعیین می‌نماییم، بدین مفهوم است که Sequence پس از چه عددی می‌بایست Restart شود.
Cache: عددی که برای Cache در نظر می‌گیریم، مفهومش این است که چه تعداد از اعداد تولید شده توسط Sequence، قبل از استفاده، می‌تواند در Cache قرار گیرد.

در ادامه با یک مثال ساده، یک Sequence ایجاد می‌نماییم:

```
CREATE SEQUENCE [dbo].[SequenceTest]
 AS [int]
 START WITH 1
 INCREMENT BY 1
 MINVALUE 1
 MAXVALUE 30
 CYCLE
 CACHE
GO
```

در مثال بالا Start with برابر یک است، یعنی اولین عددی که تولید می‌شود، برابر یک است،INCREMENT BY برابر یک است یعنی

در هر بار فراخوانی Sequence یک عدد به عدد تولید شده قبلی افزوده می‌شود. مقدار Minvalue برابر یک است، یعنی کمترین مقداری که Sequence می‌تواند تولید نماید برابر یک است. مقدار Maxvalue برابر 30 است، یعنی بیشترین مقداری که Sequence می‌تواند تولید نماید برابر 30 می‌باشد. و Cycle هم برای Sequence فوق در نظر گرفته شده است، بدین مفهوم ، که Sequence به مقدار 30 برسد، Restart شده و از مقدار یک شروع به تولید اعداد می‌نماید.

برای اینکه بتوانیم مقدار Sequence را بدست آوریم، کافیست از Syntax زیر استفاده نمایید:

```
NEXT VALUE FOR [ database_name . ] [ schema_name . ]  sequence_name
   [ OVER (<over_order_by_clause>) ]
```

به عنوان مثال داریم:

اگر Select بالا را تا 30 بار انجام دهید، برای دفعه 31 مقدار آن یک می‌شود، چون در زمان تعریف Sequence، Cycle را انتخاب کرده بودیم. در غیر اینصورت برای دفعه 31 با خطا زیر مواجه می‌شوید.

```
Msg 11728, Level 16, State 1, Line 1
The sequence object 'SequenceTest' has reached its minimum or maximum value. Restart the sequence
object to allow new values to be generated.
```

یکی از امکانات جالب Sequence این است که شما می‌توانید Sequence را روی یک فیلد Sort شده تنظیم نمایید، برای روش شدن مطلب به مثال زیر توجه نمایید:

در ابتدا مطابق Script زیر جدولی را ایجاد و مقادیری را درون آن درج می‌نماییم:

```
create table Kids
( ID int,
Name varchar(50)
);
Go
insert Kids
 values
  (1,'Emma')
, (1,'Tabitha')
, (2,'Kendall')
, (3,'Delaney')
, (4,'Kyle')
, (5,'Jessica')
, (6,'Josh')
, (7,'Kirsten')
, (8,'Amanda')
, (9,'Jimmy')
;
```

سپس یک Schema به نام Sample ایجاد می‌کنیم:

```
CREATE SCHEMA Samples ;
GO
```

در ادامه یک Sequence به نام Test ایجاد می‌کنیم:

```
CREATE SEQUENCE Samples.Test
    AS tinyint
    START WITH 1
    INCREMENT BY 1 ;
GO
```

حال Query زیر را اجرا می‌نماییم:

```
SELECT NEXT VALUE FOR Samples.Test OVER (ORDER BY Name) AS NutID, ID, Name FROM  test1.Kids
WHERE Name LIKE '%e%' ;
```

در Query بالا روی فیلد Name به صورت صعودی عملیات Sort انجام می‌شود، و سپس Sequence روی آن اعمال می‌گردد، برای روشن‌تر شدن مطلب خروجی را مشاهده نمایید که تعداد رکورد آن 6 میباشد.

امیدوارم مطلب فوق مفید واقع شده باشد.

نظرات خوانندگان

نویسنده: کاربر
تاریخ: ۱۳۹۲/۰۴/۲۹ ۱۷:۵۳

سلام
مشکلی بر IDENTITY وارد هست بوجود آمدن Gap بین داده هاست مثلا با حذف پی در پی سطرهای جدول بین مقادیر تولید شده Gap بوجود می‌آید ولی ما این را نمیخواهیم، نیاز داریم که همیشه مقادیر identity پشت سر هم و متوالی باشند یعنی با حذف یک سطر میانی تمام مقادیر شیفت پیدا کنند. و برای رفع این مساله هیچ راهی وجود ندارد. حتی اگر بخواهیم با تابع row_number مقادیر IDENTITY را بعد متوالی کردن بروز رسانی کنیم باز دستور set insert_identity table_name on این اجازه را نمی‌دهد یعنی این دستور

```
set insert_identity tbl on
update t
set id = new_id
from (select new_id = row_number() over(order by id), *) t
set insert_identity tbl off
```

در کتابی خواندم که این مشکل با sequence حل میشود.
میخوام بدانم چگونه؟

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۲/۰۵/۰۱ ۱۶:۳۰

سلام
شما قادر نیستید یک فیلد Identity را بروز رسانی نمایید، دستور set insert_identity Tablename on به شما اجازه Insert به جدول بدون Identity را می‌دهد، برای اینکه بتوانید Gap مرتبط به فیلد Identity را در جدول برطرف کنید، در ابتدا از جدول مورد نظر خود یک کپی تهیه و جدول اصلی را Truncate کنید، سپس یک Sequencer ایجاد و محتویات جدول کپی را بوسیله Sequencer در جدول اصلی کپی نمایید.
فرض کنیم جدول اصلی Table_3 باشد، ابتدا آن را کپی می‌کنیم در جدولی به نام T

```
Select * into T from table_3
```

سپس دستور Truncate را روی جدول Table_3 اجرا کنید:

```
truncate table dbo.table_3
```

حال یک Sequence ایجاد کنید:

```
CREATE SEQUENCE testEventCounter
    AS int
    START WITH 1
    INCREMENT BY 1 ;
```

در ادامه محتویات جدول کپی را به جدول اصلی منتقل نمایید:

```
SET IDENTITY_INSERT table_3 on
INSERT INTO table_3 (ID, Descritp)
SELECT
      NEXT VALUE FOR testEventCounter AS id
    , Descritp
```

```
FROM T
```

راه دیگر این است که به جای استفاده از Identity از Sequence در فیلد خود استفاده نمایید، بصورت زیر :

```
CREATE TABLE Table3
(
    ID int PRIMARY KEY CLUSTERED
        DEFAULT (NEXT VALUE FOR SequenceTest),
    De nvarchar(300) NULL
) ;
GO
```

در هنگام ایجاد جدول Sequence را به فیلد ID ست کردیم.

حال هر زمانی که بخواهید می‌توانید فیلد ID را مطابق Sequence خود بروز رسانی کنید:

```
Update table3 set id=(NEXT VALUE FOR testEventCounter )
```

موفق باشید و امیدوارم مفید واقع شده باشد

نویسنده: کاربر
تاریخ: ۱۳۹۲/۰۵/۰۱ ۲۰:۸

سلام

خیلی ممنون از پاسخی که دادین. مفید واقع شد.

البته فکر کنم برای گارانتی کردن ترتیب sequence بر اساس ستون id می‌تونیم از OVER(ORDER BY نیز استفاده کنیم.

به این شکل:

```
SET IDENTITY_INSERT table_3 on
INSERT INTO table_3 (ID, Descritp)
SELECT
      NEXT VALUE FOR testEventCounter (OVER ORDER BY ID) AS id
    , Descritp
FROM T
```

به هر شکل با مفهوم جدیدی نیز آشنا شدیم :)

عنوان: آشنایی با Column Store Index در SQL Server 2012
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۷/۲۹ ۱۶:۹
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, SQL Server 2012

Column Store Index یکی از ویژگیهای جدید SQL Server 2012 می باشد، که کارایی Query های قایل اجرا روی دیتابیس‌های با حجم داده ای بسیار بالا را (که اصطلاحا به آنها Data Warehouse یا انبار داده گویند)، چندین برابر بهبود بخشیده است .

قبل از توضیح در مورد Column Store مختصری در مورد نحوه ذخیره سازی داده‌ها در SQL Server می پردازیم. می‌توان گفت در SQL Server دو روش ذخیره سازی وجود دارد،یکی بصورت ردیفی که اصطلاحا به آن Row Store یا Row-Wise گویند، و دیگری بصورت ستونی که اصطلاحا به آن Column Store گویند .

در روش ذخیره سازی Row Store، مقادیر ستونها در یک سطر بصورت متوالی ذخیره می‌شوند، در این روش ذخیره سازی از ساختار B-Tree یا Heap استفاده می‌شود.

یادآوری: در ساختار B-Tree، یک گره Root وجود دارد، و گره بعد از Root گره ای است که آدرس گره راست بعدی و آدرس گره چپ بعدی را در خود نگه می‌دارد.

شکل زیر نمای یک درخت B-Tree می‌باشد:

جهت کسب اطلاعات بیشتر درمورد ساختار B-Tree

یادآوری: وقتی در یک جدول، ایندکسی از نوع Clustered ایجاد نماییم، SQL Server، در ابتدا یک کپی از جدول ایجاد و داده‌های جدول را از نو مرتب می‌نماید، و ساختار صفحه ریشه و دیگر صفحات را ایجاد می‌کند و سپس جدول اصلی را حذف می‌نماید. به جدولی که Clustered Index ندارد، اصطلاحا Heap گویند.

برخلاف ذخیره سازی Row Store، در ذخیره سازی Column Store، داده‌ها بصورت ستونی ذخیره می‌شوند،در این روش داده ها، فشرده سازی می‌شوند و اینکار باعث می‌شود،در زمان درخواست یک Query، نیاز به Disk I/o به حداقل برسد، در نتیجه، زمان و سرعت پاسخگویی به پرس و جوها بسیار افزایش می‌یابد.

شکل زیر نحوه ذخیره سازی داده ها،بصورت Row Store را نمایش می‌دهد:

شکل بالا ذخیره سازی داده ها، در ساختار B-Tree یا Heap را نمایش می‌دهد، در شکل فوق یک جدول چهار ستونی با N سطر (Row) در نظر گرفته شده است.بطوریکه ستونهای هر Row بطور متوالی در یک صفحه (Page) یکسان ذخیره می‌شوند.

شکل زیر نحوه ذخیره سازی داده ها،بصورت Column Store را نمایش می‌دهد:

مطابق شکل،ستونهای مربوط به هر Row،همگی در یک صفحه (Page) یکسان ذخیره شده اند. به عنوان مثال ستون C1 که مربوط به سطر اول (Row1) می‌باشد، با ستون C1 که مربوط به سطر دوم (Row2) می‌باشد، در یک ستون و در یک صفحه (Page1) ذخیره شده اند، و الی آخر ...

سئوال: یکبار دیگر به هردو شکل با دقت نگاهی بیاندازید، عمده تفاوت آنها در چیست؟

جواب: درست حدس زدید، تفاوت بارز بین دو روش Column Store و Row Store در نحوه ذخیره سازی داده‌ها می‌باشد. بطور مثال، فرض کنید،در روش ذخیره سازی Row Store، به دنبال مقادیری از ستون C2 می‌باشید، SQL Server می‌بایست کل رکوردهای جدول (منظور همه Rowها در همه Page ها)را Scan نماید، تا مقادیر مربوط به ستون C2 را بدست آورد.درحالیکه در روش ذخیره سازی Column Store، جهت یافتن مقادیر ستون C2، نیازی به Scan نمودن کل جدول نیست،بلکه SQL Server فقط به Scan نمودن ستون دوم (C2) یا Page2 بسنده می‌نماید.همین امر باعث افزایش چندین برابری، زمان پاسخگویی به هر Query می‌شود.

سئوال: در روش ذخیره سازی Column Store، چگونه مصرف حافظه بهینه می‌شود؟

جواب: واضح است، که در روش Row Store، SQL Server مجبور است، برای بدست آوردن داده‌های مورد نظرتان،کل اطلاعات جدول را وارد حافظه نماید(اطلاعات اضافه ای که به هیچ وجه بدرد، نتیجه پرس و جوی شما نمی‌خورد)، و شروع به Scan داده‌های مد نظر شما می‌نماید.بطوریکه در روش Column Store، SQL Server، فقط ستون داده‌های مورد پرس و جو را در حافظه قرار می‌دهد.(در واقع فقط داده هایی را در حافظه قرار می‌دهد، که شما به آن نیاز دارید)،بنابراین،طبیعی است که در روش Column Store مقدار حافظه کمتری نسبت به روش Row Store در هنگام اجرای Query استفاده می‌شود. به عبارت دیگر می‌توان گفت که در روش Column Store به دلیل، به حداقل رساندن استفاده از Disk I/o سرعت و زمان پاسخگویی به پرس و جوها چندین برابر می‌شود.

برای درک بیشتر Row Store و Column Store مثالی می‌زنیم:

فرض کنید،قصد بدست آوردن ستونهای C1 و C2 از جدول A را داریم، بنابراین خواهیم داشت:

```
Select C1, C2 from A
```

روش Row Store:

در این روش همه صفحات دیسک (مربوط به جدول A) درون حافظه قرار داده می‌شود، یعنی علاوه بر ستونهای C1 و C2، اطلاعات مربوط به ستونهای C3 و C4 نیز درون حافظه قرار می‌گیرد،بطوریکه مقادیر ستونهای C3 و C4 به هیچ وجه مورد قبول ما نیست، و در خروجی پرس و جوی ما تاثیری ندارد، و فقط بی جهت حافظه اشغال می‌نماید.

روش Column Store:

در این روش فقط صفحات مروبط به ستون C1 و C2 در حافظه قرار می‌گیرد.(منظور Page1 و Page2 می‌باشد) بنابراین فقط اطلاعات مورد نیاز در خروجی، در حافظه قرار می‌گیرد.
از دیگر مزایای استفاده از روش Column Store، فشرده سازی داده می‌باشد،برای درک بیشتر توضیح می‌دهم:

همانطور که در اوایل مطلبم به عرض رساندم، در روش Row Store ، داده‌ها در یک سطر و در یک Page ذخیره می‌شوند، بنابراین امکان وجود داده‌های تکراری در یک سطر به حداقل می‌رسد، چرا که، اگر فرض کنیم چهار ستون به نام‌های ID،FirstName،LastName و City، داشته باشیم،در آن صورت بطور حتم،در یک سطر، داده تکراری وجود نخواهد داشت، اما ممکن است در تعداد سطرهای زیاد داده‌های تکراری مانند Firstname یا City و غیرو بوجود بیاید، این موضوع را بیان کردم، چون می‌خواستم عنوان کنم،بسیاری از الگوریتم‌های فشرده سازی از الگوی تکراری بودن داده، جهت فشرده سازی داده‌ها استفاده می‌کنند، به همین جهت فشرده سازی در روش Row Store به حداقل می‌رسد و فضای اشغال شده در حافظه دراین روش بسیار زیاد خواهد بود. اما در روش Column Store ، امکان تکراری بودن مقادیر یک ستون بسیار زیاد است، بطور مثال ممکن است تعداد افرادی را که نام شهر آنها "تهران" باشد مثلا 20 بار تکرار شده باشد، و چون در روش Column Store، ستون‌ها در یک Page ذخیره می‌شوند، بنابراین امکان استفاده از الگوریتمهای فشرده سازی در این روش بسیار بالا می‌باشد، در نتیجه مقدار فضایی را که در حافظه یا دیسک سخت توسط این روش اشغال می‌شود، بسیار کمتر از روش Row Store است.
چه موقع می‌توانیم از Column Store استفاده نماییم:

در تعریف Column Store گفته بودم، روش فوق، جهت بهبود بخشیدن به زمان و سرعت پاسخگویی به Queryهای اجرا شده روی دیتابیس‌های با حجم داده ای بسیار بالا(Data Warehouse) می‌باشد، به بیان ساده‌تر Column Store را روی دیتابیس‌های offline یا دیتابیسهایی که صرفا جهت گزارش گیری مورد استفاده قرار می‌گیرند، تنظیم می‌نمایند.در واقع با تنظیم Column Store Index روی Databaseهای بزرگ مانند Databaseهای بانک‌ها که حجم داده ای میلیونی در جداول آنها وجود دارد، سرعت پاسخگویی Query ها، چندین برابر افزایش می‌یابد.
در یک جدول می‌توانید، هم Column Store Index داشته باشید و هم یک Row Store Index (منظور یک Clustered Index می باشد)

Syntax برای ایجاد Column Store Index به شرح ذیل می‌باشد:

```
CREATE [ NONCLUSTERED ] COLUMNSTORE INDEX index_name
    ON <object> ( column  [ ,...n ] )
    [ WITH ( <column_index_option> [ ,...n ] ) ]
    [ ON {
           { partition_scheme_name ( column_name ) }
           | filegroup_name
           | "default"
         }
    ]
[ ; ]

<object> ::=
{
    [database_name. [schema_name ] . | schema_name . ]
     table_name
```

```
{

<column_index_option> ::=
{
      DROP_EXISTING = { ON | OFF }
    | MAXDOP = max_degree_of_parallelism
 }
```

یک Column Store Index می‌بایست از نوع NONCLUSTERED باشد.

مثال از یک Column Store Index :

```
CREATE NONCLUSTERED COLUMNSTORE INDEX [IX_MyFirstName_ColumnStore]
ON [Test]
(Firstname)
```

در قطعه کد بالا، یک Column Store Index به نام Ix_MyFirstname_ColumnStore روی فیلد Firstname از جدول Test ایجاد شده است.

محدودیت‌های استفاده از Column Store Index به اختصار به شرح ذیل می‌باشد:

زمانی که در یک جدول، یک Column Store Index ایجاد نماییم، جدول ما در حالت Read-only قرار می‌گیرد، بطوریکه از آن پس اختیار Update،Delete و Insert روی جدول فوق را نخواهیم داشت. برای اینکه بتوانید عملیات Update، Insert یا Delete را انجام دهید، میبایست Column Store Index جدول مربوطه را Disable نمایید، و برای فعال نمودن Column Store Index، می‌بایست آن را Rebuild نمایید، با کلیک راست روی ایندکس ایجاد شده در SQL Server2012 موارد Disable و Rebuild قابل مشاهده می‌باشد.

یا بوسیله Scriptهای زیر می‌توانید، عملیات Disable یا Rebuild را روی Column Store Index انجام دهید:

```
ALTER INDEX [IX_MyFirstName_ColumnStore] ON [Test] DISABLE
ALTER INDEX [IX_MyFirstName_ColumnStore] ON [Test] Rebuild
```

بیشتر از یک Column Store Index نمی‌توانید روی یک جدول ایجاد نمایید.
در صورتی که تمایل داشته باشید بوسیله Alter ، نوع فیلدی (Type)، را که Column Store Index روی آنها اعمال گردیده است، تغییر دهید، در ابتدا می‌بایست Column Store Index، خود را Drop یا حذف نمایید، سپس عملیات Alter را اعمال کنید، در غیر اینصورت با خطای SQL Server مواجه می‌شوید.
یک Column Store Index می‌تواند روی 1024 ستون در یک جدول اعمال گردد.
یک Column Store Index نمی توانند، Unique باشد و نمی‌توان از آن به عنوان Primary Key یا Foreign Key استفاده نمود.

یاد آوری: با توجه به مزایای استفاده از Column Store Index، باید بگویم که در حجم‌های داده ای کم استفاده از Row Store Index بهتر می‌باشد. پیشنهاد مایکروسافت برای استفاده از Column Store Index برای دیتابیس‌های با حجم داده ای بسیار بالا می‌باشد.
موفق باشید

منابع: ”Columnstore Indexes: A New Feature in SQL Server known as Project “Apollo
Columnstore Indexes
Fundamentals of Columnstore Index
SQL Server 2012 Column Store Index Example

نظرات خوانندگان

نویسنده: صالح
تاریخ: ۱۳۹۱/۰۷/۳۰ ۰:۵۷

درود بر شما
بسیار بهره جستم از مطلبتون. عالی بود.

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۷/۳۰ ۱۱:۱۶

ممنون از لطفا شما

نویسنده: علیرضا جهانشاهلو
تاریخ: ۱۳۹۱/۰۷/۳۰ ۲۲:۵۷

بسیار بهره بردیم از مطلب خوبتون
با تشکر

نویسنده: libertad
تاریخ: ۱۳۹۱/۰۸/۰۳ ۱۰:۳۴

آیا چنین امکانی در oracle هم وجود دارد؟

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۸/۰۳ ۱۲:۳۹

بله، چنین امکانی موجود است،برای اطلاعات بیشتر می‌توانید به آدرس زیر مراجعه کنید:
Compressing Columns

نویسنده: مهران زند
تاریخ: ۱۳۹۱/۰۸/۰۵ ۱۶:۲۸

بسیار عالی و ممنون. نتیجتا این گونه ایندکس گذاری برای دیتابیس‌های که درد حال رشد هستند یعنی مداوم در حال,update insert.. هستند به درد نخواهد خورد.

نویسنده: علی قمشلویی
تاریخ: ۱۳۹۱/۰۸/۰۵ ۲۰:۵۵

با سلام و تشکر به خاطر مطلب عالی و فوق العاده قابل فهمتون
یکی از سئوالاتی که همیشه برام هنگام کوئری گرفتن از دیتابیس پیش می‌آمد این بود که :

```
select c1 from A1 where c1=x

select * from A1 where c1=x
```

کدامیک دارای کارایی بیشتری می‌باشند که دراین مقاله به طور عالی توضیح داده شده بود.
البته جوابشو نمیدم تا دوستان این مقاله زیبا را از دست ندهند.

نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۸/۰۵ ۲۲:۵۸

البته موردی که در این مقاله عنوان شده یکی از مشکلات Select * هست...

نویسنده: ali
تاریخ: ۱۳۹۲/۰۹/۰۵ ۲۳:۴۵

با سلام یک جایی مطلب شما گنگ ماند " یعنی علاوه بر ستونهای C1 و C2، اطلاعات مربوط به ستونهای C3 و C4 نیز درون حافظه قرار می‌گیرد " پس چرا میگن تا آنجا که میشه برای بالا بردن performance در دستور select حتما تعداد فیلدهای آن مشخص باشد با آنچه شما گفتید c3 وc4 درون حاظه لود میشوند اینطور نیست ممنون

عنوان: آشنایی با Contained Databases در SQL Server 2012
نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۰۸/۱۰ ۲۲:۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, SQL Server 2012

مقدمه: بعضی وقت‌ها به هر دلیلی لازم است پایگاه داده برنامه هایمان را به سرور دیگری انتقال دهیم . برخلاف Oracle که انتقال پایگاه داده به سرور دیگر کار مشکل و تخصصی می‌باشد در SQL Server براحتی با یک Detach و Attach مجدد این انتقال انجام خواهد شد. اما اطلاعات Logins در پایگاه داده Master سرور ذخیره شده است و با فایل پایگاه داده انتقال نمی‌یابد و به هیمن خاطر برخی مواقع پس از Attach پایگاه داده با پیغام The database is not accessible یا Cannot open user default database مواجع خواهیم شد که مشکل بخاطر همین نبود کاربران سرور قبلی و Owner متفاوت پایگاه داده می‌باشد که با استفاده از رویه ذخیره شده sp_change_users_login این مشکل قابل حل می‌باشد اما در SQL Server 2012 راحل بهتری فراهم شده است.

Contained Databases در Sql Server 2012

یکی از امکانات اضافه شده در SQL Server 2012 امکان Contained Databases هست که برای حل همین مشکل و رفع کامل وابستی پایگاه داده با Instance نصب شده و خاصیت قابل حمل کامل databaseها می‌باشد. در ادامه نحوه استفاده از این امکان را بررسی خواهیم کرد:

فعال کردن خاصیت Contained Databases

قبل از استفاده از Contained Databases می بایست این امکان را فعال کرد. برای این کار می‌توانید از SQL Server Management Studio یا T-SQL commands استفاده نمایید. بر روی نام Instance راست کلیت کنید و گزینه Properties را انتخاب نمایید. از گزینه Advanced که در شکل زیر مشاهده می‌نمایید خاصیت Enable Contained Databases را بر روی True قرار دهید.

یا می‌توانید از sp_configure این کار را انجام دهید.دستورات زیر این موضوع را نشان می‌دهد.

```
sp_configure 'show advanced options',1
GO
RECONFIGURE WITH OVERRIDE
GO
sp_configure 'contained database authentication',1
GO
RECONFIGURE WITH OVERRIDE
```

```
GO
```

ایجاد یا تغییر یک پایگاه داده از نوع Contained Databases

برای ایجاد یک پایگاه داده با این خاصیت یا تغییر پایگاه داده موجود کافیست مقدار گزینه Containment type را بر روی Partial قرار دهید. برای پایگاه داده موجود از پنجره Properties پایگاه داده صفحه Options را انتخاب کنید.

ایجاد یک کاربر برای پایگاه داده Contained Databases

برای تعریف یک کاربر در سطح پایگاه داده پوشه Security پایگاه داده خود را باز کنید بر روی پوشه Users راست کیلک و گزینه New User را انتخاب نمایید از گزینه User type که در شکل زیر نشان داده شده است SQL user with password را انتخاب نمایید و نام کاربر و رمز عبور و تکرار آن را وارد نمایید. کاربر ایجاد شده در سطح پایگاه داده می‌باشد و با انتقال به سرور دیگر نیز قابل دسترسی می‌باشد.

اتصال به پایگاه داده Contained Databases

برای اتصال به پایگاه داده کافیست در حالت SQL Server Authentication نام کاربری و رمز عبور جدید را وارد و گزینه Options را انتخاب و از برگه Additional Connection Parameters نام پایگاه داده مورد نظر را مانند شکل زیر وارد نمایید پس از ورود تنها پایگاه داده خود را مشاهده می‌نمایید. یکی از کاربرهای این قابلیت برای مدیران سرور پایگاه داده می‌باشد که بدون استفاده از مجوز sysadmin به کاربران اجازه دسترسی را می‌دهد.

Connect to Server
Microsoft SQL Server 2012
Login
Connection Properties
Additional Connection Parameters
Enter additional connection string parameters (will be sent in clear text):
Initial Catalog = < Database Name >
(Note: Connection string parameters override graphical selections on other panels)
Connect
Cancel
Help
Options <<

نظرات خوانندگان

نویسنده: ابراهیم شادافروغ
تاریخ: ۱۳۹۱/۰۸/۱۱ ۱۱:۵۳

مقاله عالی بود من با این مشکل در یکی از پروژه‌های کاری ام برخورد کردم و با مطالعه و تمرین و خطا موفق به پیدا کردن راه حل این مشکل شدم
خیلی خوبه که راه کار حل موضوعاتی این چنین در وب فارسی وجود داشته باشه

عنوان: آشنایی با SQL Server Data Tools
نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۰۸/۲۴ ۱۵:۱۰
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, VisualStudio.NET 2012, SQL Server 2012, Data Tools

مقدمه

یکی از امکانات اضافه شده در Sql Server 2012 ابزار Sql Server Data Tools یا به اختصار SSDT می‌باشد. این ابزار در واقع جایگزین ابزار Business Intelligence Development Studio می باشد که همه امکانات قبلی را داشته و همچنین قابلیت‌های جدیدی نیز به آن اضافه شد است. اما کاربرد این ابزار ایجاد محیطی یکپارچه برای طراحی و توسعه تمامی نسخه‌های پایگاه داده Sql Server با استفاده از Visual Studio می‌باشد. در این مقاله من به بیشتر بر روی امکانات زیر تمرکز دارم:

Installation

Creating a SQL Server Database Project

Modify Database Schema

Schema Compare and Update

Snapshot Project

Publish

نصب Sql Server Data Tools

آخرین نسخه این ابزار را از این آدرس دانلود کنید یا زمان نصب Sql Server 2012 گزینه آن را انتخاب کنید

ایجاد یک پروژه جدید از نوع SQL Server Database Project

پس از نصب SSDT شما از طریق Visual Studio 2012 Shell که همراه SqlServer 2012 نصب می‌شود یا با Visual Stadio 2012 یک پروژه جدید از نوع SQL Server Database Project ایجاد کنید.

Server Database Project Project به شما امکان توسعه پایگاه داده Sql Server را با استفاده از محیط یکپارچه Visual Studio با در اختیار گذاشتن ابزارهای همچون navigation, intellisense, validation, debugging, declarative editing و غیره را می‌دهد. شی پایگاه داده شما در پروژه و فایل‌های مجزا ذخیره می‌شود مثل آن که شما در حال توسعه برنامه C# یا VB.NET هستید.

ورود پایگاه داده موجود به پروژه SQL Server Database

شما می‌توانید شمای پایگاه داده موجود یا SQL Server DAC Package File (.dacpac) یا هر T-SQL دیگری را با راست کلیک بر روی پروژه و انتخاب گزینه Import به پروژه خود اضافه و تغییرات لازم را اعمال نماید شکل زیر پنجره Import Database را نشان می‌دهد:

پنجره Solution Explorer امکانات زیر را در اختیار شما قرارمی دهد:

- اضافه کردن اشیای همچون Table، View و غیره با راست کیلک کردن بر روی پروژه و انتخاب گزینه Add
- ویرایش اشیا موجود با دبل کیلک کردن بر روی اشیا
- مقایسه شمای پایگاه داده با پایگاه داده دیگر و یا Microsoft SQL Server DAC Package File کافیست بر روی پروژه راست کیلک و گزینه Schema Compare را انتخاب نمایید.
- ایجاد یک Snapshot از شمای پایگاه داده در یک Microsoft SQL Server DAC Package File . یک snapshot برای ایجاد یک پایگاه داده یا ورود در پروژه ای دیگر یا در مقایسه دو پایگاه داده کاربرد دارد.
- انتشار پایگاه داده که امکان ایجاد پایگاه داده یر روی یک سرور SQL Server را فراهم می‌نماید.

مقایسه شمای دو پایگاه داده با هم

SSDT ابزاری برای مقایسه دو پایگاه داده و بروز رسانی پایگاه داده مقصد از روی شمای منبع از طریق ایجاد یک Script یا به صورت مستقیم را می‌دهد.

انتشار پروژه با استفاده از SQL Server Data Tools

این ابزار امکان گسترش پروژه شما بر روی SQL Server 2005, 2008, 2008 R2, 2012, یا SQL Azure instance را می‌دهد. برای انتشار کافیت از پنجره Solution Explorer بر روی پروژه راست کیلک و گزینه Publish را انتخاب کنید.

پنجره SQL Server Object Explorer

این پنجره با اتصال به سرور SQL Server امکان دسترسی و مدیریت پایگاه داده را به ما می‌دهد. علاوه بر دسترسی به اجزای یک پایگاه داده امکان مدیریت پوشه‌های Server Objects , Security نیز فراهم می‌باشد. همچنین امکان اتصال به Sql Server Express , localdb نیز وجود دارد.

عنوان: استفاده از Sparse Columns در SQL Server 2012
نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۰۹/۱۱ ۱۶:۳۰
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, SQL Server 2012, T-SQL, DML

مقدمه

مقدار null به معنی پوچ و هیچ می‌باشد اما زمانی که در مقدار دهی جداول از آن استفاده می‌نمایم با توجه به نوع آن ستون فضای متفاوتی اشتغال می‌نماید. شاید در پایگاه داده‌های کوچک زیاد مطرح نباشد اما زمانی که حداقل چند گیگ حجم آن باشد و فرضا 20 تا 30 درصد آن از مقادیر null پر شده باشد فضای زیادی از پوچ گرفته شده است این در حالی است که خیلی از توسعه دهندگان اصلا به اهمیت استفاده از null توجهی نمی‌کنند و از مقادیری غیر معتبری مثل 0 یا 1- در آن ستون به جای null استفاده می‌کنند.

SQL Server Sparce Columns

sparse column یا ستون‌های تنک قابلیتی از که از SQL Server 2008 اضافه شده و به ستون‌های عادی امکان استفاده بهینه از فضای ذخیره شده برای مقادیر null را می‌دهد. در واقع sparse column فضای مورد نیاز برای مقادیر null نسبت به مقادیر غیر null را کاهش می‌دهد. با استفاده از sparse column فضای ذخیره شده حداقل 20 تا 40 درصد کمتر خواهد شد.

ویژگی‌های Sparse Columns

SQL Server Database Engine از کلمه کلیدی SPARSE برای تعریف یک ستون که مقادیر آن می‌بایست بهینه شود استفاده می‌نماید.

نمای Catalog جداول با ستون sparse شبیه جداول معمولی می‌باشد.

مقدار برگشتی از تابع COLUMNS_UPDATED با ستون sparce متفاوت از ستون معمولی است.

در نوع داده‌های زیر امکان استفاده از sparce columns را ندارند:

text	geography
timestamp	geometry
user-defined data types	image
	ntext

sparse column فضای بیشتری برای ذخیره داده‌های غیر null نسبت به داده‌های نشانه گذاری نشده با SPARSE لازم دارد و این فضا4 بایت بیشتر از ستون معمولی است. برآورد فضای ذخیره شده براساس نوع داده با طول ثابت در جدول زیر آورده شده است:

نوع داده	بایت بدون sparse	بایت sparse	درصد null
bit	0.125	5	98%
tinyint	1	5	86%
smallint	2	6	76%
int	4	8	64%
bigint	8	12	52%
real	4	8	64%
float	8	12	52%
smallmoney	4	8	64%
money	8	12	52%

نوع داده	بایت بدون sparse	بایت sparse	درصد null
smalldatetime	4	8	64%
datetime	8	12	52%
uniqueidentifier	16	20	43%
date	3	7	69%

نوع داده با دقت - وابسته به طول

نوع داده	بایت بدون sparse	یابت sparse	درصد null
(datetime(2	6	10	57%
(datetime(2	8	12	52%
(time(0	3	7	69%
(time(7	5	9	60%
(datetimetoffset(0	8	12	52%
(datetimetoffset (7	10	14	49%
decimal/numeric(1,s)	5	9	60%
decimal/numeric(38,s)	17	21	42%
vardecimal(p,s)			

نوع داده - داده وابسته به طول

نوع داده	بایت بدون sparse	یابت sparse	درصد null
sql_variant	2*	2*	60%
varchar or char	2*	4*+	60%
nvarchar or nchar	2*	4*	60%
varbinary or binary	2*	4*	60%
xml	2*	4*	60%
hierarchyid	2*	4*	60%

محدویت‌های استفاده از Sparse columns

sparse column می بایست nullable باشد و نمی‌تواند ROWGUIDCOL یا IDENTITY باشد.

sparse column مقدار پیش فرض نمی‌تواند داشته باشد

ستون محاسبه ای نمی‌تواند sparse باشد

sparse column نمی‌تواند بخشی از clustered index یا unique primary key index باشد

sparse column نمی تواند بخشی از user-defined table باشد

مثالی از کاربرد Sparse columns

```
CREATE TABLE Employees_sparse (
   EMP_ID INT IDENTITY(5001,1) PRIMARY KEY,
   SSN CHAR(9) NOT NULL,
   TITLE CHAR(10) SPARSE NULL,
```

```
   FIRSTNAME VARCHAR(50) NOT NULL,
   MIDDLEINIT CHAR(1) SPARSE NULL,
   LASTNAME VARCHAR(50) NOT NULL,
   EMAIL CHAR(50) SPARSE NULL)
GO
```

```
CREATE TABLE Employees (
   EMP_ID INT IDENTITY(5001,1) PRIMARY KEY,
   SSN CHAR(9) NOT NULL,
   TITLE CHAR(10) NULL,
   FIRSTNAME VARCHAR(50) NOT NULL,
   MIDDLEINIT CHAR(1) NULL,
   LASTNAME VARCHAR(50) NOT NULL,
   EMAIL CHAR(50) NULL)
GO
```

در این دو جدول یکی با سه ستون Sparse و دیگری بدون Sparse ایجاد شده و با 50000 ردیف داده پر شده است حال با رویه ذخیره شده sp_spaceused می‌توان فضای ذخیره شده دو جدول را باهم مقایسه نمود.

```
sp_spaceused 'Employees'
GO
sp_spaceused 'Employees_sparse'
```

Results | Messages

	name	rows	reserved	data	index_size	unused
1	Employees	50000	5064 KB	5008 KB	32 KB	24 KB

	name	rows	reserved	data	index_size	unused
1	Employees_sparse	50000	3080 KB	3064 KB	16 KB	0 KB

البته ذکر این نکته گفتی است که بهتر است از این تکنیک برای جداولی که تعداد زیادی ستون null دارند استفاده شود.

عنوان: **تغییر نام پایگاه داده و فایل هایش در SQL Server 2012**
نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۰۹/۱۲ ۲۱:۵۵
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, SQL Server 2012, T-SQL

بعضی وقت‌ها به هر علتی لازم است پایگاه داده و فایل هایش را تغییر نام دهیم. اگر در اینترنت جستجو کنیم روش‌های مختلفی برای تغییر نام مثل تغییر با Management Studio یا T-SQL یا روش‌های دیگری یافت می‌شود. اما اکثرا در بین انجام به مشکلی غیر قابل پیش بینی بر می‌خوریم. پایگاه داده در حالت آفلاین یا Pending قرار گرفته و به خطاهای نا مفهومی بر می‌خوریم. حالا باید دوباره کلی جستجو کنیم تا مشکل بوجود آمده را حل نمائیم.

بهترین روش تغییر نام پایگاه داده

بهترین روش استفاده از قابلیت Copy Database خود SQL Server است است که به راحتی این کار را برای ما انجام می‌دهد.
بر روی پایگاه داده مورد نظر راست کلیک کرده و از گزینه Tasks گزینه Copy Database را انتخاب کنید.
پس از ظاهر شدن پنجره کپی گزینه Next را انتخاب و در مرحله مبدا و مقصد، سرور جاری را انتخاب کنید و به مرحله بعد بروید.
این برای زمانی است که شما می‌خواهید پایگاه داده را در سرور دیگری کپی نماید
در پنجره Transfer Method دو روش Detach and Attach و استفاده از SQL Management Object وجود دارد که با همان روش اول به مرحله بعد بروید

در مرحله بعد نام پایگاه داده شما انتخاب شده به مرحله بعد بروید.
مرحله بعد پیکربندی پایگاه داده مقصد می‌باشد که نام و مسیر پایگاه داده جدید را می‌توانید مشخص نمایید.

این عملیات با SQL Server Agent صورت می‌پذیرد به همین خاطر Agent می‌بایست نصب و Start شده باشد.
با انتخاب گزینه Next مراحل بعد را رد کرده تا عملیات آغاز شود.
در مرحله آخر پایگاه داده قبلی را حذف نمایید.

نظرات خوانندگان

نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۰۴/۰۶ ۱۱:۵۲

خیلی متشکرم.
البته یک سوال هم دارم. تصور کنید حجم پایگاه داده‌ی من 400 گیگابایت می‌باشد و کل فضای خالی موجودم 350 گیگابایت می‌باشد در این صورت امکان استفاده از این روش وجود ندارد؟

عنوان: **اعمال صفحه بندی به کمک OFFSET و FETCH در SQL Server 2012**
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۱۵ ۱۶:۱۵
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server 2012, T-SQL

در T-SQL 2012 قابلیت صفحه بندی، نمایش خروجی یک Query فراهم گردیده است، که برای نرم افزارهای تحت وب بسیار پرکاربرد میباشد، به عنوان مثال، از جمله کاربردهای بارز آن، می‌توان به نمایش نتیجه یک جستجو بصورت صفحه بندی با تعداد رکورد محدود،اشاره نمود.

مایکروسافت برای ایجاد قابلیت صفحه بندی و محدود نمودن نمایش خروجی یک Query، تغییراتی را در Syntax مربوط به Order by ایجاد نموده است، که در ذیل مشاهده می‌نمایید:

```
ORDER BY order_by_expression
    [ COLLATE collation_name ]
    [ ASC | DESC ]
    [ ,...n ]
[ <offset_fetch> ]


<offset_fetch> ::=
{
    OFFSET { integer_constant | offset_row_count_expression } { ROW | ROWS }
    [
      FETCH { FIRST | NEXT } {integer_constant | fetch_row_count_expression } { ROW | ROWS } ONLY
    ]
}
```

OFFSET (نقطه شروع) : شامل یک پارامتر است،بطوریکه،پارامتر فوق می‌تواند یک عدد (integer_constant) یا یک عبارت (offset_row_count_expression) بپذیرد. در اینجا منظور از عبارت می‌تواند یک Subquery باشد، که خروجی آن فقط یک مقدار عددی است. یا یک متغیر و غیرو...

در مورد ROW یا ROWS باید بگویم باهم فرقی ندارند.

FETCH : همانند OFFSET شامل یک پارامتر است، و پارامتر آن می‌تواند یک عدد یا عبارت بپذیرد.

Next یا First نیز با هم تفاوتی ندارند و جهت سازگاری با ANSI می‌باشند.

OFFSET : در واقع تعداد سطر قابل حذف، پیش از نمایش اولین سطر در خروجی را بیان می‌کند.

FETCH : بیانگر تعداد رکورد قابل نمایش در یک صفحه می‌باشد.

برای درک بیشتر مثالی می‌زنیم:

ابتدا بوسیله Script زیر یک جدول ایجاد می‌نماییم، سپس چند رکورد درون آن درج می‌کنیم:

```
Create Table Testoffset
(BusinessEntityID int,
FirstName varchar(100) ,
LastName varchar(100)
);

Insert into Testoffset (BusinessEntityID,FirstName,LastName)
            Values(1,'Ken','Sánchez')
    ,(2,'Terri','Duffy')
,(3,'Roberto','Tamburello')
,(4,'Rob','Walters')
,(5,'Gail','Erickson')
,(6,'Jossef','Goldberg')
,(7,'Dylan','Miller')
,(8,'Diane','Margheim')
,(9,'Gigi','Matthew')
,(10,'Michael','Raheem')
```

در ادامه Script زیر را اجرا نمایید، تا تعداد رکوردهای درج شده را مشاهده کنید:

	BusinessEntityID	FirstName	LastName
1	1	Ken	S?nchez
2	2	Terri	Duffy
3	3	Roberto	Tamburello
4	4	Rob	Walters
5	5	Gail	Erickson
6	6	Jossef	Goldberg
7	7	Dylan	Miller
8	8	Diane	Margheim
9	9	Gigi	Matthew
10	10	Michael	Raheem

در شکل، سه سطر (منظور رکورد 4و5و6) در کادر قرمز رنگ دیده می‌شود، می‌خواهیم Script ی ایجاد نماییم، که فقط سه سطر فوق را نمایش دهد. بنابراین خواهیم داشت:

```
SELECT BusinessEntityID, FirstName, LastName
FROM Testoffset
ORDER BY BusinessEntityID
OFFSET 3 ROWS
FETCH First 3 ROWS only
```

خروجی:

Results | Messages | Execution plan

	BusinessEntityID	FirstName	LastName
1	4	Rob	Walters
2	5	Gail	Erickson
3	6	Jossef	Goldberg

اگر به Query اجرا شده دقت کنیم. در قسمت Order By جلوی Offset مقدار 3 اختصاص داده شده بود، یعنی نقطه شروع از سطر چهارم میباشد، به عبارت دیگر مقداری که به Offset اختصاص داده می‌شود، به SQL Server می‌فهماند،چه تعداد رکورد را نمایش ندهد. اگر شکل اول و دوم را با هم مقایسه نمایید، براحتی متوجه می‌شوید که OFFSET نقطه شروع را مشخص کرده است. مقداریکه برای Fetch در نظر گرفته شده بود برابر 3 است، که بیانگر تعداد سطر نمایش داده شده در خروجی از نقطه آغازین (offset) می‌باشد.

امیدوارم مفید واقع شده باشد.

نظرات خوانندگان

نویسنده: Ara
تاریخ: ۱۳۹۱/۰۹/۲۸ ۲۲:۳۴

بعضی وقتها مایکروسافت یکسری کارها رو اونقدر سخت می‌کنه !

paging هم یکی از همون‌ها بود تازه داره پخته می‌شه

این syntax من رو یاد mysql انداخت

```
SELECT column FROM table
LIMIT 10 OFFSET 10
```

نویسنده: شیرزادیان
تاریخ: ۱۳۹۱/۰۹/۲۹ ۰:۱۰

سلام
می‌خواستم بدونم امکان هست مرتب سازی را به صورت پارامتری ارسال کنیم. یعنی می‌خواهم به نوعی صفحه بندی را در وب همراه با مرتب سازی دلخواه کاربر انجام بدهم. متشکرم

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۲۹ ۸:۵

سلام
من منظور سئوال شما رو،بدرستی متوجه نشدم، به هرحال اگر بخواهید با کد نویسی سمت سرور، Script خود را Generate نمایید، اینکار، بستگی به نگرش کدنویسی تان و Interface ی که در اختیار کاربر قرار می‌دهید، دارد.
اگر بخواهید در SQL Server اینکار را انجام دهید، با استفاده از Case ، در قسمت Order By می‌توانید اینکار را انجام دهید. به عنوان مثال:

```
DECLARE @Varsort varchar(50)
DECLARE @Varsort1 varchar(50)

SET  @Varsort=''
SET  @Varsort1='BusinessEntityID'

SELECT BusinessEntityID, FirstName, LastName
FROM Testoffset
ORDER BY case when  @Varsort='Firstname'then  Firstname End ASC,
         case when  @Varsort1= 'BusinessEntityID'then  BusinessEntityID End ASC
OFFSET 3 ROWS
FETCH First 3 ROWS only
```

امیدوارم پاسخ تان را گرفته باشید.

نویسنده: سعید شیرزادیان
تاریخ: ۱۳۹۱/۰۹/۲۹ ۲۰:۴۸

سلام
ممنون از جواب شما. فقط اگر تعداد ستونها بیشتر بود و یا اینکه کاربر درخواست مرتب سازی بصورت صعودی و نزولی را هم داشت (در این مثال شما به صورت پیش فرض مرتب سازی صعودی را اعمال کردید) راهکاری دارید؟
با تشکر

نویسنده: Farahmandkhah
تاریخ: ۱۳۹۱/۰۹/۳۰ ۶:۵۶

سلام
دوست عزیز، مثالی که برای شما زده شد، امکان سورت دلخواه را فراهم می‌کند، شما می‌توانید، هر کدام را به دلخواه نزولی یا صعودی نمایید، محدودیتی ندارد، برای مطالعه بیشتر می‌توانید به آدرس زیر مراجعه نمایید: Dynamic/Conditional Order By Clause in SQL Server/T-SQL

یادآور شوم،چنانچه Performance برای شما اهمیت دارد، بهتر است از Case در Order by استفاده ننمایید، و بهتر است در زمان Run Time از طریق کد نویسی سمت سرور، Script خود را Generate نمایید، به عنوان مثال

```
SELECT BusinessEntityID, FirstName, LastName
FROM Testoffset
ORDER BY BusinessEntityID Desc,Firstname ASC
OFFSET 3 ROWS
FETCH First 3 ROWS only
```

موفق باشید.

عنوان: **آشنایی با Window Function ها در SQL Server بخش اول**
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۱۷ ۲۳:۲۵
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, SQL Server 2012, T-SQL

Window Functionها برای اولین بار در نسخه SQL Server 2005 ارائه گردیدند، و در ورژن‌های جدیدتر SQL Server، به تعداد این فانکشنها افزوده شده است.

تعریف Window Function :

معمولا از این نوع فانکشنها روی مجموعه ای از ROWهای یک جدول، در جهت اعمال عملیاتهای محاسباتی ،ارزیابی داده ها، رتبه بندی و غیرو... استفاده می‌گردد، به بیان ساده‌تر بوسیله Window Functionها می‌توان، ROWهای یک جدول را گروه بندی نمود. و روی گروه‌ها از توابع جمعی (Aggregate Functions) استفاده کرد. این نوع فانکشنها از قابلیت و انعطاف پذیری زیادی برخوردار می‌باشند، و بوسیله آنها می‌توان نتایج (خروجی) بسیار مفیدی از Query ها، بدست آورد، معمولا از این نوع فانکشنها در Data Mining (داده کاوی) و گزارشگیری‌ها استفاده می‌گردد. و آگاهی و روش استفاده از Window Functionها برای برنامه نویسان و DBA ها، می‌تواند بسیار مفید باشد.
مفهوم Window Function مطابق استاندارد ISO و ANSI می‌باشد، و دیتابیس هایی همچون Oracle،DB2،Sybase از آن پشتیبانی می‌نمایند.برای اطلاعات بیشتر می‌توانید به سایت‌های زیر مراجعه کنید: SQL:2003
و SQL:2008

کلمه "Window" در Window Function، به مجموعه ROW هایی اشاره می‌کند، که محاسبات و ارزیابی و غیرو... روی آنها اعمال می‌گردد.
Window Functionها برای ارائه قابلیت‌های خود، از Over Clause استفاده می‌کنند. اگر مقاله آشنایی با Row_Number،Rank،Dense_Rank،NTILE را مطالعه کرده باشید، می‌توان هریک از آنها را یک Window Function دانست.
برای شروع، به بررسی Over Clause می‌پردازیم، و Syntax آن به شرح ذیل می‌باشد:

```
OVER (
       [ <PARTITION BY clause> ]
       [ <ORDER BY clause> ]
       [ <ROW or RANGE clause> ]
      )

<PARTITION BY clause> ::=
PARTITION BY value_expression , ... [ n ]

<ORDER BY clause> ::=
ORDER BY order_by_expression
    [ COLLATE collation_name ]
    [ ASC | DESC ]
    [ ,...n ]

<ROW or RANGE clause> ::=
{ ROWS | RANGE } <window frame extent>

<window frame extent> ::=
{   <window frame preceding>
  | <window frame between>
}

<window frame between> ::=
  BETWEEN <window frame bound> AND <window frame bound>

<window frame bound> ::=
{   <window frame preceding>
  | <window frame following>
}

<window frame preceding> ::=
{
    UNBOUNDED PRECEDING
  | <unsigned_value_specification> PRECEDING
  | CURRENT ROW
}
```

```
<window frame following> ::=
{
    UNBOUNDED FOLLOWING
  | <unsigned_value_specification> FOLLOWING
  | CURRENT ROW
}

<unsigned value specification> ::=
{  <unsigned integer literal> }
```

OVER دارای سه آرگومان اختیاری است که هر کدام را به تفصیل بررسی می‌کنیم:

1- PARTITION BY clause : بوسیله این پارامتر می‌توانیم Rowهای یک جدول را گروه بندی نماییم. این پارامتر یک value_expression می پذیرد. یک Value_expression می‌تواند نام یک ستون ، یک Scalar Function ، Scalar Subquery و غیرو باشد.

2- ORDER BY clause : از نامش مشخص است و برای Sort استفاده می‌شود، و ویژگی‌های Order By در آن اعمال می‌گردد. به جز Offset.

3- ROW or RANGE clause :این پارامتر بیشتر برای محدود نمودن Row در یک Partition (گروه) مورد استفاده قرار می‌گیرد، به عنوان مثال نقطه شروع و پایان را می‌توان بوسیله پارامتر فوق تعیین نمود.

Row و Range نسبت به هم یک تفاوت عمده دارند،و آن این است که، اگر از ROW Clause استفاده نمایید، ارتباط ROWهای قبلی یا بعدی، نسبت به Row جاری،بصورت فیزیکی (physical association) سنجیده می‌شود، بطوریکه با استفاده از Range Clause ارتباط سطرهای قبلی و بعدی، نسبت به سطر جاری بصورت منطقی (logical association) در نظر گرفته می‌شود. ممکن است درک این مطلب کمی سخت باشد، در ادامه با مثالهایی که بررسی می‌نماییم، براحتی تفاوت این دو را متوجه می‌شوید.

Row یا Range در قالب‌های متفاوتی مقدار می‌پذیرند، که هر کدام را بررسی می‌کنیم:

UNBOUNDED PRECEDING : بیانگر اولین سطر Partition می‌باشد. UNBOUNDED PRECEDING فقط نقطه شروع را مشخص می‌نماید.

UNBOUNDED FOLLOWING : بیانگر آخرین سطر Partition می‌باشد. UNBOUNDED FOLLOWING فقط نقطه پایانی را مشخص می‌نماید.

CURRENT ROW : اولین سطر جاری یا آخرین سطر جاری را مشخص می‌نماید.

n PRECEDING یا <unsigned value specification> PRECEDING : تعداد سطرهای قبل از سطر جاری را تعیین می‌کند، n یا <unsigned value specification> تعداد سطرهای قبل از سطر جاری را تعیین می‌نماید. از n PRECEDING نمی توان برای Range استفاده نمود.

n FOLLOWING یا <unsigned value specification> FOLLOWING : تعداد سطرهای بعد از سطر جاری را تعیین می‌کند، n یا<unsigned value specification> تعداد سطر های بعد از سطر جاری را تعیین می‌نماید. از n FOLLOWING نمی توان برای Range استفاده نمود.

BETWEEN <window frame bound > AND <window frame bound> : از چارچوب فوق برای Range و Row می‌توان استفاده نمود، و نقطه آغازین و نقطه پایانی توسط قالب فوق تعیین می‌گردد. نکته قابل توجه آن است که نقطه پایانی نمی‌تواند، کوچکتر از نقطه آغازین گردد.

در ادامه برای درک هرچه بیشتر تعاریف بیان شده، چندین مثال می‌زنیم و هر کدام را بررسی می‌نماییم:

در ابتدا Script زیر را اجرا نمایید، که شامل جدولی به نام Revenue (سود،درآمد) و درج چند درکورد در آن:

```
CREATE TABLE REVENUE
(
[DepartmentID] int,
[Revenue] int,
[Year] int
);

insert into REVENUE
values (1,10030,1998),(2,20000,1998),(3,40000,1998),
 (1,20000,1999),(2,60000,1999),(3,50000,1999),
 (1,40000,2000),(2,40000,2000),(3,60000,2000),
 (1,30000,2001),(2,30000,2001),(3,70000,2001)
```

مثال اول : می‌خواهیم براساس فیلد DepartmentID جدول Revenue را Partition بندی نماییم و از توابع جمعی AVG و SUM روی فیلد درآمد(Revenue) استفاده کنیم.

ابتدا Script زیر را اجرا می‌کنیم:

```
select *,
 avg(Revenue) OVER (PARTITION by DepartmentID) as AverageRevenue,
 sum(Revenue) OVER (PARTITION by DepartmentID) as TotalRevenue
from REVENUE
order by departmentID, year;
```

خروجی بصورت زیر خواهد بود:

	DepartmentID	Revenue	Year	AverageRevenue	TotalRevenue
1	1	10030	1998	25007	100030
2	1	20000	1999	25007	100030
3	1	40000	2000	25007	100030
4	1	30000	2001	25007	100030
5	2	20000	1998	37500	150000
6	2	60000	1999	37500	150000
7	2	40000	2000	37500	150000
8	2	30000	2001	37500	150000
9	3	40000	1998	55000	220000
10	3	50000	1999	55000	220000
11	3	60000	2000	55000	220000
12	3	70000	2001	55000	220000

مطابق شکل، جدول براساس فیلد DepartmentID به سه Partition تقسیم شده است، و عملیات میانگین و جمع روی فیلد Revenue انجام شده است و عملیات Sort روی هرگروه بطور مستقل انجام گرفته است. چنین کاری را نمی‌توانستیم بوسیله Group By انجام دهیم.

مثال دوم : نحوه استفاده از ROWS PRECEDING،در این مثال قصد داریم عملیات جمع را روی فیلدRevenue انجام دهیم. بطوریکه جمع هر مقدار برابر است با سه مقدار قبلی + مقدار جاری:

لطفا رکوردهای زیر را به جدول فوق درج نمایید:

```
insert into REVENUE
 values(1,90000,2002),(2,20000,2002),(3,80000,2002),
 (1,10300,2003),(2,1000,2003), (3,90000,2003),
 (1,10000,2004),(2,10000,2004),(3,10000,2004),
 (1,20000,2005),(2,20000,2005),(3,20000,2005),
 (1,40000,2006),(2,30000,2006),(3,30000,2006),
 (1,70000,2007),(2,40000,2007),(3,40000,2007),
 (1,50000,2008),(2,50000,2008),(3,50000,2008),
 (1,20000,2009),(2,60000,2009),(3,60000,2009),
 (1,30000,2010),(2,70000,2010),(3,70000,2010),
 (1,80000,2011),(2,80000,2011),(3,80000,2011),
 (1,10000,2012),(2,90000,2012),(3,90000,2012)
```

سپس Script زیر را اجرا می‌نماییم:

```
select Year, DepartmentID, Revenue,
sum(Revenue) OVER (PARTITION by DepartmentID ORDER BY [YEAR]
             ROWS BETWEEN 3 PRECEDING AND CURRENT ROW) as Prev3
From REVENUE order by departmentID, year;
```

خروجی :

	Year	DepartmentID	Revenue	Prev3
1	1998	1	10030	10030
2	1999	1	20000	30030
3	2000	1	40000	70030
4	2001	1	30000	100030
5	2002	1	90000	180000
6	2003	1	10300	170300
7	2004	1	10000	140300
8	2005	1	20000	130300
9	2006	1	40000	80300

در Script بالا، جدول را براساس فیلد DepartmentID گروه بندی می‌کنیم، که سه گروه ایجاد می‌شود، هر گروه را بطور مستقل، روی فیلد Year بصورت صعودی مرتب می‌نماییم. حال برای آنکه بتوانیم سیاست جمع، روی فیلد Revenue، را پیاده سازی نماییم ، قطعه کد زیر را در Script بالا اضافه کردیم.

```
ROWS BETWEEN 3 PRECEDING AND CURRENT ROW) as Prev3
```

برای شرح چگونگی استفاده از PRECEDING،فقط به شرح گروه اول بسنده می‌کنیم. مقدار جمع فیلد Revenue سطر اول، که قبل از آن سطری وجود ندارد، برابر است با مقدار خود، یعنی 10030، مقدار جمع فیلد Revenue سطر دوم برابر است با حاصل جمع مقدار فیلدRevenue سطر اول و دوم ، یعنی 30030 . این روند تا سطر چهار ادامه دارد، اما برای بدست آوردن مقدار جمع فیلدRevenue سطر پنجم، مقدار جمع فیلد Revenue سطر دوم،سوم،چهارم و پنجم در نظر گرفته می‌شود، و مقدار فیلدRevenue سطر اول در حاصل جمع در نظر گرفته نمی‌شود،بنابراین مقدار جمع فیلد Revenue سطر پنجم برابر است با 180000. در صورت مسئله گفته بودیم، مقدار جمع فیلد Revenue هر سطر جاری برابر است با حاصل جمع مقدارسطر جاری و مقادیر سه سطر ماقبل خود.

مثال سوم: نحوه استفاده از ROWS FOLLOWING ، این مثال عکس مثال دوم است، یعنی حاصل جمع مقدار فیلد Revenue هر سطر برابر است با حاصل جمع سطر جاری با سه سطر بعد از خود. بنابراین Script زیر را اجرا نمایید:

```
select Year, DepartmentID, Revenue,
 sum(Revenue) OVER (PARTITION by DepartmentID ORDER BY [YEAR]
              ROWS BETWEEN CURRENT ROW AND 3 FOLLOWING) as Next3
From REVENUE order by departmentID, year;
```

خروجی :

	Year	DepartmentID	Revenue	Next3
1	1998	1	10030	100030
2	1999	1	20000	180000
3	2000	1	40000	170300
4	2001	1	30000	140300
5	2002	1	90000	130300
6	2003	1	10300	80300
7	2004	1	10000	140000
8	2005	1	20000	180000

مطابق شکل مقدار جمع فیلد اول برابراست با حاصل جمع مقدار سطر جاری و سه سطر بعد از آن.

نکته ای که در مثالهای دوم و سوم،می بایست به آن توجه نمود، این است که در زمان استفاده از Row یا Range ، استفاده از Order by در Partition الزامی است، در غیر این صورت با خطا مواجه می‌شوید.

```
select Year, DepartmentID, Revenue,
  sum(Revenue) OVER (PARTITION by DepartmentID --ORDER BY [YEAR]
                     ROWS BETWEEN CURRENT ROW AND 3 FOLLOWING) as Next3
From REVENUE order by departmentID, year;
```

Messages

```
Msg 10756, Level 15, State 1, Line 2
Window frame with ROWS or RANGE must have an ORDER BY clause.
```

نحوه استفاده از UNBOUNDED PRECEDING ، این امکان در T-SQL Server 2012 افزوده شده است.

مثال چهار: در این مثال می‌خواهیم کمترین سود بدست آمده در چند سال را بدست آوریم:

ابتدا Script زیر را اجرا نمایید:

```
select Year, DepartmentID, Revenue,
       min(Revenue) OVER (PARTITION by DepartmentID ORDER BY [YEAR]
                          ROWS UNBOUNDED PRECEDING) as MinRevenueToDate
From REVENUE order by departmentID, year;
```

خروجی:

Results | Messages

	Year	DepartmentID	Revenue	MinRevenueToDate
1	1998	1	10030	10030
2	1999	1	20000	10030
3	2000	1	40000	10030
4	2001	1	30000	10030
5	2002	1	90000	10030
6	2004	1	10000	10000
7	2005	1	20000	10000
8	2006	1	40000	10000
9	2007	1	70000	10000
10	2008	1	50000	10000
11	2009	1	20000	10000
12	2010	1	30000	10000
13	2011	1	80000	10000
14	2012	1	10000	10000
15	1998	2	20000	20000
16	1999	2	60000	20000

طبق تعریف UNBOUNDED PRECEDING اولین سطر هر Partition را مشخص می‌نماید، و چون از PRECEDING استفاده کرده ایم، بنابراین مقایسه همیشه بین سطر جاری و سطرهای قبل از آن انجام می‌پذیرد. بنابراین خواهیم داشت، کمترین مقدار فیلد Revenue در سطر اول، برابر با مقدار خود می‌باشد، چون هیچ سطری ماقبل از آن وجود ندارد. در سطر دوم مقایسه کمترین مقدار، بین 20000 و 10030 انجام می‌گیرد، که برابر است با 10030، در سطر سوم، مقایسه بین مقادیر سطر اول،دوم و سطر سوم صورت می‌گیرد، یعنی کمترین مقدار بین 20000،40000 و 10030، بنابراین کمترین مقدار سطر سوم برابر است با 10030.
به بیان ساده‌تر برای بدست آوردن کمترین مقدار هر سطر، مقدار سطر جاری با مقادیر همه سطرهای ماقبل خود مقایسه می‌گردد.
برای بدست آوردن کمترین مقدار در سطر ششم، مقایسه بین مقادیر سطرهای اول،دوم،سوم،چهارم،پنجم و ششم صورت می‌گیرد که عدد 10000 بدست می‌آید و الی آخر...
نکته: اگر در Over Clause شرط Order by را اعمال نماییم، اما از Row یا Range استفاده نکنیم، SQL Server بصورت پیش فرض از قالب زیر استفاده می‌نماید:

```
RANGE UNBOUNDED PRECEDING AND CURRENT ROW
```

برای روشن‌تر شدن مطلب فوق مثالی می‌زنیم:
ابتدا Script زیر را اجرا نمایید، که شامل ایجاد یک جدول و درج چند رکورد در آن می‌باشد:

```
CREATE TABLE Employees (
    EmployeeId INT IDENTITY PRIMARY KEY,
    Name VARCHAR(50),
    HireDate DATE NOT NULL,
    Salary INT NOT NULL
)
GO
INSERT INTO Employees (Name, HireDate, Salary)
VALUES
```

```
    ('Alice', '2011-01-01', 20000),
    ('Brent', '2011-01-15', 19000),
    ('Carlos', '2011-02-01', 22000),
    ('Donna', '2011-03-01', 25000),
    ('Evan', '2011-04-01', 18500)
GO
```

سپس Script زیر را اجرا نمایید:

```
SELECT
    Name,
    Salary,
    AVG(Salary) OVER(ORDER BY HireDate) AS avgSalary
FROM Employees
GO
```

خروجی :

	Name	Salary	avgSalary
1	Alice	20000	20000
2	Brent	19000	19500
3	Carlos	22000	20333
4	Donna	25000	21500
5	Evan	18500	20900

حال اگر Script زیر را نیز اجرا نمایید، خروجی آن مطابق شکل بالا خواهد بود:

```
SELECT
    Name,
    Salary,
    AVG(Salary) OVER(ORDER BY HireDate
                  RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS avgSalary
FROM Employees
GO
```

توضیح درباره Script بالا، در این روش برای بدست آوردن میانگین هر سطر، مقدار سطر جاری با مقادیر سطرهای ماقبل خود جمع و تقسیم بر تعداد سطر می‌شود.

سطر دوم 20000 + 19000 تقسیم بر دو برابر است با 19500

میانگین سطر پنجم، حاصل جمع فیلد Salary همه مقادیر سطرها تقسیم بر 5

*** اگر بخواهید بوسیله Over Clause ، میانگین همه سطرها یکسان باشد می‌توانید از Script زیر استفاده نمایید:

```
SELECT
    Name,
    Salary,
    AVG(Salary) OVER(ORDER BY HireDate
                        RANGE
                        BETWEEN UNBOUNDED PRECEDING
                        AND UNBOUNDED FOLLOWING
                    ) AS avgSalary
FROM Employees
GO
```

خروجی :

	Name	Salary	avgSalary
1	Alice	20000	20900
2	Brent	19000	20900
3	Carlos	22000	20900
4	Donna	25000	20900
5	Evan	18500	20900

منظور از ROWS BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING یعنی در محاسبه میانگین برای هر سطر تمامی مقادیر سطرهای دیگر در نظر گرفته شود.

پایان بخش اول

امیدوارم مفید واقع شده باشد.

نظرات خوانندگان

نویسنده: ناصر
تاریخ: ۱۳۹۱/۰۹/۱۸ ۰:۲۳

بسیار عالی. فقط خواستم تشکری کرده باشم :)

نویسنده: reza
تاریخ: ۱۳۹۱/۰۹/۱۸ ۰:۳۳

SQLQuery2.sql - (l...Er-PC\na3Er (51))* | SQLQuery1.sql - (l...Er-PC\na3Er (54))* | NA3ER-PC.learn - dbo.REVENUE

```
select Year, DepartmentID, Revenue,
 sum(Revenue) OVER (PARTITION by DepartmentID ORDER BY [YEAR]
              ROWS BETWEEN 3 PRECEDING AND CURRENT ROW) as Prev3
From REVENUE order by departmentID, year;
```

Messages

```
Msg 102, Level 15, State 1, Line 3
Incorrect syntax near 'ROWS'.
```

علت این خطا چیست ؟ آیا این دستور مال 2012 هستش ؟ جون من تو 2008 تست میکنم

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۱۸ ۶:۸

سلام

Syntax ارائه شده Over Clause در SQL Server 2008 شامل دو پارامتر است :

```
Ranking Window Functions
< OVER_CLAUSE > :: =
    OVER ( [ PARTITION BY value_expression , ... [ n ] ]
           <ORDER BY_Clause> )

Aggregate Window Functions
< OVER_CLAUSE > :: =
    OVER ( [ PARTITION BY value_expression , ... [ n ] ] )
```

در SQL Server 2012 پارامتر <ROW or RANGE clause> به Syntax فوق افزوده شد. بنابراین، Query هایی که از Row یا Range

استفاده کرده اند، در SQL Server 2008 با خطا مواجه می‌شوند.

نویسنده: معتمدی
تاریخ: ۱۳۹۱/۰۹/۱۸ ۱۲:۵۹

با سلام

آیا استفاده از این queryها برای محیط‌های transactional هم مناسب است؟ یا بیشتر در databaseهای آماری جهت تهیه گزارشات کاربرد دارد؟

از نظر زمان و هزینه‌ی اجرا و نیز تخصیص منابع سرور به آنها می پرسم.

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۱۸ ۱۴:۵۷

سلام
سرعت و قابلیت اجرایی Over Clause به مراتب از Group by بهتر است. بطوریکه اگر یک عملیات یکسان را،بطور جداگانه،هم با Over Clause و هم با Group By انجام دهید. و در Execution Plan مشاهده نمایید، تفاوت را حس خواهید نمود. سایت زیر یک مثال ساده در این رابطه قرار داده است: Windowing functions - Underappreciated

در مورد اینکه برای محیط های Transactional مناسب است یا نه، عوامل زیادی در آن دخیل است و بسته به حجم داده ای مورد انتظار شما در خروجی دارد،بطور مثال اگر بخواهید یک گزارش 400 صفحه ای ایجاد نمایید، بطور حتم در چنین محیط هایی هیچ Scriptی مناسب، نیست، اما بطور قاطع می‌توان گفت که Window Functionها از کارایی بسیار خوبی برخوردار هستند، و انجام عملیات‌های پیچیده محاسباتی را برای ما آسانتر نموده اند.

نویسنده: فاطمه
تاریخ: ۱۳۹۱/۱۰/۲۳ ۱۷:۲۲

بسیار عالی بود
با تشکر

نویسنده: zarei
تاریخ: ۱۳۹۲/۰۲/۲۱ ۲۲:۱۴

سلام
ممنون از آموزش مفیدتون . سوال من اینه که اگه بخواهیم رکوردهای تکراری حذف بشن باید چیکار کنیم ؟ مثلا من میخام مجموع مبلغ بدهکار برای یک کد کل و در یک سند را در یک ردیف و همین مورد برای مجموع مبلغ بستانکار را نیز در یک رکورد یا ردیف دیگر بدهد . در صورتی که اگر از توابع Over ()Sum استفاده کنیم به ازای هر کد کل درآن سند یک رکورد در خروجی داریم (چه بدهکار و چه بستانکار)

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۲/۰۲/۲۲ ۸:۱۹

سلام
اگر سئوال شما را درست متوجه شده باشم، فکر می‌کنم، می‌بایست از Pivot استفاده نمایید،مقاله زیر می‌تواند در درک Pivot به شما کمک نماید. Pivot

موفق باشید.

نویسنده: محمد شهریاری
تاریخ: ۱۳۹۳/۰۲/۲۱ ۹:۰

با سلام
آیا امکان استفاده از scaler functionها هم هست ؟

ممنون

نویسنده: فرهاد
تاریخ: ۱۳۹۳/۰۲/۲۲ ۱۵:۱۶

سلام
معمولا از توابع Aggregate Functions می‌توان در Window Function استفاده نمود: ^

عنوان: **گرفتن خروجی XML از جداول در SQL Server 2012**
نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۰۹/۱۹ ۰:۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, xml, SQL Server 2012, T-SQL

فرض کنید که می‌خواهیم خروجی از جدول خود را به صورت XML نمایش یا از طریق وب سرویس در برنامه مان استفاده نماییم. اولین راهی که به ذهنمان می‌رسد خودمان رشته xml را با حلقه ای ایجاد نماید یا استفاده از فضای نام System.Xml و کلاس‌های نوشته شده برای این کار . اما خود Sql Server امکانات ویژه ای برای کار با ساختار xml مهیا نموده که براحتی می‌توانید خروجی xml از داده هایتان ایجاد نمایید.

برای این کار از عبارت For XML در Select می‌توان استفاده نمود. برای مثال برای بدست آوردن ساختار ساده از For Xml Auto استفاده نمایید

```
SELECT BusinessEntityID, PersonType, Title, FirstName, MiddleName, LastName
FROM Person
WHERE BusinessEntityID = 10001
FOR XML AUTO
```

که خروجی بصورت **node attribute** زیر می‌باشد:

```
<Person.Person BusinessEntityID="10001" PersonType="IN" FirstName="Carolyn" LastName="Alonso" />
```

اما اگر بخواهیم خروجی به صورت node Elements باشد کافیست از پارامتر Elements استفاده نمایید

```
SELECT BusinessEntityID, PersonType, Title, FirstName, MiddleName, LastName
FROM Person
WHERE BusinessEntityID = 10001
FOR XML AUTO, ELEMENTS
```

خروجی بصورت زیر می‌باشد:

```
<Person.Person>
  <BusinessEntityID>10001</BusinessEntityID>
  <PersonType>IN</PersonType>
  <FirstName>Carolyn</FirstName>
  <LastName>Alonso</LastName>
</Person.Person>
```

اگر بخواهیم node attributes و node elements با هم ترکیب کنیم بصورت زیر عمل می‌کنیم:

```
SELECT BusinessEntityID AS '@ID', PersonType, Title,  FirstName, MiddleName, LastName
FROM Person
WHERE BusinessEntityID = 10001
FOR XML ELEMENTS
```

خروجی بصورت زیر است:

```
<Person ID="10001">
  <PersonType>IN</PersonType>
  <FirstName>Carolyn</FirstName>
  <LastName>Alonso</LastName>
</Person>
```

حال می‌خواهیم همه nodeها را یک node ریشه قرار دهیم برای این کار از پارامتر ROOT در کنار AUTO به صورت زیر استفاده نمایید:

```
SELECT *
FROM Person
WHERE BusinessEntityID = 15291
FOR XML AUTO , ROOT('Persons')
```

اما اگر بخواهیم نام جدول را با نام دلخواه خود تغییر دهیم از پارامتر PATH به جای AUTO به صورت زیر استفاده نمایید:

```
SELECT *
FROM Person
WHERE BusinessEntityID = 15291
FOR XML PATH('P') , ROOT('Persons')
```

نظرات خوانندگان

نویسنده: احسان میرسعیدی
تاریخ: ۱۳۹۱/۰۹/۱۹ ۰:۲۶

بسیار تکنیک کارامدی بود. واقعا متشکرم

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۱۹ ۷:۲۹

سلام
مرسی از مطلب مفید تان

نویسنده: بهراد
تاریخ: ۱۳۹۱/۰۹/۱۹ ۱۱:۸

بسیار عالی بود ، خیلی برام مفید واقع شد
راه مشابه ای برای خروجی Json نیست؟
ممنون

نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۰۹/۱۹ ۱۱:۵۷

ممنون .هنوز به صورت native نه اما از تیم microsoft در این لینک خواسته شده که For Json را هم اضافه کند.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۹/۱۹ ۱۱:۵۹

« استفاده از JSON در SQL Server »

نویسنده: مهدی ناظم السادات
تاریخ: ۱۳۹۲/۰۹/۱۷ ۱۸:۵۸

حالا دستور Insert نداریم که بشه یه فایل xml رو مثل فایل backup روی دیتابیس restore کنیم؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۹/۱۸ ۰:۵۹

- می‌تونی با کدنویسی اینکار رو انجام بدی:

```
var reportData = new DataSet();
reportData.ReadXml("yourfile.xml");
var connection = new SqlConnection("DB ConnectionString");
var sbc = new SqlBulkCopy(connection);
sbc.DestinationTableName = "yourXMLTable";
```

- یا می‌تونی از import و export خود SQL Server استفاده کنی.
- و یا از OPENXML میشه استفاده کرد:

```
INSERT Customers
SELECT *
FROM OPENXML ...
```

نویسنده: ناصرنیازی
تاریخ: ۱۳۹۲/۱۱/۲۹ ۲۱:۵۱

تشکر فراوان از این مطلب فوق العاده کاربردی. یه نکته کوچک به ذهنم رسید. سومین قطعه کدی که نوشتید در جدول من کار نکرد به نظرم اومد که شاید اینچنین بوده باشه :
کد شما

```
FOR XML ELEMENTS
```

کدی که من مد نظرم هست و در جدول من کار می‌کنه :

```
for xml auto, ELEMENTS
```

عنوان: آشنایی با Window Function ها در SQL Server بخش دوم
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۱۹ ۹:۳۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, SQL Server 2012, T-SQL

قبل از مطالعه این بخش لطفا آشنایی با Window Functionها در SQL Server بخش اول را مطالعه نمایید.

در بخش اول ،در مورد Syntax مربوط به Over Clause صحبت کردیم، و برای درک استفاده از Over Clause، مثالهایی را بررسی نمودیم، در این بخش نیز،به تفاوت Row Clause و Range Clause می پردازیم.

مثال: با ایجاد یک Script،عملیات جمع روی یک فیلد خاص، بوسیله Row Clause و Range Clause انجام می‌دهیم. تا تفاوت آنها را درک نماییم.

در ادامه Script زیر را اجرا نمایید:

```
DECLARE @Test TABLE
(
RowID INT IDENTITY,
FName VARCHAR(20),
Salary SMALLINT
);
INSERT INTO @Test (FName, Salary)
VALUES ('George', 800),
('Sam', 950),
('Diane', 1100),
('Nicholas', 1250),
('Samuel', 1250),
('Patricia', 1300),
('Brian', 3000),
('Thomas', 1600),
('Fran', 2450),
('Debbie', 2850),
('Mark', 2975),
('James', 3000),
('Cynthia', 3000),
('Christopher', 5000);

SELECT RowID,FName,Salary,
       SumByRows = SUM(Salary) OVER (ORDER BY Salary ROWS UNBOUNDED PRECEDING),
   SumByRange = SUM(Salary) OVER (ORDER BY Salary RANGE UNBOUNDED PRECEDING)
FROM @Test
ORDER BY RowID;
```

خروجی بصورت زیر خواهد بود:

	RowID	FName	Salary	SumByRows	SumByRange
1	1	George	800	800	800
2	2	Sam	950	1750	1750
3	3	Diane	1100	2850	2850
4	4	Nicholas	1250	4100	5350
5	5	Samuel	1250	5350	5350
6	6	Patricia	1300	6650	6650
7	7	Brian	1500	8150	8150
8	8	Thomas	1600	9750	9750
9	9	Fran	2450	12200	12200
10	10	Debbie	2850	15050	15050
11	11	Mark	2975	18025	18025
12	12	James	3000	21025	24025
13	13	Cynthia	3000	24025	24025
14	14	Christopher	5000	29025	29025

با مشاهده شکل بالا، به وضوح می‌توان تفاوت Row و Range را تشخیص داد. در Script بالا از UNBOUNDED PRECEDING استفاده کردیم ، و مفهوم قالب آن به شرح ذیل می‌باشد:

مقدار فیلد Salary سطر جاری = جمع مقادیر فیلد Salary همه سطرهای ماقبل،سطر جاری + مقدار فیلد Salary سطر جاری

Row Clause بصورت فیزیکی به سطرها می‌نگرد و قالب بیان شده در Script را،روی تمامی سطرها،نسبت به جایگاه آنها در جدول، به ترتیب اعمال می‌نماید.و در شکل نیز قابل مشاهده می باشد، یعنی به چیدمان سطرها در خروجی که بصورت فیزیکی نمایش داده شده است، توجه می کند، و حاصل جمع هر سطر برابر است با حاصل جمع سطرهای ماقبل + سطر جاری

اما Range Clause:به چیدمان فیزیکی سطرها توجه نمی‌کند، بلکه بصورت منطقی به مقدار فیلد Salary سطرها توجه می‌نماید، یعنی مقادیری که در یک محدوده(Range) قرار دارند، حاصل جمع آنها،یکی است.

مقدار فیلد Salary سطر چهار و پنج برابر است با 1250 بنابراین حاصل جمع آنها برابر هم می‌باشد. و بصورت زیر محاسبه می‌شود:

5350= 1250 + 1250 + 1100 + 950 + 800

روش بیان شده، در مورد سطرهای 12 و 13 نیز صادق است.

امیدوارم با مثالهایی که در بخش اول و بخش دوم بررسی نمودیم، روش استفاده از Over Clause را درک کرده باشیم.

Window Functionها را به چهار بخش تقسیم بندی شده اند، که به شرح ذیل می‌باشد:

1- Ranking functions (توابع رتبه بندی)، که بررسی نمودیم.

2- NEXT VALUE FOR ، که در بحث ایجاد Sequence آن را بررسی نمودیم.

3- Aggregate Functions (توابع جمعی)، اکثرا با اینگونه توابع آشنا هستیم.

4- Analytic Functions (توابع تحلیلی) که در بخش بعدی آن را بررسی می‌نماییم.

یکی از منابع بسیار مفید در مورد Window Function ها کتاب Microsoft SQL Server 2012 High-Performance T-SQL Using Window Functions ، می باشد،که بطور کامل به Window Functionها اختصاص دارد و تکنیک‌های بسیار مفیدی را بیان می‌کند. مطالعه آن به علاقمندان، پیشنهاد می‌گردد.

موفق باشید.

نظرات خوانندگان

نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۹/۱۹ ۱۲:۲۱

دوست عزیز ممنون...
من قسمت Range و Row رو که شما توضیح دادی درست متوجه نشدم سرچی که زدم متوجه شدم این قابلیت تقریبا شبیه قسمت WITH TIES تو Select هست .برای مثال اگه بخواهیم 3 شاگرد برتر کلاس رو کوئری بزنیم اگه تو کلاس 3 نفر معدل 18 داشته باشن(با توجه به اینکه یک معدل 20 و 19 داریم) 2 نفر از شاگردها که معدل 18 دارن تو این کوئری نمیان(TOP 3) و...
برداشت منم ازRange اینه که بواسط ی برابر بودن تاریخ‌ها این 2 مقدار به هم گره خوردن و هنگام محاسبه مقدار یکسانی را تولید میکنن.

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۱۹ ۱۴:۲۱

سلام
اگر سئوال شما رو درست متوجه شده باشم،با یک مثال مفهوم Range رو بررسی می‌کنیم:

```
CREATE TABLE #Transactions
(
AccountId INTEGER,
TranDate DATE,
TranAmt NUMERIC(8, 2)
);
INSERT INTO #Transactions
SELECT *
FROM ( VALUES ( 1, '2011-01-15', 50),( 1, '2011-01-17', 500),( 1, '2011-01-17', 500),
  ( 1, '2011-01-16', 500),( 1, '2011-01-24', 75),( 1, '2011-01-26', 125),
  ( 1, '2011-02-28', 500),( 2, '2011-01-01', 500),( 2, '2011-01-15', 50),
              ( 2, '2011-01-22', 25),( 2, '2011-01-23', 125),( 2, '2011-01-26', 200),
              ( 2, '2011-01-29', 250),( 3, '2011-01-01', 500),( 3, '2011-01-15', 50 ),
              ( 3, '2011-01-22', 5000),( 3, '2011-01-25', 550),( 3, '2011-01-27', 95 ),
              ( 3, '2011-01-30', 2500)
) dt (AccountId, TranDate, TranAmt);
```

روی جدول فوق دو نوع Script اجرا می‌کنیم، مثال اول، براساس AccountID جدول را گروه بندی می‌نماییم. سپس هر گروه را براساس تاریخ Sort می‌کنیم، و در هر گروه مقدار Sum آن را بدست می‌آوریم:

```
SELECT
    AccountId,
    TranDate,
  TranAmt,
    Sum(TranAmt) OVER(partition by Accountid ORDER BY TranDate RANGE UNBOUNDED PRECEDING) AS SumAmt
FROM  #Transactions
GO
```

خروجی :

	AccountId	TranDate	TranAmt	SumAmt
1	1	2011-01-15	50.00	50.00
2	1	2011-01-16	500.00	550.00
3	1	2011-01-17	500.00	1550.00
4	1	2011-01-17	500.00	1550.00
5	1	2011-01-24	75.00	1625.00
6	1	2011-01-26	125.00	1750.00
7	1	2011-02-28	500.00	2250.00

مطابق شکل Sort براساس TranDate است، که چهار مقدار 500 در سه بازه تاریخی دیده می‌شود، حال محاسبه جمع هر سطر بصورت زیر است:

سطر دوم با وجود اینکه مقدار آن 500 است و در بازه تاریخی 2011-01-16 قرار دارد: مقدار آن برابر است با 500 + 50=550

سطر سوم و چهارم که در بازه تاریخی 2011-01-17 می‌باشد(به عبارتی در یک محدوده می‌باشند): برابر است با :

500+500+500+50=1550

در اینجا چیزی حذف نشده، حاصل جمع سطر سوم و چهارم ، چون در یک محدوده (Range) می‌باشد، برابر است با حاصل جمع سطرهای ما قبل یعنی سطر اول و دوم (500+50=550) + حاصل جمع تمامی سطرهای آن محدوده(Range)، یعنی سطر سوم و چهارم (500+500=1000)

مثال دیگر، در این حالت Sort روی فیلد TranAMT انجام می‌شود، و جدول همچنان روی فیلد Accountid گروه بندی می‌شود، بنابراین خواهیم داشت:

```
SELECT
    AccountId,
    TranDate,
  TranAmt,
    Sum(TranAmt) OVER(partition by Accountid ORDER BY TranAmt RANGE UNBOUNDED PRECEDING) AS SumAmt
FROM  #Transactions
GO
```

خروجی :

	AccountId	TranDate	TranAmt	SumAmt
1	1	2011-01-15	50.00	50.00
2	1	2011-01-24	75.00	125.00
3	1	2011-01-26	125.00	250.00
4	1	2011-02-28	500.00	2250.00
5	1	2011-01-17	500.00	2250.00
6	1	2011-01-17	500.00	2250.00
7	1	2011-01-16	500.00	2250.00
8	2	2011-01-22	25.00	25.00
9	2	2011-01-15	50.00	75.00
10	2	2011-01-23	125.00	200.00

در شکل، مقدار جمع هیچ سطری حذف نشده است، و Top ی هم در کار نیست.
حال اگر مثال فوق را روی میانگین در نظر بگیرید، باز هم تمام مقادیر، در محاسبه میانگین تاثیر گذار میباشند.

نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۹/۱۹ ۱۴:۵۹

ممنون...
« سطرهای فیزیکی » و «سطرهای منطقی» که شما گفتید برا من گیج کننده بود من با مثال Top ی که زدم فرق رو متوجه شدم
گفتم عنوان کنم دیگران هم استفاده کنن.البته این پست شما خیلی بهتر موضوع رو توضیح داد.

نویسنده: محمد سلم ابادی
تاریخ: ۱۳۹۱/۱۰/۳۰ ۱۲:۱۱

سلام،
ممنون از مطالب مفیدتون.
آیا دو دستور زیر با هم یکسان هستند یا خیر؟
range unbounded preceding
range between unbounded preceding and current row

و کوئری اولتون باید مساله running total باشه، که به سادگی توسط Over Clause حل شده.
کوئری زیر روشی بوده که قبل از نسخه 2012 برای حل اینگونه مسائل مورد استفاده قرار میگرفته

```
SELECT AccountId,
       TranDate,
       TranAmt,
       D.sumAmt AS older_method
       Sum(TranAmt) OVER(partition by Accountid
                         ORDER BY TranDate
                         RANGE UNBOUNDED PRECEDING) AS SumAmt
FROM  #Transactions AS T1
CROSS APPLY (SELECT SUM(T2.TranAmt)
   FROM #Transactions AS T2
  WHERE T2.AccountId = T1.AccountId
    AND T2.TranDate <= T1.TranDate) AS D(SumAmt)
ORDER BY T1.AccountId, T1.TranDate;
```

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۱۰/۳۰ ۱۳:۲۵

سلام
در جواب سؤال شما باید بگویم هر دو دستور یکی میباشند و هر دو از اولین سطر تا سطر جاری را در نظر می‌گیرند.

نویسنده: zarei
تاریخ: ۱۳۹۲/۰۲/۲۲ ۱۹:۳

سلام
من یه کوئری توسط Over ()Sum() .. نوشتم که تو در تو هست که ترتیب جمع دستور بیرونی برام مهمه .

```
;WITH cteBed ([Counter], id_doc , [Year] ,id_Total , date_duc ,Number_Temp , number_fix , sumbed ,
sumbes , row_no ) AS (
SELECT [Counter], d.id_doc , d.[Year] ,r.id_Total , d.date_duc ,d.Number_Temp ,d.number_fix ,
SUM( r.Mablagh_bed) OVER(PARTITION BY d.[Year] ,r.id_Total , d.Number_Temp) AS sumbed ,
 sumbes= 0,
```

```
ROW_NUMBER() OVER (PARTITION BY d.[Year] ,r.id_Total , d.date_duc , d.Number_Temp , d.number_fix  ORDER
BY  d.date_duc )AS  row_no
FROM tbl_Records r JOIN tbl_Documents d ON d.id_doc = r.id_doc  ) ,

 cteBes ([Counter], id_doc , [Year] ,id_Total , date_duc ,Number_Temp , number_fix , sumbed , sumbes  ,
row_no) AS (
SELECT   [Counter], d.id_doc , d.[Year] ,r.id_Total , d.date_duc ,d.Number_Temp ,d.number_fix , sumbed
= 0 ,
SUM( r.Mablagh_bes ) OVER(PARTITION BY d.[Year] ,r.id_Total , d.Number_Temp ) AS sumbes,
ROW_NUMBER() OVER (PARTITION BY d.[Year] ,r.id_Total ,  d.date_duc ,d.Number_Temp , d.number_fix ORDER
BY  d.date_duc )AS row_no
FROM tbl_Records r JOIN tbl_Documents d ON d.id_doc = r.id_doc )

SELECT [Counter], id_doc , [Year] ,id_Total , date_duc ,Number_Temp , number_fix , sumbed , sumbes ,
amountBed ,amountBes
,SUM(amountBed)OVER(  ORDER BY [Year] ,id_Total , date_duc , number_Temp, number_Fix ROWS BETWEEN
UNBOUNDED PRECEDING AND CURRENT ROW ) AS bed
,SUM(amountBes)OVER(  ORDER BY [Year] ,id_Total , date_duc , number_Temp, number_Fix ROWS BETWEEN
UNBOUNDED PRECEDING AND CURRENT ROW ) AS bes
FROM (
SELECT [Counter], id_doc , [Year] ,id_Total , date_duc ,Number_Temp , number_fix , sumbed , sumbes ,
amountBed = CASE WHEN id_Total  LIKE '1%' OR  Id_Total LIKE '2%' OR  Id_Total LIKE '7%' OR  Id_Total
LIKE '8%' THEN (tt.sumbed-tt.sumbes) ELSE 0 END ,
amountBes=CASE WHEN Id_Total LIKE '3%' OR Id_Total LIKE '4%' OR Id_Total LIKE '5%' OR Id_Total LIKE
'6%' OR Id_Total LIKE '9%' THEN (tt.sumbes-tt.sumbed)ELSE 0 END ,
ROW_NUMBER() OVER (PARTITION BY [Year] ,id_Total , date_duc , Number_Temp , number_fix  ORDER BY
date_duc )AS  row_no
FROM (
SELECT * FROM cteBed cb WHERE cb.row_no = 1
UNION ALL
SELECT * FROM cteBes cb WHERE cb.row_no = 1
) AS tt ([Counter], id_doc , [Year] ,id_Total , date_duc ,Number_Temp , number_fix , sumbed ,
sumbes,row_no ) WHERE not(sumbed = 0 AND sumbes = 0)
) AS rr
```

اگه تو یه دستور Select از Row_Number استفاده کرده باشم ، اول خروجی رو بدست میاره بعد خروجی رو بر حسب نوع مرتب سازی مربوط به Row_Number مرتب میکنه ؟ و دیگه اینکه خروجی دستور اول که شامل Row_Number هست بعد از مرتب شدن به همون صورت به دست دستور دوم (یا همون Select بیرونی) میرسه یا باز باید روی اون نیز مرتب سازی انجام بدم ؟ اصلا جای ستونی که مربوط به Row_Number هست اول یا آخر فرق میکنه ؟

اینارو به این خاطر پرسیدم ، چون هر بار داده هام جواب متفاوتی میداد و نتونستم تشخیص بدم . ممنون

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۲/۰۲/۲۵ ۸:۱۴

سلام

جواب سوال اول: در Syntax تابع Row_Number عملیات order by اجباری است، بنابراین عملیات سورت در ابتدا انجام می‌شود و سپس Row_Number (اعداد ترتیبی) روی رکوردها اعمال می‌گردد.

در سایت مایکروسافت به خوبی اشاره شده است که هیچ تضمینی وجود ندارد، خروجی یک Query با استفاده از Row_number در هر بار اجرا، با اجرای قبلی یکی باشد مگر آنکه موارد زیر را رعایت کرده باشید:

1- مقادیر ستونی که برای قسمت Partition در نظر گرفته اید، منحصربفرد باشد.

2- مقادیری که برای قسمت Order by در نظر گرفته اید منحصربفرد باشد.

3- ترکیب مقادیر Partition و Order by نیز مقدار منحصربفردی را ایجاد نماید.

جواب سوال دوم: جای ستون Row_number در زمان نمایش اهمیتی ندارد.

پیشنهاد دوستانه:

1- تاجایی که امکان دارد از OR در Queryهای خود استفاده ننمایید، باعث افزایش زمان اجرای Query شما می‌شود و هزینه بالایی دارد.

2- از Like نیز در نوشتن Queryهای خود اجتناب کنید.

برای اطلاعات بیشتر در مورد Row_Number به آدرس زیر مراجعه نمایید: Row_Number()

موفق باشید.

نویسنده: zarei
تاریخ: ۱۳۹۲/۰۲/۲۷ ۱:۳۸

ممنون از راهنماییتون . ولی برای رسیدن به پاسخ راه دیگه ای به ذهنم نرسید (استفاده از OR و Like)
میشه خواهش کنم راه جایگزین رو بهم بگید ؟ اموزشی در خصوص بررسی مفهومی Planهای تولیدی SQL سراغ دارید ؟
دیگه اینکه از کجا میتونم به طور دقیق و مفهومی هزینه استفاده از دستورات SQL رو مثل OR و ... رو بخونم ؟
ممنون

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۲/۰۲/۲۷ ۲۲:۱۹

سلام
مقاله زیر به خوبی طرز استفاده از Execution Plan را آموزش می‌دهد. How to read SQL Server graphical query execution plans

دو کتاب زیر، جهت مطالعه و بهینه سازی در ایجاد Query مفید است: SQL Server 2012 T-SQL Recipes
SQL Server 2012 Query Performance Tuning

موفق باشید.

عنوان: **گرفتن خروجی JSON از جداول در SQL Server 2012**
نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۰۹/۱۹ ۱۳:۱۵
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, JSON, SQL Server 2012, T-SQL

در مطلب قبلی با استفاده از دستور For XML خروجی xml تولید کردیم اما با همین دستور می‌توان تا حدودی خروجی Json نیز تولید نمود. البته به صورت native هنور در sql server این امکان وجود ندارد که با رای دادن به این لینک از تیم ماکروسافت بخواهید که این امکان را در نسخه بعدی اضافه کند.
برای این کار یک جدول موقت ایجاد کرده و چند رکورد در آن درج می‌کنیم:

```
declare @t table(id int, name nvarchar(max), active bit)
insert @t values (1, 'Group 1', 1), (2, 'Group 2', 0)
```

حال با استفاده از همان for xml و پارامتر type که نوع خروجی xml را خودمان می‌توانیم تعیین نماییم و پارمتر Path این کار را بصورت زیر انجام می‌دهیم:

```
select '[' + STUFF((
        select
            ',{"id":' + cast(id as varchar(max))
            + ',"name":"' + name + '"'
            + ',"active":' + cast(active as varchar(max))
            +'}'

        from @t t1
        for xml path(''), type
    ).value('.', 'varchar(max)'), 1, 1, '') + ']'
```

توجه کنید در این جا از پارامتر path بدون نام استفاده شده است و از تابع STUFF برای در یک رشته در رشته دیگر استفاده شده است. خروجی در زیر آورده شده است:

```
[{"id":1,"name":"Group 1","active":1},{"id":2,"name":"Group 2","active":0}]
```

حالت پیشرفته‌تر آن است که بتوانیم یک join را بصورت فرزندان آن در json نمایش دهیم قطعه کد زیر را مشاهده فرمایید:

```
declare @group table(id int, name nvarchar(max), active bit)
insert @group values (1, 'Group 1', 1), (2, 'Group 2', 0)

declare @member table(id int, groupid int,name nvarchar(max))
insert @member values (1, 1,'Ali'), (2, 1,'Mojtaba'),(3,2,'Hamid')

select '[' + STUFF((
        select
            ',{"id":' + cast(g.id as varchar(max))
            + ',"name":"' + g.name + '"'
+ ',"members": { "children": [' +
(select + STUFF((
select
 ',{"id":' + cast(m.id as varchar(max))
+ ',"name":"' + m.name + '"}'
from @member m
where m.groupid = g.id
for xml path(''), type
 ).value('.', 'varchar(max)'), 1, 1, '')

 + ']}'
            + ',"active":' + cast(g.active as varchar(max))
            +'}')

        from @group g
        for xml path(''), type
    ).value('.', 'varchar(max)'), 1, 1, '') + ']'
```

خروجی json بصورت زیر است:

```
[{"id":1,"name":"Group 1","members":
     { "children": [{"id":1,"name":"Ali"},{"id":2,"name":"Mojtaba"}]}
,"active":1},
{"id":2,"name":"Group 2","members":
      { "children": [{"id":3,"name":"Hamid"}]}
,"active":0}]
```

حالت‌های خاص و پیشرفته‌تر را با امکانات t-sql خودتان می‌توانید به همین شکل تولید نمایید.

عنوان: **آشنایی با Window Function ها در SQL Server بخش سوم**
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۰۹/۲۴ ۱۹:۳۵
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, SQL Server 2012, T-SQL

آشنایی با Window Functionها در SQL Server بخش اول
آشنایی با Window Functionها در SQL Server بخش دوم

در این بخش به دو Function از Analytic Functionها (توابع تحلیلی)، یعنی Lead Function و LAG Function می پردازیم.
قبل از اینکه به توابع ذکرشده بپردازیم، باید عرض کنم، شرح عملکرد اینگونه توابع کمی مشکل می‌باشد، بنابراین با ذکر مثال و توضیح آنها،سعی می‌کنیم،قابلیت هریک را بررسی و درک نماییم.

Lead Function:

این فانکشن در SQL Server 2012 ارائه شده است، و امکان دسترسی، به Dataهای سطر بعدی نسبت به سطر جاری را در نتیجه یک پرس و جو (Query)، ارائه می‌دهد. بدون آنکه از Self-join استفاده نمایید،
Syntax تابع فوق بصورت زیر است:

```
LEAD ( scalar_expression [ ,offset ] , [ default ] )
    OVER ( [ partition_by_clause ] order_by_clause )
```

شرح Syntax:

Scalar_expression: در Scalar_expression، نام یک فیلد یا ستون درج می‌شود، و مقدار برگشتی فیلد مورد نظر، به مقدار تعیین شده offset نیز بستگی دارد. خروجی Scalar_expression فقط یک مقدار است.
offset: منظور از Offset در این Syntax همانند عملکرد Offset در Syntax مربوط به Over می‌باشد. یعنی هر عددی برای offset در نظر گرفته شود، بیانگر نقطه آغازین سطر بعدی یا قبلی نسبت به سطر جاری است. به بیان دیگر، عدد تعیین شده در Offset به Sql server می‌فهماند چه تعداد سطر را در محاسبه در نظر نگیرد.
Default: زمانی که برای Offset مقداری را تعیین می‌نمایید، SQL Server به تعداد تعیین شده در Offset، سطرها را در نظر نمی‌گیرد، بنابراین مقدار خروجی Scalar_expression بطور پیش فرض Null در نظر گرفته می‌شود، چنانچه بخواهید، مقداری غیر از Null درج نمایید، می‌توانید مقدار دلخواه را در قسمت Default وارد کنید.
OVER **(** [partition_by_clause] order_by_clause) : در بخش اول بطور کامل توضیح داده شده است.

برای درک بهتر Lead Function چند مثال را بررسی می‌نماییم:
ابتدا Script زیر را اجرا می‌نماییم، که شامل ایجاد یک جدول و درج 18 رکورد در آن:

```
Create Table TestLead_LAG
(SalesOrderID int not null,
 SalesOrderDetailID int not null ,
 OrderQty smallint not null);
 GO
Insert Into TestLead_LAG
       Values (43662,49,1),(43662,50,3),(43662,51,1),
          (43663,52,1),(43664,53,1),(43664,54,1),
  (43667,77,3),(43667,78,1),(43667,79,1),
  (43667,80,1),(43668,81,3),(43669,110,1),
  (43670,111,1),(43670,112,2),(43670,113,2),
  (43670,114,1),(43671,115,1),(43671,116,2)
```

مثال:قصد داریم در هر سطر مقدار بعدی فیلد SalesOrderDetailID در فیلد دیگری به نام LeadValue نمایش دهیم، بنابراین Script زیر را ایجاد می‌کنیم:

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
LEAD(SalesOrderDetailID) OVER (ORDER BY SalesOrderDetailID) LeadValue
FROM TestLead_LAG s
WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
```

```
ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
```

خروجی بصورت زیر خواهد بود:

	SalesOrderID	SalesOrderDetailID	OrderQty	LeadValue
1	43663	52	1	77
2	43667	77	3	78
3	43667	78	1	79
4	43667	79	1	80
5	43667	80	1	110
6	43669	110	1	111
7	43670	111	1	112
8	43670	112	2	113
9	43670	113	2	114
10	43670	114	1	NULL

مطابق شکل، براحتی واضح است، که در هر سطر مقدار بعدی فیلد SalesOrderDetailID در فیلد LeadValue درج و نمایش داده می‌شود. فقط در سطر 10، چون مقدار بعدی برای فیلد SalesOrderDetailID وجود ندارد، SQL Server مقدار فیلد LeadValue را، Null در نظر می‌گیرد.

در این مثال فقط از آرگومان Scalar_expression، استفاده کردیم، و Offset و Default را مقدار دهی ننمودیم، بنابراین SQL Server بطور پیش فرض هیچ سطری را حذف نمی‌کند و مقدار Default را Null در نظر می‌گیرد.

مثال دوم: قصد داریم در هر سطر مقدار دو سطر بعدی فیلد SalesOrderDetailID را در فیلد LeadValue نمایش دهیم، و در صورت وجود نداشتن مقدار فیلد SalesOrderDetailID، مقدار پیش فرض صفر ،در فیلد LeadValue قرار دهیم،بنابراین Script آن بصورت زیر خواهد شد:

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
LEAD(SalesOrderDetailID,2,0) OVER (ORDER BY SalesOrderDetailID) LeadValue
FROM TestLead_LAG s
WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
```

خروجی:

	SalesOrderID	SalesOrderDetailID	OrderQty	LeadValue
1	43663	52	1	78
2	43667	77	3	79
3	43667	78	1	80
4	43667	79	1	110
5	43667	80	1	111
6	43669	110	1	112
7	43670	111	1	113
8	43670	112	2	114
9	43670	113	2	0
10	43670	114	1	0

در صورت مسئله بیان کرده بودیم، در هر سطر،مقدار فیلد SalesOrderDetailID دو سطر بعدی، را نمایش دهیم، بنابراین مقداری که برای Offset در نظر می‌گیریم، برابر دو خواهد بود، سپس گفته بودیم، چنانچه در هر سطر مقدار فیلد SalesOrderDetailID وجود نداشت،بجای مقدار پیش فرض Null،از مقدار صفر استفاده شود، بنابراین به Default مقدار صفر را نسبت دادیم.

```
LEAD(SalesOrderDetailID,2,0)
```

در شکل، مطابق صورت مسئله، مقدار فیلد LeadValue سطر اول برابر است با 78،
به بیان ساده‌تر برای بدست آوردن مقدار فیلد LaedValue هر سطر، می‌بایست هر سطر را به علاوه 2 (Offset) نماییم، تا سطر بعدی بدست آید، سپس مقدار SalesOrderDetailID را در فیلد LeadValue قرار می‌دهیم.
به سطر 9 و 10 توجه نمایید، که مقدار فیلد LeadValue آنها برابر با صفر است، واضح است، سطر 10 + 2 برابر است با 12(12=2+10)، چنین سطری در خروجی نداریم، بنابراین بطور پیش فرض مقدار LeadVaule توسط Sql Server برابر Null در نظر گرفته می‌شود، اما نمی‌خواستیم، که این مقدار Null باشد، بنابراین به آرگومان Default مقدار صفر را نسبت دادیم، تا SQL Server ، به جای استفاده از Null، مقدار در نظر گرفته شده صفر را استفاده نماید.
اگر چنین فانکشنی وجود نداشت، برای شبیه سازی آن می‌بایست از Join روی خود جدول استفاده می‌نمودیم، و یکسری محاسبات دیگر، که کار را سخت می‌نمود، مثال دوم را با Script زیر می‌توان شبیه سازی نمود:

```
WITH cteLead
AS
(
SELECT SalesOrderID,SalesOrderDetailID,OrderQty,
       ROW_NUMBER() OVER (ORDER BY SalesOrderDetailID) AS sn
FROM TestLead_LAG
WHERE
SalesOrderID IN (43670, 43669, 43667, 43663)
)
SELECT m.SalesOrderID, m.SalesOrderDetailID, m.OrderQty,
       case  when sLead.SalesOrderDetailID is null Then 0 Else sLead.SalesOrderDetailID END as
leadvalue
FROM cteLead AS m
LEFT OUTER JOIN cteLead AS sLead ON sLead.sn = m.sn+2
ORDER BY m.SalesOrderID, m.SalesOrderDetailID, m.OrderQty
```

جدول موقتی ایجاد نمودیم، که ROW_Number را در آن اضافه کردیم، سپس جدول ایجاد شده را با خود Join کردیم، و گفتیم، که مقدار فیلدLeadValue هر سطر برابر است با مقدار فیلد SalesOrderDetailID دو سطر بعد از آن. و با Case نیز مقدار پیش فرض را صفر در نظر گرفتیم.

LAG Function:

این فانکشن نیز در SQL Server 2012 ارائه شده است، و امکان دسترسی، به Dataهای سطر قبلی نسبت به سطر جاری را در نتیجه یک پرس و جو (Query)، ارائه می‌دهد. بدون آنکه از Self-join استفاده نمایید،

Syntax آن شبیه به فانکشن Lead میباشد و بصورت زیر است:

```
LAG (scalar_expression [,offset] [,default])
    OVER ( [ partition_by_clause ] order_by_clause )
```

Syntax مربوط به فانکشن LAG را شرح نمی‌دهم، بدلیل آنکه شبیه به فانکشن Lead می‌باشد، فقط تفاوت آن در Offset است، Offset در فانکشن LAG روی سطرهای ماقبل سطر جاری اعمال می‌گردد.

مثال دوم را برای حالت LAG Function شبیه سازی می‌نماییم:

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
LAG(SalesOrderDetailID,2,0) OVER (ORDER BY SalesOrderDetailID) LAGValue
FROM TestLead_LAG s
WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
go
```

خروجی :

	SalesOrderID	SalesOrderDetailID	OrderQty	LAGValue
1	43663	52	1	0
2	43667	77	3	0
3	43667	78	1	52
4	43667	79	1	77
5	43667	80	1	78
6	43669	110	1	79
7	43670	111	1	80
8	43670	112	2	110
9	43670	113	2	111
10	43670	114	1	112

همانطور که گفتیم، LAG Function عکس LEAD Function میباشد. یعنی مقدار فیلد LAGValue سطر جاری برابر است با مقدار SalesOrderDetailID دو سطر ما قبل خود.

مقدار فیلد LAGValue دو سطر اول و دوم نیز برابر صفر است، چون دو سطر ماقبل آنها وجود ندارد، و مقدار صفر نیز بدلیل این است که Default را برابر صفر در نظر گرفته بودیم.

مثال: در این مثال از Laed Function و LAG Function بطور همزمان استفاده می‌کنیم، با این تفاوت، که از گروه بندی نیز استفاده شده است:

Script زیر را اجرا نمایید:

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       Lead(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID ORDER BY SalesOrderDetailID) LeadValue,
       LAG(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID ORDER BY SalesOrderDetailID) LAGValue
FROM TestLead_LAG s
WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
go
```

خروجی:

	SalesOrderID	SalesOrderDetailID	OrderQty	LeadValue	LAGValue
1	43663	52	1	NULL	NULL
2	43667	77	3	78	NULL
3	43667	78	1	79	77
4	43667	79	1	80	78
5	43667	80	1	NULL	79
6	43669	110	1	NULL	NULL
7	43670	111	1	112	NULL
8	43670	112	2	113	111
9	43670	113	2	114	112
10	43670	114	1	NULL	113

با بررسی هایی که در مثالهای قبل نمودیم،خروجی زیر را می‌توان براحتی تشخیص داد، و توضیح بیشتری نمی‌دهم.
موفق باشید.

نظرات خوانندگان

نویسنده: محمد
تاریخ: ۱۳۹۱/۱۰/۲۸ ۱۸:۵۳

سلام،
این توابع واقعا کار رو آسون کردن، ما رو از بکارگیری چندین بار self join بی نیاز کردن.
بطور نمونه اگه بخواهیم مقدار SalesOrderDetailID سطر قبلی، دو سطرقبلی، سطربعدی و دو سطر بعدی را بدست بیاریم در نسخه 2008 ساده‌ترین و مناسب‌ترین کوئری این هست:

```
WITH cteLead
AS
(
SELECT SalesOrderID,SalesOrderDetailID,OrderQty,
       ROW_NUMBER() OVER (PARTITION BY SalesOrderID
                          ORDER BY SalesOrderDetailID) AS sn
FROM TestLead_LAG
WHERE
SalesOrderID IN (43670, 43669, 43667, 43663)
)
SELECT m.SalesOrderID, m.SalesOrderDetailID, m.OrderQty,
       COALESCE(sLead1.SalesOrderDetailID, 0) as leadvalue1,
       COALESCE(sLead2.SalesOrderDetailID, 0) as leadvalue2,
       COALESCE(sLag1.SalesOrderDetailID, 0) as lagvalue2,
       COALESCE(sLag2.SalesOrderDetailID, 0) as lagvalue2
FROM cteLead AS m
LEFT OUTER JOIN cteLead AS sLead1
ON m.sn = sLead1.sn - 1
AND m.SalesOrderID = sLead1.SalesOrderID
LEFT OUTER JOIN cteLead AS sLead2
ON m.sn = sLead2.sn - 2
AND m.SalesOrderID = sLead2.SalesOrderID
LEFT OUTER JOIN cteLead AS sLag1
ON m.sn = sLag1.sn + 1
AND m.SalesOrderID = sLag1.SalesOrderID
LEFT OUTER JOIN cteLead AS sLag2
ON m.sn = sLag2.sn + 2
AND m.SalesOrderID = sLag2.SalesOrderID
ORDER BY m.SalesOrderID, m.SalesOrderDetailID, m.OrderQty;
```

در حالی که با دو تابعی که شما در اینجا پوشش دادین میشه کوئری فوق را فوق العاده ساده‌تر نمود:

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       Lead(SalesOrderDetailID, 1, 0) OVER (PARTITION BY SalesOrderID ORDER BY SalesOrderDetailID)
LeadValue1,
       LAG(SalesOrderDetailID, 1, 0) OVER (PARTITION BY SalesOrderID ORDER BY SalesOrderDetailID)
LAGValue1,
       Lead(SalesOrderDetailID, 2, 0) OVER (PARTITION BY SalesOrderID ORDER BY SalesOrderDetailID)
LeadValue2,
       LAG(SalesOrderDetailID, 2, 0) OVER (PARTITION BY SalesOrderID ORDER BY SalesOrderDetailID)
LAGValue2,
FROM TestLead_LAG s
WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
```

نویسنده: حسین
تاریخ: ۱۳۹۱/۱۰/۳۰ ۱۴:۵۸

جالب بود مرسی

عنوان: **آشنایی با Window Function ها در SQL Server بخش چهارم**
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۱۰/۰۳ ۲۳:۳۰
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, SQL Server 2012, T-SQL

برای مطالعه این بخش لازم است، به Syntax مربوط به Over آشنا باشیم، در بخش اول بطور کامل به Syntax مربوط به Over پرداختیم.

در این بخش دو فانکشن دیگر از توابع تحلیلی (Analytic functions) به نامهای First_Value و Last_Value را بررسی می‌نماییم.

First_Value

این فانکشن نیز همانند دیگر فانکشنهای تحلیلی در نسخه SQL Server 2012 ارائه گردیده است. و اولین مقدار از یک مجموعه مقادیر را بر می‌گرداند. و Syntax آن بصورت ذیل می‌باشد:

```
FIRST_VALUE ( [scalar_expression ]
    OVER ( [ partition_by_clause ] order_by_clause [ rows_range_clause ] )
```

شرح Syntax:

1- Scalar_expression : مقدار آن می‌تواند نام یک فیلد یا Subquery باشد.

2- Over : در بخش اول بطور مفصل آن را بررسی نمودیم.

قبل از بررسی تابع First_Value،ابتدا Script زیر را اجرا نمایید، که شامل یک جدول و درج چند رکورد در آن است.

```
Create Table Test_First_Last_Value
(SalesOrderID int not null,
 SalesOrderDetailID int not null ,
 OrderQty smallint not null);
 GO
Insert Into Test_First_Last_Value
       Values (43662,49,1),(43662,50,3),(43662,51,1),
          (43663,52,1),(43664,53,1),(43664,54,1),
  (43667,77,3),(43667,78,1),(43667,79,1),
          (43667,80,1),(43668,81,3),(43669,110,1),
          (43670,111,1),(43670,112,2),(43670,113,2),
          (43670,114,1),(43671,115,1),(43671,116,2)
```

مثال: ابتدا Scriptی ایجاد می‌نماییم،بطوریکه جدول Test_Firts_Last_Value را براساس فیلد SalesOrderID گروه بندی نموده و اولین مقدار فیلد SalesOrderDetailID در هرگروه را مشخص نماید.

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       FIRST_VALUE(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID
       ORDER BY SalesOrderDetailID) FstValue
FROM Test_First_Last_Value s
     WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
     ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
```

خروجی:

	SalesOrderID	SalesOrderDetailID	OrderQty	FstValue
1	43663	52	1	52
2	43667	77	3	77
3	43667	78	1	77
4	43667	79	1	77
5	43667	80	1	77
6	43669	110	1	110
7	43670	111	1	111
8	43670	112	2	111
9	43670	113	2	111
10	43670	114	1	111

مطابق Script چهار گروه در خروجی ایجاد شده است و در فیلد FstValue ، اولین مقدار هر گروه نمایش داده می‌شود. اگر بخش‌های قبلی Window Functionها را مطالعه کرده باشید، تحلیل این تابع کار بسیار ساده ای است.

Last_Value

این تابع نیز در نسخه SQL Server 2012 ارائه گردیده است. و آخرین مقدار از یک مجموعه مقادیر را بر می‌گرداند، به عبارتی فانکشن Last_Value عکس فانکشن First_Value عمل می‌نماید و Syntax آن به شرح ذیل میباشد:

```
LAST_VALUE ( [scalar_expression ]
    OVER ( [ partition_by_clause ] order_by_clause rows_range_clause )
```

شرح Syntax تابع Last_Value شبیه به تابع First_Value می‌باشد.

مثال: همانند مثال قبل Scriptی ایجاد می‌نماییم،بطوریکه جدول Test_Firts_Last_Value را براساس فیلد SalesOrderID گروه بندی نموده و آخرین مقدار فیلد SalesOrderDetailID در هرگروه را مشخص نماید.

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       LAST_VALUE(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID
       ORDER BY SalesOrderDetailID RANGE BETWEEN UNBOUNDED PRECEDING  AND UNBOUNDED FOLLOWING)
LstValue
FROM Test_First_Last_Value s
     WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
     ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
```

خروجی:

	SalesOrderID	SalesOrderDetailID	OrderQty	LstValue
1	43663	52	1	52
2	43667	77	3	80
3	43667	78	1	80
4	43667	79	1	80
5	43667	80	1	80
6	43669	110	1	110
7	43670	111	1	114
8	43670	112	2	114
9	43670	113	2	114
10	43670	114	1	114

خروجی جدول،به چهار گروه تقسیم،و آخرین مقدار هر گروه،در فیلد LstValue نمایش داده شده است. در این مثال نیز تحلیلی نخواهیم داشت، چون فرض بر آن است که بخش‌های قبلی را مطالعه نموده ایم.

موفق باشید.

نظرات خوانندگان

نویسنده: محمد
تاریخ: ۱۳۹۱/۱۰/۲۸ ۱۰:۳۹

سلام،
مطلب اول: قسمت order by در ماده over در هر دو کوئری به چه جهت آمده است؟
مطلب دوم: first_value چه مزیتی نسبت به min() over داره، منظورم اینه که میشه خروجی کوئری اولتون رو با این کوئری بدست آورد:

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       MIN(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID) FstValue
FROM Test_First_Last_Value s
WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
```

نویسنده: محمد
تاریخ: ۱۳۹۱/۱۰/۲۸ ۱۲:۱۱

سلام،
من SQL Server 2012 ندارم، ولی تا اونجایی که متوجه شدم بر اساس شواهد دو کوئری زیر باید یک نتیجه رو برگردانند. منظورم اینکه که با first_value میشه last_value هم شبیه سازی کرد، فقط کافیه که در ماده order by از کلید واژه DESC استفاده بشه.
اگه من اشتباه میکنم لطفا راهنمایی بفرمایید.

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       LAST_VALUE(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID
       ORDER BY SalesOrderDetailID)  LstValue
FROM Test_First_Last_Value s
     WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
     ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty

SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       FITST_VALUE(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID
       ORDER BY SalesOrderDetailID DESC) FstValue
FROM Test_First_Last_Value s
     WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
     ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
```

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۱۰/۲۹ ۱۱:۵۳

سلام
جواب سئوال اول: در Syntax تابع First_value استفاده از Order by اجباری می‌باشد.
جواب سئوال دوم:
First_Value اولین مقدار یا اولین Row در یک گروه را مشخص می‌کند و به مفهوم کوچکترین مقدار نمی‌باشد، شاید، مثالی که در مقاله زدم شما را به اشتباه انداخت، در زیر با یک مثال، First_value و Min را مقایسه می‌کنیم.
ابتدا یک جدول و چند رکورد، در آن درج می‌کنیم:

```
CREATE TABLE Employees (
    EmployeeId INT IDENTITY PRIMARY KEY,
    Name VARCHAR(50),
    HireDate DATE NOT NULL,
    Salary INT NOT NULL
    )
GO
```

```
INSERT INTO Employees (Name, HireDate, Salary)
VALUES
    ('Alice', '2011-01-01', 20000),
    ('Brent', '2011-01-15', 19000),
    ('Carlos', '2011-02-01', 22000),
    ('Donna', '2011-03-01', 25000),
    ('Evan', '2011-04-01', 18500)
GO
```

در ادامه Script زیر را اجرا می‌کنیم:

```
Select EmployeeId,Name,Salary,HireDate,
   First_VALUE(HireDate) OVER(ORDER BY Salary RANGE BETWEEN UNBOUNDED PRECEDING
                              AND UNBOUNDED FOLLOWING) AS First,
   Min(HireDate) OVER(ORDER BY Salary
                      RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING) AS Min
FROM Employees
ORDER BY EmployeeId
GO
```

خروجی بصورت زیر می‌شود:

	EmployeeId	Name	Salary	HireDate	First	Min
1	1	Alice	20000	2011-01-01	2011-04-01	2011-01-01
2	2	Brent	19000	2011-01-15	2011-04-01	2011-01-01
3	3	Carlos	22000	2011-02-01	2011-04-01	2011-01-01
4	4	Donna	25000	2011-03-01	2011-04-01	2011-01-01
5	5	Evan	18500	2011-04-01	2011-04-01	2011-01-01

در شکل بالا تفاوت Min و First_Value بطور کامل مشخص است، اگر به Query دقت نمایید، Sort براساس Salary انجام شده است، برای حالت First_value ،مقدار فیلد HireDate در اولین رکورد، برابر است با 2011-04-01 ، بنابراین سورت روی نمایش First_value تاثیر گذار است، بطوریکه Sort برای حالت Min، تاثیر گذار نمی‌باشد، و تابع Min ، کوچکترین مقدار، از مقادیر ستون HireDate را بدست می‌آورد، به بیان ساده‌تر در حالت استفاده از Min، عملیات Sort بیهوده می‌باشد. چون تابع MIN روی کل مقادیر یک گروه یا ستون تاثیر می‌گذارد.

نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۱/۱۰/۲۹ ۱۲:۵۱

سلام

شما می‌توانید، با دستکاری Queryها خروجی‌های یکسانی را ایجاد نمایید، دو Query که ایجاد نمودید، خروجی یکسانی ندارند. Query دوم شما با خروجی Last_Value،مقاله یکسان است، اما باید بگویم که مفهوم Last_Value این است که آخرین سطر در یک گروه را بر می‌گرداند. بهتر است بخش اول را مطالعه نمایید.

علت یکسان نبودن نتیجه دو Query شما در نحوه Sort و مفهوم First_value و Last_Value می‌باشد:

نکته: اگر در Over Clause شرط Order by را اعمال نماییم، اما از Row یا Range استفاده نکنیم، SQL Server بصورت پیش فرض از قالب زیر استفاده می‌نماید:

```
RANGE UNBOUNDED PRECEDING AND CURRENT ROW
```

نویسنده: محمد
تاریخ: ۱۳۹۱/۱۰/۲۹ ۱۳:۲۳

ممنون از پاسختون، الان متوجه تفاوتشون شدم.

ستون older_method باید مقادیرش دقیقا مشابه با first_value شما باشد:

```
Select EmployeeId,Name,Salary,HireDate,
       First_VALUE(HireDate) OVER(ORDER BY Salary RANGE BETWEEN UNBOUNDED PRECEDING
                                  AND UNBOUNDED FOLLOWING) AS First,
   D.HireDate AS older_method
FROM Employees
CROSS APPLY (SELECT TOP 1 HireDate
   FROM Employees
  --WHERE E1.EmployeeId = E2.EmployeeId
  ORDER BY Salary ASC) AS D
ORDER BY EmployeeId;
```

نویسنده: محمد
تاریخ: ۱۳۹۱/۱۰/۲۹ ۱۳:۳۱

ممنون از شما، من مطالب بخش اول رو مطالعه کردم.

عبارت RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING به معنای تمام سطرهای جدول هست دیگه درسته.

یعنی تمام سطرهای جدول از اولین گرفته، جاری گرفته و آخرین رو پوشش میده.

با این توضیحات باید دو کوئری زیر اینبار جواب یکسانی بدهند:

```
SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       FIRST_VALUE(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID
       ORDER BY SalesOrderDetailID RANGE BETWEEN UNBOUNDED PRECEDING  AND UNBOUNDED FOLLOWING)
LstValue
FROM Test_First_Last_Value s
     WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
     ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty

SELECT s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty,
       LAST_VALUE(SalesOrderDetailID) OVER (PARTITION BY SalesOrderID
       ORDER BY SalesOrderDetailID DESC RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING)
LstValue
FROM Test_First_Last_Value s
     WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
     ORDER BY s.SalesOrderID,s.SalesOrderDetailID,s.OrderQty
```

دو کوئری کاملا یکسان هستند به غیر از اینکه در کوئری دوم یک DESC اضافه شده و نام تابع از first به last تغییر کرده است.

عنوان: محاسبه میانگین متحرک (moving average) در SQL Server 2012
نویسنده: محمد سلیم آبادی
تاریخ: ۱۳۹۱/۱۱/۱۴ ۱۹:۲۵
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, window function, querying, moving average

شرح مساله

میانگین متحرک یا moving average به چند دسته تقسیم می‌شود که ساده‌ترین آنها میان متحرک ساده است.
برای محاسبه میانگین متحرک باید بازه زمانی مورد نظر را مشخص کنیم. مثلا میانگین فروش در 3 روز گذشته.

به جدول زیر توجه بفرمایید:

روز	فروش	میانگین متحرک 3 روزه	میانگین متحرک 4 روزه
1	10	***	***
2	12	***	***
3	13	***	***
4	16	11.6	***
5	19	13.6	12.7
6	23	16.0	15.0
7	26	19.3	17.7

میانگین متحرک فروش سه روز و چهار روز گذشته در جدول فوق قابل مشاهده است.
بطور مثال مقدار میانگین متحرک سه روزه برای روز چهارم برابر است با جمع فروش سه روز گذشته تقسیم بر سه. یعنی
(13+12+10)/3
و برای روز ششم میانگین متحرک 4 روزه برابر است با جمع فروش چهار روز گذشته و تقسیم آنها بر چهار. یعنی
16+13+12+10 تقسیم بر 4 که برابر است با 12.7

در نمودار زیر، خط قرم رنگ مربوط به میانگین متحرک ساده (میانگین فروش سه روز گذشته) است و خط آبی رنگ نیز میزان فروش است

راه حل در SQL Server 2012

توسط توابع window این مساله را به سادگی می‌توانیم حل کنیم. همانطور که مشاهده می‌شود در تصویر زیر. کافیست ما به سطرهایی در بازه‌ی سه سطر قبل تا یک سطر قبل (برای میانگین متحرک سه روزه) دسترسی پیدا کرده و میانگین آن را بگیریم.

روز	فروش
1	10
2	12
3	13
4	16
5	19
6	23
7	26

ابتدا این جدول را ایجاد و تعدادی سطر برای نمونه در آن درج کنید:

```
CREATE TABLE Samples
(
[date] SMALLDATETIME,
selling SMALLMONEY
);

INSERT  Samples
VALUES
('2010-12-01 00:00:00', 10),
('2010-12-02 00:00:00', 12),
('2010-12-03 00:00:00', 13),
('2010-12-04 00:00:00', 16),
('2010-12-05 00:00:00', 19),
('2010-12-06 00:00:00', 23),
('2010-12-07 00:00:00', 26),
('2010-12-08 00:00:00', 27),
('2010-12-09 00:00:00', 20),
('2010-12-10 00:00:00', 18),
('2010-12-11 00:00:00', 19);
```

سپس برای محاسبه میانگین متحرک در بازه سه روز گذشته query زیر را اجرا کنید:

```
SELECT [date],
       selling,
       CASE WHEN rnk < 4 THEN NULL ELSE mv END AS SimpleMovingAverage
  FROM (SELECT *,
               AVG(selling) OVER(ORDER BY [date]
                                 ROWS BETWEEN 3 PRECEDING AND 1 PRECEDING) AS mv,
               ROW_NUMBER() OVER(ORDER BY [date]) AS rnk
          FROM Samples
       ) AS d;
```

قلب query دستور ROWS BETWEEN 3 PRECEDING AND 1 PRECEDING می‌باشد.

به این معنا که سطرهایی در بازه‌ی سه سطر قبل و یک سطر قبل را در Window انتخاب کرده و عمل میاگنین گیری را بر اساس مقادیر مورد نظر انجام بده.

راه حل در SQL Server 2005

به درخواست یکی از کاربران من راه حلی را پیشنهاد می‌کنم که جایگزین مناسبی برای روش قبلی است در صورت عدم استفاده از نسخه 2012. توابع window در اینگونه مسائل بهترین عملکرد را خواهند داشت.

```
SELECT S.[date], S.selling, CASE WHEN COUNT(*) < 3 THEN NULL ELSE AVG(s) END AS SimpleMovingAverage
  FROM Samples AS S
       OUTER APPLY (SELECT TOP(3) selling
                      FROM Samples
                     WHERE [date] < S.[date]
                     ORDER BY [date] DESC) AS D(s)
 GROUP BY S.[date], S.selling
 ORDER BY S.[date];
```

FOR FUN

توسط توابع Analytical ای چون LAG نیز می‌توان اینگونه مسائل را حل نمود. بطور مثال توسط تابع LAG به یک مقدار قبلی، دو مقدار قبلی و سه مقدار قبلی دسترسی پیدا کرده و آنها را با یکدیگر جمع نموده و تقسیم بر تعدادشان می‌کنیم یعنی:

```
select [date],
       selling,
       (
       lag(selling, 1) over(order by [date]) +
       lag(selling, 2) over(order by [date]) +
       lag(selling, 3) over(order by [date])
       ) / 3
  from Samples;
```

نظرات خوانندگان

نویسنده: سعید
تاریخ: ۱۳۹۱/۱۱/۱۵ ۱۸:۰

ممنون از شما. لطفا در صورت امکان راه حل بدون استفاده از توابع window را هم جهت مقایسه ارائه کنید.

با تشکر بسیار

نویسنده: محمد سلم ابادی
تاریخ: ۱۳۹۱/۱۱/۱۵ ۱۸:۳۷

یک راه حل جدید بدون کمک گرفتن از توابع Window به مقاله افزوده شد.

نویسنده: اسحق مهرجویی
تاریخ: ۱۳۹۲/۱۱/۱۸ ۱۳:۹

با سلام و تشکر از شما. برای محاسبه میانگین متحرک در این سایت به شیوه زیر عمل کرده. می‌تونید یه توضیحی راجع به اون بدهید.

```
SELECT
  T0.StockId
  ,T0.QuoteId
  ,T0.QuoteDay
  ,T0.QuoteClose
  , AVG (T0.QuoteClose) OVER (PARTITION BY T0.StockId ORDER BY T0.QuoteId ROWS 19 PRECEDING) AS MA20
FROM
  dbo.Quotes AS T0
```

نویسنده: محمد سلیم آبادی
تاریخ: ۱۳۹۲/۱۱/۱۸ ۱۵:۱۵

سلام اسحق،
در مثالی که من تهیه کردم، میانگین داده‌های مربوط به 3 سطر قبل محاسبه شده، بدون لحاظ مقدار جاری. اما در مساله مربوط به آن سایت میانگین داده‌های مرتبط به 19 سطر قبل و سطر جاری محاسبه شده.
و همچنین در بخش Specification مربوط به آن تابع میانگین، در مثال سایت از PARTITION استفاده شده آن هم به این خاطر که داده‌های جدول به گروه‌های مختلفی بر اساس مقادیر ستون StockId تقسیم شده است. و میخواسته میانگین مرتبط به هر StockId بطور مجزا محاسبه بشه. در واقع نتیجه را به تعداد StorckIdها بخش بندی کرده.

نام مستعاری که به جدول Quotes داده شده، غیر ضروری بوده، چرا که تنها یک جدول بیشتر در Query مشارکت نداشته و نیازی به ذکر نام جدول یا نام مستعار جدول نیست.
همچنین برای شفافیت بیشتر و ابهام زدایی، بهتر است قسمت Rows تابع تجمعی را کامل و صریح بنویسیم به این صورت:

```
SELECT StockId, QuoteId, QuoteDay, QuoteClose,
       AVG (QuoteClose) OVER (PARTITION BY StockId
                              ORDER BY QuoteId
                               ROWS BETWEEN 19 PRECEDING AND CURRENT ROW) AS MA20
FROM dbo.Quotes AS T0;
```

عنوان: **آشنایی با Window Function ها در SQL Server بخش پنجم**
نویسنده: فرهاد فرهمندخواه
تاریخ: ۱۳۹۲/۰۱/۲۹ ۱۸:۵۰
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, SQL Server 2012, T-SQL

در این بخش فانکشن دیگری از توابع تحلیلی به نام CUME_DIST را بررسی می‌نماییم.

:CUME_DIST

بوسیله تابع CUME_DIST می‌توان ارزیابی نمود، در یک گروه، چه درصد از مقادیر،مساوی یا کوچکتر از مقدار سطر جاری می‌باشند، به این تابع cumulative distribution نیز گفته می‌شود.

Syntax تابع CUME_DIST به صورت زیر است:

```
CUME_DIST( )
    OVER ( [ partition_by_clause ] order_by_clause )
```

شرح Syntax:

1- Partition By Clause : بوسیله پارامتر فوق می‌توانید، نتیجه پرس جو (Query)،خود را دسته بندی نمایید.

2- order by clause : همانطور که از نامش مشخص است، جهت مرتب نمودن خروجی Query می‌باشد.

معمولا شرح عملکرد توابع تحلیلی، کمی مشکل است. بنابراین برای درک، عملکرد تابع CUME_DIST چند مثال را بررسی می‌کنیم.

در ابتدا بوسیله Script زیر یک جدول ایجاد و 10 رکورد در آن درج می‌کنیم:

```
Create Table TestCUME_DIST
(SalesOrderID int not null,
OrderQty smallint not null,
ProductID int not null
 );
 GO
Insert Into TestCUME_DIST
       Values (43663,1,760),(43667,3,710),(43667,1,773),
  (43667,1,775),(43667,1,778),(43669,1,747),
  (43670,1,709),(43670,2,710),(43670,2,773),(43670,1,776)
```

مثال اول: Script زیر را اجرا می‌کنیم، سپس خروجی آن را بررسی می‌نماییم:

```
SELECT SalesOrderID, OrderQty,
       CUME_DIST() OVER(ORDER BY SalesOrderID) AS [CUME_DIST]
FROM TestCUME_DIST ORDER BY [CUME_DIST] DESC
```

پس از اجرا خروجی بصورت زیر خواهد بود:

	SalesOrderID	OrderQty	CUME_DIST
1	43670	1	1
2	43670	2	1
3	43670	2	1
4	43670	1	1
5	43669	1	0.6
6	43667	3	0.5
7	43667	1	0.5
8	43667	1	0.5
9	43667	1	0.5
10	43663	1	0.1

در ادامه اجازه دهید،مقادیری که در فیلد CUME_DIST بدست آمده است را بصورت تصویری بررسی کنیم.
مقادیر سطر اول تا چهارم:

	SalesOrderID	OrderQty	CUME_DIST
1	43670	1	1
2	43670	2	1
3	43670	2	1
4	43670	1	1
5	43669	1	0.6
6	43667	3	0.5
7	43667	1	0.5
8	43667	1	0.5
9	43667	1	0.5
10	43663	1	0.1

سوال اول:چند سطر، SalesOrderID آن برابر ۴۳۶۷۰ می باشد؟
جواب: چهار سطر(Row)، فرض می کنیم C1=4

سوال دوم: چند سطر (Row)، زیر SalesOrderID=43670 قرار دارد؟
جواب: شش سطر (Row)، فرض می کنیم C2=6

سوال سوم: تعداد کل سطرها (Rows)، چند عدد می باشد؟
جواب: ۱۰ سطر (Rows) ، فرض می کنیم C3=10

برای بدست آوردن CUME_DIST چهار سطر اول از فرمول زیر پیروی می نماییم:
Rows= (C1+C2)/C3= (4+6)/10 =1

*** برای بدست آوردن CUME_DIST سطر پنجم نیز خواهیم داشت:

Rows=(c1+c2)/c3 بنابراین خواهیم داشت: Rows=(1+5)/10=6/0

مثال دوم : ابتدا Script زیر را اجرا نمایید:

```
SELECT SalesOrderID, OrderQty, ProductID,
       CUME_DIST() OVER(PARTITION BY SalesOrderID ORDER BY ProductID ) AS [CUME_DIST]
FROM TestCUME_DIST
WHERE SalesOrderID IN (43670, 43669, 43667, 43663)
ORDER BY SalesOrderID DESC, [CUME_DIST] DESC
```

خروجی :

	SalesOrderID	OrderQty	ProductID	CUME_DIST
1	43670	1	776	1
2	43670	2	773	0.75
3	43670	2	710	0.5
4	43670	1	709	0.25
5	43669	1	747	1
6	43667	1	778	1
7	43667	1	775	0.75
8	43667	1	773	0.5
9	43667	3	710	0.25
10	43663	1	760	1

همانگونه که ملاحظه می‌کنید، در این مثال، خروجی، براساس SalesOrderID به چهار گروه تقسیم می‌شود و عملیات مرتب سازی روی فیلد ProductID انجام می گیرد، بنابراین CUME_DIST، روی هر گروه بر روی فیلد ProductID محاسبه می‌شود.

گروه اول : نحوه محاسبه Cume_DIST سطر اول:

سوال:چه تعداد از مقادیر ProductID آن برابر 776 میباشد؟

جواب: فقط مقدار سطر اول، بنابراین خواهیم داشت C1=1

سوال: چه تعداد از مقادیر کوچکتر از ProductID=776 می‌باشد؟

جواب: مقدار سه سطر، در واقع مقادیر سطر دوم،سوم و چهارم کوچکتر از مقدار سطر اول می‌باشند، c2=3

سوال: تعداد کل سطرهای گروه اول چه مقدار می‌باشد؟

جواب: 4سطر

بنابراین برای بدست آوردن CUME_DIST سطر اول خواهیم داشت:

Rows=(1+3)/4=1

محاسبه سطر دوم از گروه اول بدون شرح:

Rows=(1+2)/4=0/75

امیدوارم مفید واقع شده باشد.

عنوان: **افزودن یک DataType جدید برای نگه‌داری تاریخ خورشیدی - 1**
نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۰۹ ۲۳:۳۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, SQL Server 2012, C#.NET, Persian, CLR

ثبت و نگه‌داری تاریخ خورشیدی در SQL Server از دیرباز یکی از نگرانی‌های برنامه‌نویسان و طراحان پایگاه داده‌ها بوده است. در این نوشتار، راه‌کار تعریف یک DataType در SQL Server 2012 به روش CLR آموزش داده خواهد شد.

در ویژوال استودیو یک پروژه‌ی جدید از نوع SQL Server Database Project به شکل زیر ایجاد کنید:

نام پروژه را به یاد تقویم خیام، prgJalaliDate می‌گذارم. در Solution Explorer روی نام پروژه راست‌کلیک کرده، سپس روی Add New Item کلیک کنید. در پنجره‌ی بازشده مطابق شکل SQL CLR C# User Defined Type را برگزینید؛ سپس نام JalaliDateType را برای آن انتخاب کنید.

متن موجود در صفحه‌ی بازشده را کاملاً حذف کرده و با کد زیر جای‌گزین کنید.

(در کد زیر همه‌ی توابع لازم برای مقداردهی به سال، ماه، روز، ساعت، دقیقه و ثانیه و البته گرفتن مقدار از آن‌ها، تبدیل تاریخ خورشیدی به میلادی، گرفتن تاریخ به تنهایی، گرفتن زمان به تنهایی، افزایش یا کاهش زمان برپایه‌ی یکی از متغیرهای زمان و بررسی و اعتبارسنجی انواع بخش‌های زمان گنجانده شده است. در صورت پرسش یا پیشنهاد روی هر کدام در قسمت نظرات، پیام خود را بنویسید.)

```
using System;
using System.Data.SqlTypes;
using Microsoft.SqlServer.Server;

[Serializable()]
[SqlUserDefinedType(Format.Native)]
public struct JalaliDate : INullable
{
    private Int16 m_Year;
    private byte m_Month;
    private byte m_Day;
    private byte m_Hour;
    private byte m_Minute;
    private byte m_Second;
    private bool is_Null;


    public Int16 Year
    {
        get
        {
            return (this.m_Year);
        }
        set
        {
            m_Year = value;
        }
    }

    public byte Month
    {
        get
        {
            return (this.m_Month);
```

```
        }
        set
        {
            m_Month = value;
        }
    }

    public byte Day
    {
        get
        {
            return (this.m_Day);
        }
        set
        {
            m_Day = value;
        }
    }

    public byte Hour
    {
        get
        {
            return (this.m_Hour);
        }
        set
        {
            m_Hour = value;
        }
    }

    public byte Minute
    {
        get
        {
            return (this.m_Minute);
        }
        set
        {
            m_Minute = value;
        }
    }

    public byte Second
    {
        get
        {
            return (this.m_Second);
        }
        set
        {
            m_Second = value;
        }
    }

    public bool IsNull
    {
        get
        {
            return is_Null;
        }
    }

    public static JalaliDate Null
    {
        get
        {
            JalaliDate jl = new JalaliDate();
            jl.is_Null = true;
            return (jl);
        }
    }


    public override string ToString()
    {
        if (this.IsNull)
        {
            return "NULL";
        }
        else
```

```
        {
            return this.m_Year.ToString("D4") + "/" + this.m_Month.ToString("D2") + "/" +
this.m_Day.ToString("D2") + " " + this.Hour.ToString("D2") + ":" + this.Minute.ToString("D2") + ":" +
this.Second.ToString("D2");
        }
    }


    public static JalaliDate Parse(SqlString s)
    {
        if (s.IsNull)
        {
            return Null;
        }

        System.Globalization.PersianCalendar pers = new System.Globalization.PersianCalendar();
        string str = Convert.ToString(s);
        string[] JDate = str.Split(' ')[0].Split('/');

        JalaliDate jl = new JalaliDate();

        jl.Year = Convert.ToInt16(JDate[0]);
        byte MonthsInYear = (byte)pers.GetMonthsInYear(jl.Year);
        jl.Month = (byte.Parse(JDate[1]) <= MonthsInYear ? (byte.Parse(JDate[1]) > 0 ?
byte.Parse(JDate[1]) : (byte)1) : MonthsInYear);
        byte DaysInMonth = (byte)pers.GetDaysInMonth(jl.Year, jl.Month); ;
        jl.Day = (byte.Parse(JDate[2]) <= DaysInMonth ? (byte.Parse(JDate[2]) > 0 ?
byte.Parse(JDate[2]) : (byte)1) : DaysInMonth);
        if (str.Split(' ').Length > 1)
        {
            string[] JTime = str.Split(' ')[1].Split(':');
            jl.Hour = (JTime.Length >= 1 ? (byte.Parse(JTime[0]) < 23 && byte.Parse(JTime[0]) >=
(byte)0 ? byte.Parse(JTime[0]) : (byte)0) : (byte)0);
            jl.Minute = (JTime.Length >= 2 ? (byte.Parse(JTime[1]) < 59 && byte.Parse(JTime[1]) >=
(byte)0 ? byte.Parse(JTime[1]) : (byte)0) : (byte)0);
            jl.Second = (JTime.Length >= 3 ? (byte.Parse(JTime[2]) < 59 && byte.Parse(JTime[2]) >=
(byte)0 ? byte.Parse(JTime[2]) : (byte)0) : (byte)0);
        }
        else { jl.Hour = 0; jl.Minute = 0; jl.Second = 0; }

        return (jl);
    }

    public SqlString GetDate()
    {
        return this.m_Year.ToString("D4") + "/" + this.m_Month.ToString("D2") + "/" +
this.m_Day.ToString("D2");
    }

    public SqlString GetTime()
    {
        return this.Hour.ToString("D2") + ":" + this.Minute.ToString("D2") + ":" +
this.Second.ToString("D2");
    }

    public SqlDateTime ToGregorianTime()
    {
        System.Globalization.PersianCalendar pers = new System.Globalization.PersianCalendar();
        return SqlDateTime.Parse(pers.ToDateTime(this.Year, this.Month, this.Day, this.Hour,
this.Minute, this.Second, 0).ToString());
    }

    public SqlString JalaliDateAdd(SqlString interval, int increment)
    {
        System.Globalization.PersianCalendar pers = new System.Globalization.PersianCalendar();
        DateTime dt = pers.ToDateTime(this.Year, this.Month, this.Day, this.Hour, this.Minute,
this.Second, 0);
        string CInterval = interval.ToString();
        bool isConvert = true;
        switch (CInterval)
        {
            case "Year":
                dt = pers.AddYears(dt, increment);
                break;
            case "Month":
                dt = pers.AddMonths(dt, increment);
                break;
            case "Day":
                dt = pers.AddDays(dt, increment);
                break;
            case "Hour":
```

```
                dt = pers.AddHours(dt, increment);
                break;
            case "Minute":
                dt = pers.AddMinutes(dt, increment);
                break;
            case "Second":
                dt = pers.AddSeconds(dt, increment);
                break;
            default:
                isConvert = false;
                break;
        }

        if (isConvert == true)
        {
            this.Year = (Int16)pers.GetYear(dt);
            this.Month = (byte)pers.GetMonth(dt);
            this.Day = (byte)pers.GetDayOfMonth(dt);
            this.Hour = (byte)pers.GetHour(dt);
            this.Minute = (byte)pers.GetMinute(dt);
            this.Second = (byte)pers.GetSecond(dt);
        }


        return this.m_Year.ToString("D4") + "/" + this.m_Month.ToString("D2") + "/" +
this.m_Day.ToString("D2") + " " + this.Hour.ToString("D2") + ":" + this.Minute.ToString("D2") + ":" +
this.Second.ToString("D2");
    }
}
```

از منوهای بالا روی منوی Bulild و سپس گزینه‌ی Publish prgJalaliDate کلیک کنید:

در پنجره‌ی بازشده روی دکمه‌ی Edit کلیک کنید سپس تنظیمات مربوط به اتصال به پایگاه داده را انجام دهید.

روی دکمه‌ی OK کلیک کنید و سپس در پنجره‌ی اولیه، روی دکمه‌ی Publish کلیک کنید:

به همین سادگی، DataType مربوطه در SQL Server 2012 ساخته می‌شود. خبر خوش این‌که شما می‌توانید با راست‌کلیک روی نام پروژه و انتخاب گزینه‌ی Properties در قسمت Project Setting تنظیمات مربوط به نگارش SQL Server را انجام دهید. (از نگارش 2005 به بعد در VS 2012 پشتیبانی می‌شود.)

اکنون زمان آن رسیده است که DataType ایجادشده را در SQL Server 2012 بیازماییم. SQL Server را باز کنید و دستور زیر را در آن اجرا کنید.

```
USE Northwind

GO

CREATE TABLE dbo.TestTable
(
Id int NOT NULL IDENTITY (1, 1),
TestDate dbo.JalaliDate NULL
)  ON [PRIMARY]
GO
```

همین‌طور که مشاهده می‌کنید؛ امکان به‌کارگیری DataType تعریف‌شده وجود دارد.
اکنون چند رکورد درون این جدول درج می‌کنیم:

```
Insert into TestTable (TestDate) Values ('1392/02/09'),('1392/02/09 22:40'),('1392/12/30 22:40')
```

پس از اجرای این دستور خطای زیر در پایین صفحه‌ی SQL Server نمایان می‌شود:

```
Messages
Msg 6263, Level 16, State 1, Line 1
Execution of user code in the .NET Framework is disabled. Enable "clr enabled" configuration option.
```

این خطا به این خاطر است که CLR را در SQL Server فعال نکرده ایم. جهت فعال‌کردن CLR دستور زیر را اجرا کنید:

```
sp_configure 'clr enabled', 1
Reconfigure
```

بار دیگر دستور درج را اجرا می‌کنیم:

```
Insert into TestTable (TestDate) Values ('1392/02/09'),('1392/02/09 22:40'),('1392/12/30 22:40')
```

	Id	TestDate
▶	1	1392/02/09 00:00:00
	2	1392/02/09 22:40:00
	3	1392/12/29 22:40:00
*	NULL	NULL

ملاحظه می‌کنید که داده‌ها در جدول مربوطه ذخیره شده است. در رکورد نخست چون ساعت، دقیقه و ثانیه تعریف نشده است؛ به طور هوشمند صفر درج شده است. در رکورد دوم، ساعت و دقیقه مقدار دارد ولی ثانیه صفر ثبت شده است. و در رکورد سوم چون سال 1392 کبیسه نیست؛ به صورت هوشمند آخرین روز ماه به جای روز ثبت شده است. هرچند می‌توان با دست‌کاری در توابع سی‌شارپ، این قوانین را عوض کرد.

اکنون زمان آن رسیده است که توسط یک پرس‌وجو، همه‌ی توابعی که در سی‌شارپ برای این نوع داده نوشتیم، بیازماییم.
پرس‌وجوی زیر را اجرا کنید:

```
Select TestDate.ToString() as JalaliDateTime,
          TestDate.GetDate() as JalaliDate, TestDate.GetTime() as JalaliTime,
          TestDate.ToGregorianTime() as GregorianTime,
          TestDate.JalaliDateAdd('Day',1) JalaliTomorrow,
          TestDate.Month as JalaliMonth from TestTable
```

خروجی این پرس‌وجو به شکل زیر خواهد بود:

	JalaliDateTime	JalaliDate	JalaliTime	GregorianTime	JalaliTomorrow	JalaliMonth
1	1392/02/09 00:00:00	1392/02/09	00:00:00	2013-04-29 00:00:00.000	1392/02/10 00:00:00	2
2	1392/02/09 22:40:00	1392/02/09	22:40:00	2013-04-29 22:40:00.000	1392/02/10 22:40:00	2
3	1392/12/29 22:40:00	1392/12/29	22:40:00	2014-03-20 22:40:00.000	1393/01/01 22:40:00	12

البته درباره‌ی ستون پنجم و ششم شما می‌توانید روی همه‌ی اجزای تاریخ افزایش و کاهش داشته باشید و هم‌چنین می‌توانید با تابع مربوطه هر کدام از اجزای زمان را جداگانه به دست بیاورید که در این مثال عدد ماه نشان داده شده است.

نیازی به گفتن نیست که می‌توانید به سادگی از توابع مربوط به DateTime در SQL Server بهره ببرید. برای مثال برای به دست

آوردن فاصله‌ی میان دو روز از پرس‌وجوی زیر استفاده کنید:

```
Declare @a JalaliDate  = '1392/02/07 00:00:00'
Declare @b JalaliDate = '1392/02/05 00:00:00'

SELECT DATEDIFF("DAY",@b.ToGregorianTime(),@a.ToGregorianTime()) AS DiffDate
```

شاد و پیروز باشید.

نظرات خوانندگان

نویسنده: ali
تاریخ: ۱۳۹۲/۰۲/۱۰ ۰:۵

Nice article

Thanks

نویسنده: سام ناصری
تاریخ: ۱۳۹۲/۰۲/۱۰ ۴:۲۲

بسیار خوب.
فقط من تو مقاله شما دلیلی برای اینکه چرا زمان و تاریخ را میخواهی به این صورت ذخیره کنی متوجه نشدم؟ چرا به همان شکل استانداردش ذخیره نکنیم؟
دیگر اینکه نوع داده جدید به چه شکل در دیتابیس ذخیره میشود. Sql Server از کجا میداند که باید چگونه لیترالها را پارس کند(چگونه متود Parse برگزیده میشود)؟ عملگرهای مقایسه چگونه کار خواهند کرد؟

نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۰ ۷:۸

با درود
به پرسش‌های شما در بخش دوم پاسخ خواهم داد.

نویسنده: قاسم
تاریخ: ۱۳۹۲/۰۲/۱۰ ۹:۴۲

سلام، میشه مراحل انجام کار توی VS2010 هم بنویسید، فقط مراحل کار. خیلی ممنون میشم

نویسنده: فرشید علی اکبری
تاریخ: ۱۳۹۲/۰۲/۱۰ ۹:۵۵

سلام
کدهای شمارو درست همونطوریکه گفتین کپی کردم ولی موقعی که Publish رو میزنم پیغام زیر رو میده :
Publish cannot begin until your project(s) build successfully
و پنجره publish ظاهر نمی‌شه و هیچ جایی هم برای تعریف کانکشن وجود نداره... مشکل از کجاست؟
دوم اینکه clr رو هربار که خواستیم روی یک دیتابیس جدید این نوع رو تعریف کنیم باید فعال کنیم... مثلا توی ایجاد سال مالی جدید که سیستم یک دیتابیس خام ایجاد میکنه تا مانده حسابها رو بهش انتقال بدیم قبلش باید دستور فعال سازی clr رو هم مجددا بدیم یافقط توی زمان طراحی کفایت میکنه ؟
سوم اینکه : در مورد استفاده اون توی EF Code First هم اگه نکته‌ی خاصی وجود داره محبت کنین ممنون میشم.
شاد وپیروز باشید.

نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۰ ۱۱:۴

با درود
New Project -> Database -> SQL Server -> Visual C# SQL CLR Database Project
- تنظیمات اتصال به پایگاه داده ها
- انتخاب دکمه Yes

- Add New Item -> User Defined Type
- کپی کدها
- استفاده از Deploy در منوی Build یا استفاده از روشی که در بخش دوم نوشتار آموزش داده خواهد شد.

نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۰ ۱۱:۲۶

با درود
1- لطفاً پروژه را پیش از Publish یک بار Rebuild کنید. احتمالاً به خاطر یک ارور خاص Publish نمی‌شود. اگر به راحتی Rebuild شده ولی باز هم Publish کار نمی‌کند؛ می‌توانیم با روش دیگری که در بخش دوم آموزش می‌دهم DLL را به SQL معرفی کنیم.
2- کافی است یک بار پس ساخت Database این پرس‌وجو را اجرا کنید.
3- تست نکردم ولی به نظر می‌رسد این نوع داده از سمت EF شبیه به Hierarchy باشد.

نویسنده: ش.د
تاریخ: ۱۳۹۲/۰۲/۱۰ ۱۴:۳۵

آیا فقط در sql2012 قابل اجرا می‌باشد؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۰ ۱۴:۵۴

در متن نوشتن از نگارش 2005 به بعد اس کیوال سرور این قابلیت استفاده از افزونه‌های CLR اضافه شدن.

نویسنده: سید امیر سجادی
تاریخ: ۱۳۹۲/۰۲/۱۰ ۱۸:۳۸

با تشکر از مطلب مفیدتون.
چند تا سوال برام پیش اومده.
اول اینکه آیا به صورت یک DLL به بانک اضافه میشه؟
دوم اینکه اگه از بانک بک آپ بگیریم و جایی دیگه خواستیم اون رو ریستور کنیم چی میشه؟
آپلود بانک روی هاست (بک آپ یا اتچ) ؟

نویسنده: امیر بختیاری
تاریخ: ۱۳۹۲/۰۲/۱۱ ۹:۴۳

به صورت اسمبلی به بانک اضافه میشه
وقتی بک آپ بگیرید و ریستور کنید همراه بک آپ این اسمبلی هم انقال داده می‌شود
در اتچ هم به همین شکل
فقط زمانی که از این اسمبلی در توابع و پروسیجرها استفاده می‌کنید نمی‌تونید حذفش کنید و در صورت تغیرات باید اسمبلی را به روز کنید.
یک مورد دیگه که من زیاد تو هاست‌های شیر شده بهش برخوردم اینه که این امکان توشون فعال نیست و ادمین سرورها هم به سختی این امکان را فعال می‌کنند پس اگر خواستید از این امکان استفاده کنید ابتدا از فعال بودن آن مطمئن شوید

نویسنده: rahim
تاریخ: ۱۳۹۳/۰۵/۱۱ ۱۲:۱۴

با تشکر از مطلب مفیدتون
من زمانی که از این روش استفاده میکنم در هنگام درج رکورد جدید با پیغام خطای زیر مواجه میشم.
:"A .NET Framework error occurred during execution of user-defined routine or aggregate "JalaliDate

.System.OverflowException: Value was either too large or too small for an unsigned byte

نویسنده: محمد
تاریخ: ۱۳۹۳/۰۸/۲۲ ۰:۲۳

با سلام؛ من زمانی که این کد را در SQl 2012 میکنم یک Error در خصوص ToGregorianTime() نمایش داده میشه. لطفا راهنمایی بفرمایید. با تشکر

```
Msg 6522, Level 16, State 2, Line 1
A .NET Framework error occurred during execution of user-defined routine or aggregate
"SpatialDateTime":
System.FormatException: String was not recognized as a valid DateTime.
System.FormatException:
at System.DateTimeParse.ParseExactMultiple(String s, String[] formats, DateTimeFormatInfo dtfi,
DateTimeStyles style)
at System.DateTime.ParseExact(String s, String[] formats, IFormatProvider provider, DateTimeStyles
style)
at System.Data.SqlTypes.SqlDateTime.Parse(String s)
at SpatialDateTime.ToGregorianTime().
```

نویسنده: محسن خان
تاریخ: ۱۳۹۳/۰۸/۲۲ ۰:۵۵

یک try/catch بذار، تا بتونی تاریخ مشکل دار رو پیدا کنی:

```
var pers = new PersianCalendar();
var date = pers.ToDateTime(this.Year, this.Month, this.Day, this.Hour, this.Minute, this.Second,
0).ToString();
try
{
   return SqlDateTime.Parse(date);
}
catch(Exception ex)
{
  throw new InvalidOperationException("Can't parse "+ date);
}
```

عنوان: افزودن یک DataType جدید برای نگه‌داری تاریخ خورشیدی - 2
نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۰ ۲۱:۱۵
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, SQL Server 2012, Persian, CLR, SQL

پیش از هرچیز به شما پیش‌نهاد می‌کنم؛ بار دیگر کد سی‌شارپ درس نخست را در پروژه‌ی خود کپی کنید و سپس Publish را بزنید. پس از ارسال آن مطلب، تغییراتی در جهت بهینه‌سازی کد دادم که به نظرم بهتر است شما نیز در پروژه‌ی خود به کار برید.

چرا از این نوع داده استفاده کنیم؟

نخستین پرسشی که ممکن است برای شما پیش بیاید این است که چرا بهتر است از این نوع داده استفاده کنیم. برای پاسخ به این پرسش باید راه‌کارهای گذشته را بررسی کنیم. معمولاً طراحان پایگاه داده‌ها برای استفاده از تاریخ خورشیدی، زمان را به صورت میلادی ثبت می‌کنند؛ سپس با یک scalar-valued function زمان درج شده را به خورشیدی تبدیل می‌کنند. در این صورت می‌توان با یک تابع کوچک دیگر بخش مربوط به ساعت را نیز از همان ستون به دست آورد. در این صورت می‌توانیم از کلیه‌ی متدهای مربوط به DateTime در SQL از جمله افزایش و کاهش و تفاضل دو تاریخ بهره برد. برخی دیگر از طراحان، ستونی از نوع char(10) در نظر می‌گیرند و تاریخ خورشیدی را به صورت ده‌کاراکتری در آن ذخیره می‌کنند. این روش هرچند نیاز به تبدیل به خورشیدی را ندارد ولی کلیه‌ی مزایایی که در استفاده از DateTime به آن‌ها دسترسی داریم از دست می‌دهیم. افزون بر این جهت نگه‌داری زمان باید یک فیلد دیگر از نوع کاراکتری و یا در نگارش‌های نوین‌تر از نوع time تعریف کنیم. برخی دیگر از هر دو را در کنار هم استفاده می‌کنند و در واقع جهت سرعت بالاتر نمایش و بررسی داده‌ها از طریق محیط SQL Server از فیلد کاراکتری تاریخ خورشیدی و برای مقایسه و بدست آوردن ساعت از فیلد نوع DateTime استفاده می‌کنند.

از نظر فضای اشغال‌شده نوع DataTime، هشت بایت، smalldatetime (در صورت استفاده) 4 بایت و فیلد 10 کاراکتری تاریخ 10 بایت فضا اشغال می‌کند در صورتی که نوع JalaliDate با درنظر گرفتن همه‌ی مزایای انواع داده‌ی استفاده‌شده برای تاریخ، فقط 8 بایت فضا اشغال می‌کند. با استفاده از این نوع به راحتی داده‌ی تاریخ را بر اساس تقویم ایرانی اعتبارسنجی می‌کنید و بخش‌های مختلف زمان از سال تا ثانیه را با یک متد به دست می‌آورید. می‌توانید به راحتی به تاریخ خود زمانی را بیفزایید یا بکاهید و در گزارش‌ها بدون نگرانی از تبدیل درست استفاده کنید. چون کدباز است می‌توانید با کمی حوصله امکانات دیگر مد نظر خود را به آن بیفزایید و از آن در SQL بهره ببرید.

چگونه این نوع داده را حذف کنم!؟

شما می‌توانید به سادگی نوع داده‌ی ایجادشده توسط CLR را در مسیر زیر بیابید و اقدام به حذف آن نمایید:

همان‌طور که مشاهده می‌شود؛ حتی نوع داده‌ی سیستمی hierarchyid که جهت ساختار سلسله‌مراتبی مانند چارت سازمانی یا درخت تجهیزات استفاده می‌شود؛ نیز یک نوع داده‌ی CLR است.

آیا راه دیگری نیز برای افزودن این نوع داده به SQL به جز Publish کردن وجود دارد؟

مانند بسیاری دیگر از گونه‌های پروژه، در اینجا نیز شما یک فایل DLL خواهید داشت. این فایل برپایه‌ی تنظیماتی که شما در قسمت Properties پروژه‌ی خود انجام می‌دهید ساخته می‌شود. پس از تغییر مسیر فایل DLL در دستور زیر توسط یک New Query از Database خود، آن را اجرا کنید:

```
CREATE ASSEMBLY JalaliDate
FROM 'F:\prgJalaliDate.dll'
WITH PERMISSION_SET = SAFE;
```

هم‌چنین در صورت ویرایش‌های دوباره پروژه از دستور زیر استفاده کنید:

```
ALTER ASSEMBLY JalaliDate
FROM 'F:\prgJalaliDate.dll'
```

با استفاده از دستورهای زیر می‌توانید از چگونگی درج فایل‌های افزوده شده آگاه شوید:

```
select * from sys.assemblies
select * from sys.assembly_files
```

Results | Messages

	name	principal_id	assembly_id	clr_name	permission...	permission_set_desc	is_visible	create_date	modify_date	is_user_defined
1	Microsoft.SqlServer.Types	4	1	microsoft.sqlserver.types, ver...	3	UNSAFE_ACCESS	1	2012-02-10 20:...	2012-02-10 ...	0
2	prgJalaliDate	1	65536	prgjalalidate, version=0.0.0.0,...	1	SAFE_ACCESS	1	2013-04-29 22:...	2013-04-29 ...	1

	assembly_id	name	file_id	content
1	1	microsoft.sqlserver.types.dll	1	0x4D5A900003000000040000 00FFFF0000B80000000000000...
2	65536	prgJalaliDate	1	0x4D5A900003000000040000 00FFFF0000B80000000000000...
3	65536	prgJalaliDate.pdb	2	0x4D6963726F736F667420432F432B2B204D534620372E30...

تا اینجا SQL Server، دی‌ال‌ال مربوط به پروژه را شناخته است. برای تعریف نوع داده از دستور زیر بهره ببرید:

```
CREATE TYPE dbo.JalaliDate
EXTERNAL NAME JalaliDate.[JalaliDate];
```

این کار همانند استفاده از گزینه‌ی Publish در Visual Studio است.

هم‌چنین چنان‌چه در SQL Server 2012 از منوی راست‌کلیک پایگاه داده‌ها روی گزینه Tasks و سپس Generate Scripts را انتخاب کنیم، از مشاهده‌ی سند ساخته شده، درخواهیم یافت که حتی دستورهای مربوط به ساخت اسمبلی CLR با تبدیل فایل به کد در Scripts وجود دارد و با اجرای آن در سروری دیگر، انتقال می‌یابد.

```
GO
/****** Object:  SqlAssembly [prgJalaliDate]    Script Date: 2013/04/30 08:27:00 ب.ظ ******/
CREATE ASSEMBLY [prgJalaliDate]
FROM 0x4D5A900003000000040000 00FFFF0000B8000000000000 ..... بقیه‌ی کدها حذف شده
WITH PERMISSION_SET = SAFE

GO
ALTER ASSEMBLY [prgJalaliDate]
ADD FILE FROM 0x4D6963726F736F667420432F432B2B204D534620372E30300D0A1A44530..... بقیه‌ی کدها حذف شده
AS N'prgJalaliDate.pdb'

GO
/****** Object:  UserDefinedType [dbo].[JalaliDate]    Script Date: 2013/04/30 08:27:00 ب.ظ ******/
CREATE TYPE [dbo].[JalaliDate]
EXTERNAL NAME [prgJalaliDate].[JalaliDate]

GO
```

دنباله دارد ...

نظرات خوانندگان

نویسنده: محمد عادل
تاریخ: ۱۳۹۲/۰۲/۱۱ ۱:۴۳

در این حالت ، چطور میتونیم در EF Code First از این DataType استفاده کنیم ؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۱ ۸:۳۴

از برنامه Entity Framework Power Tools برای مهندسی معکوس ساختار موجود استفاده کنید و بعد مشاهده کنید که چه کدی رو تولید می‌کنه. کار شما در این حالت code first نیست.

نویسنده: حامد حسین نژاد
تاریخ: ۱۳۹۲/۰۲/۱۱ ۹:۲۰

البته باید این را هم در نظر گرفت که کلا CLR Integration باعث کندی دیتابیس می‌شود و انواع داده CLR بسیار کندتر از بقیه اجرا می‌شوند. این موضوع ممکن است که در دیتابیس‌های کوچک چندان مهم بنظر نیاید ولی در دیتابیس‌های بزرگ باعث بروز مشکل خواهد شد. علاوه بر این استفاده از انواع داده CLR و یا توابع CLR در دیتابیس باعث می‌شود که امکان پارتیشن بندی جداول آن دیتابیس وجود نداشته باشد.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۱ ۱۱:۳۴

مقاله‌ای هست اینجا در مورد کارآیی CLR در SQL Server. به نظر می‌رسه سریعتر است حدود 11 درصد نسبت به T-SQL معمولی. برای پارتشین بندی می‌تونید اینکار رو انجام بدید فقط این نوع خاص قابل انتخاب نیست. مابقی فیلدها رو می‌تونید انتخاب کنید.

نویسنده: حامد حسین نژاد
تاریخ: ۱۳۹۲/۰۲/۱۱ ۱۸:۳۱

البته کارایی CLR، بسته به مورد استفاده، متفاوته (این لینک). در مواردی مثل همین مثال، اگه تعداد سطور جدول زیاد باشه، کارایی رو به شدت کاهش میده. مخصوصا اگه بخواین از Data Warehousing هم استفاده کنید.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۱ ۱۹:۲۸

این به شدت رو می‌تونید در موردش عدد و رقم ارائه بدید؟ در همون لینکی که دادید عنوان شده پیاده سازی RegEx روی سطور بالا خیلی سریع‌تر هست با CLR. در هر حال بهتره با عدد و رقم و محاسبات بحث کرد.

نویسنده: حامد حسین نژاد
تاریخ: ۱۳۹۲/۰۲/۱۲ ۱۴:۲۸

مثلا فرض کنید جدولی دارید که اطلاعات Taskهای یک شرکت رو نگه میداره که شامل تاریخ شروع و تاریخ پایان هر Task هم هست. اگه این جدول یک میلیون سطر داده داشته باشه و بخوایم Task هایی که مدت زمان انجام اونها کمتر از 5 روز بوده رو انتخاب کنیم تفاوت فاحشی با datetime خواهیم داشت.
البته، همونطور که گفتم بسته به نوع استفاده داره و ممکنه از دیتابیسی به دیتابیس دیگه فرق بکنه.

عنوان: **افزودن یک DataType جدید برای نگه‌داری تاریخ خورشیدی - 3**
نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۳ ۱۳:۲۵
آدرس: www.dotnettips.info
برچسب‌ها: SQL Server, SQL Server 2012, Persian, CLR

در دو درس گذشته با DataType ای که با هم نوشتیم؛ می‌توانستیم این پرس‌وجوها را داشته باشیم:

```
Declare @MyDate JalaliDate = '1392/02/11 21:38:24'
Select @MyDate.ToString() as MyDateTime
           , @MyDate.GetDate() as MyDate
           , @MyDate.GetTime() as MyTime
           , @MyDate.Year as MyYear
           , @MyDate.Month as MyMonth
          , @MyDate.Day as MyDay
          , @MyDate.Hour as MyHour
          , @MyDate.Minute as MyMinute
          , @MyDate.Second as MySecond
          , @MyDate.JalaliDateAdd('Year',1) as NextYear
          , @MyDate.JalaliDateAdd('Month',1) as NextMonth
          , @MyDate.JalaliDateAdd('Day',1) as NextDay
          , @MyDate.JalaliDateAdd('Hour',1) as NextHour
          , @MyDate.JalaliDateAdd('Minute',1) as NextMinute
          , @MyDate.JalaliDateAdd('Second',1) as NextSecond
```

که در نهایت خروجی کار به شکل زیر خواهد بود:

Results | Messages

	MyDateTime	MyDate	MyTime	MyYear	MyMonth	MyDay	MyHour	MyMinute	MySecond
1	1392/02/11 21:38:24	1392/02/11	21:38:24	1392	2	11	21	38	24

NextYear	NextMonth	NextDay	NextHour	NextMinute	NextSecond
1393/02/11 21:38:24	1392/03/11 21:38:24	1392/02/12 21:38:24	1392/02/11 22:38:24	1392/02/11 21:39:24	1392/02/11 21:38:25

با این همه، همان‌سان که در پرس‌وجوی بالا هم می‌بینید؛ ناگزیر شدم تاریخ و زمان را خودم وارد کنم و هرآینه در واقعیت شما جهت پرس‌وجو روی زمان کنونی، ناگزیر به استفاده از یک تابع برای تبدیل تقویم میلادی به خورشیدی هستیم. به نظر شما بهتر نیست دست به کار شویم و تابعی برای تبدیل تاریخ میلادی به خورشیدی بنویسیم؟

برای این‌کار پروژه‌ای را که در دو درس 1 ساختیم باز کنید و سپس روی نام پروژه در Solution Explorer راست‌کلیک کرده و Add New Item را انتخاب کنید.

محتویات فایل بازشده را حذف کنید و دستورهای زیر را جای‌گزین کنید:

```
using System;
using System.Data;
using System.Data.SqlClient;
using System.Data.SqlTypes;
using Microsoft.SqlServer.Server;

public partial class UserDefinedFunctions
{
    [Microsoft.SqlServer.Server.SqlFunction]
    public static JalaliDate GetCurrentDateTime()
    {
        System.Globalization.PersianCalendar pers = new System.Globalization.PersianCalendar();
        DateTime CurrentDate = DateTime.Now;
        JalaliDate jl;
        jl.Year = (Int16)pers.GetYear(CurrentDate);
        jl.Month = (byte)pers.GetMonth(CurrentDate);
        jl.Day = (byte)pers.GetDayOfMonth(CurrentDate);
        jl.Hour = (byte)pers.GetHour(CurrentDate);
        jl.Minute = (byte)pers.GetMinute(CurrentDate);
        jl.Second = (byte)pers.GetSecond(CurrentDate);
        return jl;
    }
}
```

بار دیگر پروژه را به روشی که در درس یک آموزش داده شد؛ Publish کنید. اکنون می‌توانیم تاریخ و زمان کنونی را در متغیری از نوع JalaliDate وارد کنیم.

```
Declare @MyDate JalaliDate = dbo.GetCurrentDateTime()
```

یا به این شکل تاریخ و زمان جاری را مشاهده کنیم:

```
Select dbo.GetCurrentDateTime().ToString()
```

دنباله دارد...

نظرات خوانندگان

نویسنده: فرشید علی اکبری
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۰:۳۴

از دوستان اگه کسی نحوه استفاده این تکنیک رو توی EF Code First میدونه لطفا مطلب رو Share کنه تا امکان استفاده این دسته از عزیزان هم از زحمتی که آقای قنادی کشیدن باشه.
با تشکر.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۰:۴۲

```
public DateTime AddDate { set; get; }

[NotMapped]  // فیلد محاسباتی سمت برنامه که در بانک اطلاعاتی قرار نخواهد گرفت
public string PersianDate
{
    get { return MyDateConverter(AddDate); }
}
```

در code first از ویژگی NotMapped استفاده کنید تا بتونید با استفاده از تابع کمکی تبدیل تاریخ خودتون راحت این تبدیلات رو انجام بدید. در بانک اطلاعاتی DateTime استاندارد رو ذخیره کنید، در سمت کدها برای مسایل نمایشی از خاصیت PersianDate که NotMapped تعریف شده، میشه استفاده کرد. به علاوه روی DateTime استاندارد راحت میشه کوئری‌های LINQ رو اجرا کرد بدون محدودیت. زمانیکه مثلا یک List تهیه شد، قسمت بعدی، کار نمایشی است که از خاصیت NotMapped میشه کمک گرفت. این روش با تمام بانک‌های اطلاعاتی کار می‌کنه.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۱:۴۱

```
[Column(TypeName = "xml")]
public string XmlValue { get; set; }

[NotMapped]
public XElement XmlValueWrapper
{
    get { return XElement.Parse(XmlValue); }
    set { XmlValue = value.ToString(); }
}
```

روش عمومی کار با نوع‌های خاصی که در EF تعریف نشدن، استفاده از ویژگی Column و مشخص کردن Type آن است؛ مانند مثالی که در بالا ملاحظه می‌کنید. البته این نوع خاص، در سمت کدها باید به صورت رشته تعریف شود. مثلا از سال 2005 به این طرف فیلد XML به SQL Server اضافه شده. اما نمی‌شود ازش در EF به همون شکل XML استفاده کرد. باید تبدیلش کنی به String تا قابل استفاده بشه. یک نمونه دیگرش نوع خاص Spatial هست که در نگارش‌های اخیر SQL Server اضافه شده (geography و geometry). این مورد فقط از EF 5.0 به بعد پشتیبانی توکاری ازش ارائه شده. یا برای hierarchyID در EF معادلی وجود نداره. برای تعریف این مورد نیز در یک مدل باید از string استفاده کرد.

بعد اگر این نوع خاص (که الان به صورت رشته دریافت شده) قابل نگاشت به نوعی مشخص در سمت کدهای برنامه بود (یعنی صرفا یک رشته ساده نبود) مثلا می‌شود از ویژگی NotMapped برای تبدیل آن و تعریف آن به شکل یک فیلد محاسباتی استفاده کرد.

نویسنده: پرنیا
تاریخ: ۱۳۹۲/۰۷/۲۱ ۲۳:۱۶

با سلام
پروژه ای که شرح دادین رو ایجاد و در Publish ، SQL server 2012 کردم و در جدول هم مقادیر تستی درج کردم.
زمانی که جدول را در Object Browser با Mode ویرایش باز می‌کنم هیچ مشکلی وجود ندارد و داده‌ها درست نمایش داده می‌شوند اما زمانی که با دستورات T-SQL کار میکنم مقادیر را به صورت یک رشته از کاراکترها نمایش می‌دهد که نامفهوم می‌باشد.
تصویر اجرای کوئری‌ها را میذارم لطفا راهنمایی کنید.

	Id	TestDate
▶	۱	۱۳۹۲/۰۲/۰۹ ۰۰:۰۰:۰۰
	۲	۱۳۹۲/۰۲/۰۹ ۲۲:۴۰:۰۰
	۳	۱۳۹۲/۱۲/۲۹ ۲۲:۴۰:۰۰
*	NULL	NULL

```
select *
from TestTable
```

۱۰۰ %

Results | Messages

	Id	TestDate
۱	۱	۰x۸۵۷۰۰۲۰۹۰۰۰۰۰۰۰۰
۲	۲	۰x۸۵۷۰۰۲۰۹۱۶۲۸۰۰۰۰
۳	۳	۰x۸۵۷۰۰C۱D۱۶۲۸۰۰۰۰

```
select dbo.GetCurrentDateTime()
```

۱۰۰ %

Results | Messages

	(No column name)
۱	۰x۸۵۷۰۰۷۱۵۱۷۰A۳A۰۰

نویسنده: وحید نصیری

تاریخ: ۱۳۹۲/۰۷/۲۲ ۰:۴

SQLCLR types به صورت پیش فرض با فرمت serialized binary value بازگشت داده می‌شوند.

```
SELECT id, TestDate.ToString() FROM TestTable;
```

در ابزارهای کوئری گرفتن ad-hoc مثل SSMS باید یک متد ToString را هم به انتهای نام ستون اضافه کنید تا مقدار نمایشی واضحی حاصل شود.

عنوان: **آشنایی با FileTable در SQL Server 2012 بخش 1**
نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۶ ۲۰:۲۵
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, SQL Server 2012

پیش از آشنایی با FileTable نیاز است که پیشینه‌ای از شیوه‌های ذخیره‌سازی فایل و یا بهتر بگویم BLOB در SQL Server را داشته باشیم. نخستین شیوه‌ی نگه‌داری فایل استفاده از Image است که در SQL Server 2000 کاربرد داشت و هم‌اکنون استفاده از آن به دلیل کاهش بسیار کارآیی منسوخ‌شده است. به دلایل مشکلات بسیار فراوان Image هم‌زمان بسیاری از طراحان پایگاه داده‌ها، جهت کاهش حجم جدول‌ها و پیروی آن حجم پایگاه داده‌ها، فایل را در سیستم‌فایل نگه‌داری می‌کردند و تنها مسیر آن را در فیلدی از نوع کاراکتری در پایگاه‌داده‌ها ذخیره می‌کردند. این روش هرچند از حجم پایگاه داده‌ها می‌کاست ولی به دلیل عدم دخالت SQL Server در مدیریت فایل‌ها مشکلات دیگری را به وجود آورد.

از SQL Server 2005 نوع داده‌ی varbinary(max) معرفی شد که برخی از چالش‌های به‌کاربری Image را کاست و درباره‌ی بسیاری از موارد مانند ذخیره‌ی عکس پرسنلی هنوز هم کاربرد دارد؛ ولی توجه داشته باشید که استفاده از این فیلد فقط برای فایل‌های کم‌تر از 256 کیلوبایت سفارش شده است و برای بالاتر از آن، کارآیی کاهش فراوانی خواهد یافت.

در SQL Server 2008 نوع داده‌ی جدیدی به نام FileStream به وجود آمد به این شکل که یک FileGroup از نوع Data FileStream به پایگاه‌داده افزوده می‌شود و در واقع با یک پوشه در سیستم فایل در پیوند است. از این پس هنگام ساخت یک جدول به جای استفاده از نوع داده‌ی varbinary از نوع FileStream استفاده می‌کنیم با مد نظر داشتن این نکته که حتماً باید یک فیلد از نوع Uniqueidentifier هم در آن جدول تعریف شده باشد. شیوه‌ی کار نیز به این صورت خواهد بود که خود رکورد در جدول ذخیره می‌شود و فقط محتوای فایل در آن مسیری از NTFS ذخیره می‌شود. برخلاف روش درج مسیر فایل در جدول که پس از حذف رکورد، فایل هم‌چنان در سیستم فایل می‌ماند؛ این بار با حذف رکورد فایل مربوطه نیز حذف خواهد شد. افزون بر این مدیریت پشتیبانی از فایل‌ها نیز برعهده‌ی پایگاه داده‌ها خواهد بود. اندازه‌ی فایل‌ها در FileStream محدودیت‌های پیشین را نخواهد داشت و شما به اندازه‌ی حجم درایو هارددیسک می‌توانید فایل در آن ذخیره کنید. نکته‌ی دیگر درباره‌ی فایل‌های با حجم سنگین که می‌توانید Stream مربوط به یک فایل را به صورت بخش‌بخش در سمت مشتری بارگذاری کنید و به او نشان دهید. در FileStream امنیت و تراکنش فایل‌ها برعهده‌ی SQL Server است و از این دیدگاه بسیار ساده‌تر و کارآتر از FileSystem است.

(برای آشنایی بیشتر با FileStream، این نوشتار از مهندس وحید نصیری را مطالعه کنید.)

گونه‌ی FileTable از ویژگی‌های نوین SQL Server 2012 است که تکمیل‌کننده‌ی FileStream است. FileTable آمیزشی از FileStream با hierarchyid و سیستم فایل ویندوز برای ارائه‌ی توانایی‌های نوین مدیریت BLOB در SQL Server است. FileTable همان‌گونه که از دو واژه‌ی تشکیل‌دهنده‌اش پیداست؛ هم‌زمان یک جدول و یک سیستم فایل معمولی است.

FileTable به هر روی یک جدول از پایگاه‌داده‌های SQL Server است با یک تفاوت که ساختار آن از پیش تعریف‌شده است. ستون‌های FileTable و نوع داده‌ی آن از پیش توسط SQL Server مشخص شده است. ستون‌های تشکیل‌دهنده‌ی FileTable دربرگیرنده‌ی جدول زیر است:

نام ستون	DataType	شرح
stream_id	Uniqueidentifier ROWGUIDCOL	شناسه‌ی یکتای ردیف
file_stream	varbinary(max) FILESTREAM	محتوای فایل، اگر پوشه باشد؛ Null است.
name	nvarchar(255)	نام فایل یا پوشه
path_locator	hierarchyid	جای فایل یا پوشه در سیستم‌فایل
creation_time	datetimeoffset(7)	زمان ایجاد
last_write_time	datetimeoffset(7)	زمان واپسین ویرایش
ast_access_time	datetimeoffset(7)	زمان واپسین دسترسی
is_directory	bit	۰ = فایل / ۱ = پوشه
is_offline is_hidden is_readonly is_archive is_system is_temporary	bit	

هر ردیف از FileTable نماینده‌ی یک فایل یا پوشه در File System است. ستون path_locator که از نوع hierarchyid است نشان‌دهنده‌ی مسیر یک فایل یا پوشه است. hierarchyid که از SQL Server 2008 معرفی شده است؛ بهترین نوع داده برای نگه‌داری ارتباط ساختار سلسله‌مراتبی مانند چارت سازمانی، درخت تجهیزات یک کارخانه و یا در همین نمونه درخت فایل‌ها و پوشه‌ها است. پس می‌توانیم از همه‌ی امکانات hierarchyid در FileTable نیز برخوردار شویم. این‌که این فایل به ترتیب در چه پوشه‌هایی قرار گرفته است یا این‌که این پوشه شامل چه فایل‌ها یا پوشه‌هایی خواهد بود. این‌که پوشه‌های هم‌فرزند پوشه‌ی جاری کدام است و یا یا توابع مربوط به جابه‌جایی فایل‌ها و پوشه‌ها.

دنباله دارد...

نظرات خوانندگان

نویسنده: امیر بختیاری
تاریخ: ۱۳۹۲/۰۲/۱۷ ۱۶:۲۶

با سلام و خسته نباشید
مطلب خیلی خوبی بود فقط اگر امکان دارد در یک بخش درباری hierarchyid نیز کامل به همراه مثال توضیح دهید
با تشکر

نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۷ ۱۷:۲۷

درود بر شما
پیشنهاد خوبی است. پیش از این نوشتاری در این باره نوشته بودم (هرچند منتشر نکرده ام.) ولی با یادداشت شما به این اندیشه افتادم که مروری بر این بحث در SQL Server 2012 داشته باشم و با ویرایشی نوین، در این تارنما منتشر کنم.

نویسنده: ali.rezayee
تاریخ: ۱۳۹۲/۰۲/۱۷ ۱۹:۳۱

با سلام و تشکر از این مطلب خوب.
آیا نوع فیلدها قابل تغییر است؟
برای مثال در فیلد Name چون varchar است نمیتوان نام فارسی برای فایل ثبت کرد، آیا امکان تغییر آن به Nvarchar هست؟
ممنون از شما.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۷ ۲۲:۵۸

در اصل nvarchar هست که به نظر در جدول فوق باید اصلاح بشه.

نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۸ ۶:۵۳

بله. اشتباه از من بود. این فیلدها غیرقابل تغییر هستند ولی هیچ مشکلی هم با زبان پارسی ندارند.

نویسنده: سید امیر سجادی
تاریخ: ۱۳۹۲/۰۹/۱۴ ۲۰:۱۹

سلام. ممنون از مطلب مفیدتون. فقط یک مشکلی هست و اون عدم پشتیبانی توسط Linq2Sql هست. فکر کنم فقط با DataSet مشکلی نداشته باشه و اون هم بخاطر این هست که نوع hierarchyid به نوع nvachar(867) تبدیل میکنه.
لطفا اگه برای این مشکل راه حلی هست بنویسید که دوستان هنگام استفاده به مشکل برنخورند

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۹/۱۴ ۲۰:۳۱

Linq2Sql که خیلی وقت هست با EF جایگزین شده. EF هم از نوع‌های خاص دیتابیس‌ها عموما پشتیبانی نمی‌کنه. بحث EF هم نیست. اکثر ORMهای موجود همینطور هستند.

نویسنده: سید امیر سجادی

تاریخ: ۱۳۹۲/۰۹/۱۴ ۲۳:۸

یعنی فقط باید از DataSet استفاده کرد !؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۹/۱۵ ۱:۱

ADO.NET می‌تونه خروجی از نوع DataSet هم بده؛ ولی اسمش DataSet نیست. تمام ORMها هم برفراز ADO.NET کار می‌کنند. بنابراین اگر تصور کنیم که ابدا نمی‌شود با این‌ها کار کرد، خیر. اینطور نیست. شما با ORMها هم می‌تونی مستقیما SQL بنویسی ef-code-first-13 تا به قابلیت‌هایی دسترسی پیدا کنی که در ORM پیش بینی نشده.

نویسنده: سعید
تاریخ: ۱۳۹۲/۱۰/۱۴ ۲۳:۵۱

سلام
چطور امنیت فایل‌ها رو برقرار کنیم ؟ منظورم اینه که کسی نتونه اونا رو پاک یا مشاهده و یا اضافه کنه ، مگه اینکه ما بهش اجازه بدیم
راهی برای این کار هست ؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۵ ۰:۴۵

مگه کاربران شما مستقیما با فایل سرور کار می‌کنند؟ روال عادی آن کار از طریق یک برنامه مثلا وب هست. امنیت رو در اونجا پیاده سازی کنید.

نویسنده: سعید
تاریخ: ۱۳۹۲/۱۰/۲۲ ۱۱:۱۱

من دارم از اپلیکیشن استفاده میکنم . از داخل برنامه درست هست ، یعنی همونی که شما میگی . ولی اگه کسی از بیرون برنامه بخواد این کارو کنه چی ؟
مستقیم بره تو پوشه و فایل‌ها رو تغییر بده . من میخوام بدونم sql میتونه جلوی این کارو هم بگیره ؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۲۲ ۱۲:۱۲

- به چه نحوی از SQL Server استفاده می‌کنید؟ آیا سرور و برنامه دسکتاپ شما روی یک کامپیوتر هستند؟ برای اینکار بهتر است از SQL CE یا SQLite استفاده کنید؛ یا حتی LocalDB. هدف از SQL Server نصب آن روی یک سرور و خدمات دهی به چندین و چند کامپیوتر تحت شبکه است. برای استفاده روی یک کامپیوتر یعنی کسب و کار کوچک و عملا نیازی به SQL Server 2012 ندارد اینکار. زندگی مصرف کننده را سخت نکنید. نصب و نگهداری یک سرور کار هر شخصی نیست و برای سازمان‌ها طراحی شده و نه مصارف کوچک تک کاربره دسکتاپ.

- با این توضیح اگر کسی به سرور شما دسترسی دارد، آیا نمی‌تواند مثلا اگر فایل‌ها در دیتابیس ذخیره می‌شدند، اون‌ها رو دستی با یک کوئری حذف کند؟ امنیت کار با سرور این‌جا است که مطرح می‌شود و همچنین اطمینان به ادمین‌ها.

- در مورد امنیت file table مراجعه کنید به مستندات مایکروسافت. مثلا: FileTables are secured by SQL Server security only

عنوان: آشنایی با FileTable در SQL Server 2012 بخش 2
نویسنده: حامد قنادی
تاریخ: ۱۳۹۲/۰۲/۱۸ ۱۵:۳۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server, SQL Server 2012

ستون دیگر **stream_id** نام دارد که از نوع uniqueidentifier ROWGUIDCOL است. همان‌گونه که در یاد دارید، در FileStream نیز ناگزیر به تعریف چنین ستونی بودیم. بنابراین FileTable استثناء نیست و در این‌جا نیست چنین فیلدی توسط SQL Server تعریف می‌شود. اگر فایل‌ها و پوشه‌ها جابه‌جا نمی‌شدند می‌توانستید از هر دو ستون path_locator یا stream_id برای شناسایی یک رکورد از جدول بهره ببرید. ولی با جابه‌جایی یک فایل و یا به عبارت دیگر تغییر پدر آن در ساختار سلسله‌مراتبی، مقدار path_locator نیز تغییر می‌کند، پس ناگزیر به استفاده از این ستون برای ارجاع به یک ردیف در جدول هستیم.
هر ردیف از جدول نمایان‌گر یک فایل یا پوشه است، بنابراین به ستونی نیاز داریم که بتوانیم این موضوع را نشان دهیم. بر این پایه از ستون **is_directory** بهره می‌بریم که 1 بودن آن نشان‌دهنده‌ی این است که این ردیف از جدول به یک پوشه ارجاع دارد.
نام فایل یا پوشه در ستونی به نام **name** نگه‌داری می‌شود که رشته‌ای از نوع nvarchar(255) است. افزون بر این ستون، ستون‌های دیگری نیز در این جدول وجود دارد که ویژگی‌های یک فایل مانند پنهان‌بودن، فقط‌خواندنی و ... توسط آن توسط آن به دست می‌آید. ستون پسین **file_stream** نام دارد که برای پوشه‌ها، محتوای آن Null است. علت آن این است که محتوای واقعی فایل در این ستون نگه‌داری می‌شود. در واقع یک varbinary(max) با ویژگی‌های fileStream است که محتوای باینری آن در سیستم فایل NTFS ذخیره می‌شود. مدیریت پشت صحنه‌ی این ستون برعهده‌ی SQL Server است.
افزون بر این 14 ستون، هر FileTable شامل سه ستون محاسباتی به شرح زیر است:

نام ستون	DataType	شرح
parent_path_locator	hierarchyid	مسیر به‌دست‌آمده از path_locator
file_type	nvarchar(255)	گونه‌ی فایل به دست‌آمده از name
cached_file_size	bigint	اندازه‌ی فایل به دست‌آمده از file_stream

ستون parent_path_locator نتیجه‌ی فراخوانی تابع GetAncestor(1) در ستون path_locator است که جهت به دست آوردن پوشه‌ی پدر یک فایل و پوشه استفاده می‌شود. ستون file_type که از مقدار رشته‌ای ستون name تجزیه شده است، پسوند فایل را برمی‌گرداند. و ستون cached_file_size اندازه‌ی بایت ذخیره‌شده ستون file_stream را برمی‌گرداند. با این ساختار ثابت در اینجا، هر FileTable هر آن‌چه از File System نیاز دارید در یک پوشه‌ی اشتراکی به شما می‌دهد.

این یعنی نمایش بی‌واسطه FileTable به هر کاربر یا برنامه. به طوری که برای نمایش یا به‌روزرسانی جدول می‌توانید از روش استاندارد I/O مانند کشیدن و رهاکردن با Windows Explorer یا برنامه‌نویسی با System.IO.FileStream و APIهای ویندوز استفاده کنید. این‌چنین:

- ایجاد یک فایل یا پوشه در سیستم فایل -> افزودن یک ردیف به جدول

- افزودن یک ردیف به جدول -> ایجاد یک فایل یا پوشه در سیستم فایل

با کپی فایل‌ها در مسیر بالا، به صورت خودکار رکوردهای زیر در جدول PhotoTable در پایگاه‌داده‌ها افزوده می‌شود:

	stream_id	file_stream	name	path_locator	parent_pa
1	FABBA0DA-A1B7-E211-...	0x89504E470D0A1A0A0000000D49484452...	Pic13920218.png	0xFF1E127B0A0A9B2FDDD353FFA0403...	NULL
2	A0CB07E7-A1B7-E211-...	0xFFD8FFE000104A4649460001020100600...	Hydrangeas.jpg	0xFECCB4A9F110A38FD391534845521...	NULL
3	A2CB07E7-A1B7-E211-...	0xFFD8FFE000104A4649460001020100600...	Chrysanthemum.jpg	0xFD64A656CE5A91CFDB125242D522...	NULL
4	A4CB07E7-A1B7-E211-...	0xFFD8FFE000104A4649460001020100600...	Desert.jpg	0xFF6A01070D9AB3EFC61352DE20FB1...	NULL
5	3DCCC316-A2B7-E211-...	NULL	Archives-92-02	0xFD00916BA6D2930FCFD2351E24281...	NULL
6	63B6DF2A-A2B7-E211-...	0xFFD8FFE000104A4649460001020100600...	Tulips.jpg	0xFD00916BA6D2930FCFD2351E24281...	0xFD009

به طور خلاصه پیش از این برای افزودن به FileStream دو راه کار پیش رو داشتید. یکی استفاده از T-SQL و دیگر sqlFileStream اکنون SQL Server 2012 راه کار سوم را پیشنهاد می‌کند. استفاده از File System در این روش FileStream به طور خودکار پر می‌شود.

پیش از ساخت یک FileTable بیان این نکته دارای اهمیت است که با کپی فایل‌ها و پوشه‌ها هیچ چیز جدیدی به NTFS افزوده نمی‌شود بلکه محتوای فایل به FileStream افزوده می‌شود و SQL Server با بررسی همزمان FileStream و FileTable نمایشی از ردیف‌های FileTable به صورت یک پوشه‌ی اشتراکی نشان می‌دهد. این نکته پاسخی به این پرسش خواهد بود که آیا با استفاده از FileTable حجم پایگاه‌داده‌ها دو برابر خواهد شد و در نتیجه دشواری‌ها و چالش‌های نگه‌داری و پشتیبانی را پیش رو خواهیم داشت!؟ که پاسخ "خیر" خواهد بود.

ایجاد یک FileTable

پیش از این در همین تارنما، روش فعال کردن FileStream در SQL Server را آموزش دیده اید. اگر درست به خاطر داشته باشید، چیزی شبیه به دستورهای زیر بود:

```
CREATE DATABASE MyFileArchive
ON PRIMARY
(NAME = MyFileArchive_data,
FILENAME = 'C:\Demo\MyFileArchive_data.mdf'),
FILEGROUP FileStreamGroup CONTAINS FILESTREAM
(NAME = PhotoFileLibrary_blobs,
FILENAME = 'C:\Demo\MyFiles')
LOG ON
(NAME = PhotoFileLibrary_log,
FILENAME = 'C:\Demo\MyFileArchive_log.ldf')
```

FileTable به FileStream متکی است؛ بر این پایه پیش از ایجاد یک FileTable باید FileStream را روی پایگاه‌داده‌ها فعال کنیم. این کار با یک تعریف درست توسط بند FILEGROUP...CONTAINS FILESTREAM انجام می‌شود.

برای ایجاد FileTable تنها کافی است که بند WITH FILESTREAM را به دستور CREATE DATABASE بیفزایید. (یا برای فعال‌کردن FileTable روی یک پایگاه‌داده‌ی ساخته شده بند SET FILESTREAM را در دستور ALTER DATABASE بنویسید.) در این بند، از DIRECTORY_NAME برای نام‌گذاری یک پوشه برای پایگاه‌داده‌ها استفاده می‌کنیم. این پوشه در یک پوشه ریشه به نام SQL Server instance نمایش داده خواهد شد. بخش دوم بند NON_TRANSACTED_ACCESS=FULL است که دسترسی غیرتراکنشی را فعال می‌کند. با این کار برای هر FileTable در پایگاه داده یک زیرپوشه درون پوشه‌ای که به نام DIRECTORY_NAME نام‌گذاری شده است؛ ساخته می‌شود.

با توجه به آنچه گفته شد برای ایجاد یک پایگاه‌داده با امکان ساخت FileTable دستورهای زیر را اجرا کنید:

```
CREATE DATABASE MyFileArchive
ON PRIMARY
(NAME = MyFileArchive_data,
FILENAME = 'C:\Demo\MyFileArchive_data.mdf'),
FILEGROUP FileStreamGroup CONTAINS FILESTREAM
(NAME = PhotoFileLibrary_blobs,
FILENAME = 'C:\Demo\MyFiles')
LOG ON
(NAME = PhotoFileLibrary_log,
FILENAME = 'C:\Demo\MyFileArchive_log.ldf')
WITH FILESTREAM
(DIRECTORY_NAME='FilesLibrary',
NON_TRANSACTED_ACCESS=FULL)
```

اکنون برای ساخت یک FileTable درون این پایگاه‌داده‌ها از دستور زیر استفاده کنید:

```
USE MyFileArchive
GO
CREATE TABLE PhotoTable AS FileTable
GO
```

توجه داشته باشید که چون ستون‌های FileTable از پیش تعریف شده است؛ ایجاد آن فقط با نوشتن دستور امکان پذیر است و مانند یک Table عادی از محیط کاربری SQL Server نمی‌توان بهره برد.
در Object Explorer از گره‌ی Tables، گره‌ی FileTables را باز کنید و روی جدولی که هم‌اکنون ساختیم راست‌کلیک کنید. با انتخاب گزینه‌ی Explore FileTable Directory پنجره‌ی زیر بازمی‌شود:

\\041-computer-35\mssqlserver\FilesLibrary
Search FilesLibrary
Organize
New folder
DVD Drive (G:)
CD Drive (H:) CD_ROM
Network
041-COMPUTER-35
mssqlserver
FilesLibrary
PhotoTable
Archives-92-02
SQLFileStream
Name
Date modified
Type
Size
PhotoTable
File folder
1 item

دنباله دارد ...

نظرات خوانندگان

نویسنده: ali.rezayee
تاریخ: ۱۳۹۲/۰۳/۰۴ ۱۶:۳۶

با سلام
موضوع بسیار جالب و مهمی است، لطفاً سریع‌تر ادامه دهید.

نویسنده: علی رضایی
تاریخ: ۱۳۹۳/۰۴/۲۷ ۱۹:۵۸

بهروش‌های استفاده از FILESTREAM

نویسنده: M.Shakeri
تاریخ: ۱۳۹۳/۰۵/۰۸ ۱۰:۱۴

فکر می‌کنم برای افزودن این قابلیت به دیتابیس دسترسی به دیتابیس master نیاز باشه که هاست‌ها این امکان رو نمیدن.

نویسنده: محسن خان
تاریخ: ۱۳۹۳/۰۵/۰۸ ۱۲:۸

خیلی از قابلیت‌ها مخصوص سازمان‌ها طراحی می‌شن.

نویسنده: Hamidbp
تاریخ: ۱۳۹۳/۰۷/۱۴ ۱۴:۴۰

می خواستم بدونم چطور میشه یک فایل رو که در Filetable ذخیره شده، به اصطلاح Fetch کنم و توسط برنامه مربوطه (مثلا adobe acrobat یا word یا ...) نشونش بدم؟

عنوان: **آموزش MDX Query - قسمت پنجم - باز کردن یک پایگاه داده ی Multidimensional در محیط BIMS و ساخت یک پروژه ی جدید.**
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۱ ۱۳:۳۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

در این قسمت در ابتدا نحوه‌ی باز کردن یک پایگاه داه‌ی چند بعدی را در محیط BIMS بررسی کرده و سپس چگونگی ساخت یک MDB را از پایه بررسی می‌کنیم. برای ادامه دادن این قسمت نیاز می‌باشد که پایگاه داده‌ی AdventureWorkDW2008 را در SSAS نصب کرده باشید .

در ابتدا مطابق شکل زیر منوی File سپس زیر منوی Open و Analysis Service Database را انتخاب نمایید.

در ادامه می‌بایست نام Server را مشخص نمایید و دقت داشته باشید که در اینجا منظور از نام سرور، نام سرور SSAS می‌باشد (در صورتیکه بر روی خود سرور در حال کار می‌باشید از . به جای نام سرور استفاده کنید). سپس در قسمت Database ، نام پایگاه داده‌ی چند بعدی را انتخاب نمایید. در صورتی که به جز Adventure Work DW 2008 ، پایگاه داده‌های چند بعدی دیگری را در SSAS داشته باشید، یک لیست از آنها را مشاهده خواهید کرد و در صورتیکه لیست شما خالی می‌باشد، احتمال دارد نام سرور اشتباه باشد یا روی سرویس SSAS مربوط به آن سرور هیچ پایگاه داده‌ی چند بعدی نصب نباشد.

حال مسیری را برای ذخیره سازی پروژه‌ی جدید در نظر بگیرید:

پس از کمی شکیبایی، واکشی اطلاعات از روی پایگاه داده‌ی چند بعدی انتخاب شده انجام می‌شود و یک پروژه در ارتباط با آن پایگاه داده ساخته می‌شود.

Solution Explorer
Solution 'Adventure Works DW 2008R2(PCCLIENT)' (1 pr
Adventure Works DW 2008R2(PCCLIENT)
Data Sources
Adventure Works DW
Data Source Views
Adventure Works DW
Clustered Customers
Subcategory Basket Analysis
Cubes
Adventure Works
Mined Customers
Dimensions
Promotion
Product
Customer
Geography
Reseller
Sales Territory
Employee
Scenario
Department
Organization
Account
Date
Source Currency
Sales Reason
Reseller Sales Order Details
Internet Sales Order Details
Clustered Customers
Subcategory Basket Analysis
Sales Channel
Destination Currency
Sales Summary Order Details
Mining Structures

همان طور که مشخص می‌باشد، یک شیء درون شاخه‌ی Data Source وجود دارد که مشخص کننده‌ی ارتباط این پروژه با پایگاه داده‌ی Data Warehouse است. برای مشاهده‌ی این ارتباط، بر روی Adventure Work DW کلیک راست کنید و سپس گزینه‌ی Open را انتخاب نمایید. در ادامه گزینه‌ی Edit را بزنید.

سپس در پنجره‌ی جدید، تنظیمات رشته‌ی ارتباطی با DW را مشاهده نمایید

با زدن کلید Test Connection باید پیام Test Connection Succeeded را مشاهده نمایید. اکنون پنجره‌ها را با زدن کلید OK ببندید.

در قسمت Data Source View سه شی تعریف شده است؛ براساس دسته بندی مورد نظر و جاری در Business موجود در Adventure Work .

با کلیک راست کردن بر روی Adventure Works DW و انتخاب گزینه‌ی Open ، اقدام به باز کردن DSV انتخاب شده کنید. در صفحه‌ی باز شده می‌توانید انواع دیاگرام تهیه شده را مشاهده نمایید و همچنین لیستی از جداول موجود در این DSV مشخص می‌باشد.

با کلیک راست کردن در فضای خالی دیاگرام ، امکان Add/Remove کردن جداول را به دیاگرام دارید.

در شکل بالا بعد از انتخاب یک جدول در سمت راست و انتقال آن به سمت چپ می‌توانید با زدن دکمه‌ی Add Related Table براساس کلیدهای خارجی، جداول مرتبط با جدول انتخاب شده را به صورت خودکار انتخاب نمایید و به قسمت چپ انتقال دهید.

شما در ساخت Cube مشخص می‌نمایید که Cube را از کدام DSV خواهید ساخت. بنابراین انتخاب جداول در DSV ها می‌بایست براساس نوع Business شما باشد تا در ساخت Cube به مشکلی برخورد نکنید.

در ساختار درختی موجود در پنجره‌ی Solution در شاخه‌ی Cube ، می‌توانید Adventure Works را باز کنید (کلیک راست و انتخاب Open) .

در شکل بالا در سمت چپ، می‌توانید Measure ها و Dimension های موجود در این Cube را مشاهده کنید. همچنین در قسمت بالا چندین Tab وجود دارند که در هر کدام تنظیمات بیشتری را بر روی Cube اعمال می‌کنیم. با توجه به اینکه طراحی Cube ها کاری تخصصی می‌باشد و نیاز به اطلاعات زیادی دارد اجازه دهید مقاله ای در خصوص طراحی Cube در SSAS جداگانه انتشار داده شود و فعلا در همین حد بسنده کنیم. با این حال در صورت نیاز می‌توانید برای اطلاعات بیشتر در این خصوص کتاب Microsoft SQL Server Analysis Services 2008 With MDX از انتشارات Wrox را مطالعه نمایید.

در Solution Explorer در شاخه‌ی Dimensions، می‌توانید تمامی بعدهایی که در تمامی Cube های شما استفاده شده‌اند را مشاهده نمایید.

با انتخاب یک بعد (ترجیحا بعد Date) و با کلیک راست کردن و انتخاب گزینه‌ی Open آن را باز نمایید.

در پنجره‌ی باز شده می‌توانید 4 Tab در بالا را مشاهده نمایید و در Tab نخست، Attribute ها و همچنین ساختار Hierarchies و در آخر Data source View را مشاهده نمایید.

در Attribute relationships می توانید ارتباط صفت‌های یک بعد را مشخص نمایید.

در Browsing Tab می‌توانید محتوای Dimension را بررسی نمایید (البته اگر در پروژه‌ی جدید قرار دارید حتما می‌بایست پروژه را Deploy کرده باشید. در حالتیکه یک پایگاه داه‌ی چند بعدی را باز می‌کنید، نیازی به Deploy کردن نمی‌باشد؛ زیرا حتما قبلا این کار انجام شده است (زیرا شما پایگاه داده‌ی چند بعدی را بعد از Deploy کردن پروژه‌ی SSAS خواهید داشت))

در صورتیکه مانند روش بالا یک پایگاه داده‌ی چند بعدی را باز کنیم، دیگر نیازی به Deploy کردن نمی‌باشد و فقط برای اعمال تغییرات روی پایگاه داده‌ی چند بعدی باید پروژه را Process کنیم و برای این منظور روی نام پروژه کلیک راست کرده و گزینه‌ی Process را انتخاب کنید. با این کار تغییرات اعمال شده در BIMS روی پایگاه داده‌ی SSAS اعمال می‌گردند و داده‌ها با توجه به ساختار Cube ها دوباره پردازش می‌شوند.

برای ساخت یک پروژه‌ی جدید به شکل زیر عمل می‌کنیم :

در ابتدا BIMS را باز کرده و سپس به منوی File رفته و در قسمت New گزینه‌ی Project را انتخاب می‌کنیم. سپس در صفحه‌ی باز شده، مطابق شکل زیر عمل کرده و یک پروژه از نوع Analysis Service Multidimensional ... می‌سازیم.

سپس برروی شاخه‌ی Data Source کلیک راست کرده و گزینه‌ی New Data Source را می‌زنیم و پنجره‌های ویزارد را به جلو می‌رویم.

در ابتدا باید یک Connection به DW تولید کنیم. برای این منظور در پنجره‌ی فوق دکمه‌ی New را زده و اطلاعات را مطابق شکل زیر پر می‌کنیم.

و سپس OK را میزنیم.

در صورتی که SSAS در یک سرور دیگر نصب شده است در پنجره‌ی بعدی نیاز می‌باشد نام کاربری را که به سرویس SSAS در آن سرور دسترسی دارد را وارد کنیم.

در صورتی که SSAS روی سیستم Local نصب شده است و کاربری که با آن Login هستیم دسترسی کافی به SSAS را دارد، گزینه‌ی Use the credentials of the current user را انتخاب می‌کنیم.

در صفحه‌ی آخر یک نام برای DS انتخاب می‌کنیم.

سپس نیاز می‌باشد یک DSV بسازیم. برای این منظور روی شاخه‌ی Data Source View کلیک راست کرده و گزینه‌ی New را انتخاب کرده و سپس در پنجره‌ی Wizard باید Data Source ساخته شده در مرحله‌ی قبل را انتخاب کرده و سپس Next را بزنیم. در اینجا بر اساس بیزینس‌های مختلف، راه کارهای گوناگونی را داریم. به عبارت دیگر می‌توان جداول Fact و Dimension های مرتبط با آن‌را بر اساس زیر سیستم‌های مختلف انتخاب کرده و برای هر کدام از آنها یک DSV بسازیم . به نظر من می‌توانیم تمامی جداول را در این مرحله انتخاب کرده و سپس این تفکیک بندی را در سطح Cube ها انجام داد. به طور کلی دقت داشته باشید به هیچ عنوان DSV و Cube های سیستم را خیلی تفکیک نکنید. زیرا در نوشتن کوئری‌ها و Join بین Cube ها با مشکل و سختی روبرو خواهید شد. (از لحاظ تجربی تفکیک بندی به شرطی صورت گیرد که نیازی به Join کردن Cube ها در MDX Query ها نباشد.)

سپس یک نام برای DSV خود انتخاب کرده و Finish را بزنید.

خوب؛ آخرین مرحله ساخت Cube می‌باشد (البته در طراحی Cube مطالب بسیاری وجود دارند که در یک مقاله‌ی دیگر تلاش خواهم کرد تمامی آن موارد را توضیح دهم.)

برای ساخت Cube ، روی شاخه‌ی Cube کلیک راست کرده و گزینه‌ی New را بزنید.

سپس Use Existing Table را انتخاب کرده و Next را بزنید.

در پنجره‌ی بعدی باید DSV را انتخاب کرد و بعد جداول مورد نیاز در طراحی Cube را انتخاب کنید. فراموش نکنید در صورت انتخاب یک Fact تمامی Dimension های مرتبط با آن را انتخاب نماید. دکمه Next را بزنید.

در پنجره‌ی بعدی باید جداول Fact را انتخاب کرده و دکمه‌ی Next را بزنید.

سپس در پنجره‌ی بعدی دایمنشن را انتخاب نمایید. (ترجیحا اجازه بدهید خود BIMS برای شما Dimension ها را بسازد، هرچند که خود شما می‌توانید بعدا به صورت دستی Dimension ها را ایجاد کنید).

بعد از زدن دکمه‌ی Next نامی برای Cube خود انتخاب نمایید و سپس دکمه‌ی Finish را بزنید.

بعد از ساخت Cube ، چندین دایمنشن به صورت خودکار ساخته می‌شوند . البته گاهی نیاز می‌باشد که اقدام به ساخت ساختارهای سلسله مراتبی در Dimension ها کنیم (این مورد را در یک مقاله جداگانه آموزش خواهم داد.)

پروژه با کلیدهای ترکیبی Ctrl+Shift+B ساخته می‌شود و بعد از اطمینان از درست بودن ساخت پروژه، آن را باید Deploy کرد.

برای Deploy کردن یک پروژه کافی است بعد از تنظیم کردن رشته‌ی ارتباطی در DS (قبلا توضیح داده شده است) روی پروژه کلیک راست کرده و گزینه‌ی Deploy را بزنیم.

نظرات خوانندگان

نویسنده: باغبان
تاریخ: ۱۳۹۲/۱۰/۰۱ ۲۱:۵۶

ممنون دوست عزیز
میشه بفرمایید چه موقع باید از این روش استفاده کرد و مزایای اون چی هستش

نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۱ ۲۳:۵۳

متاسفانه سوال شما برایم مشخص نمی‌باشد؟!؟
{ چه موقع باید از این روش استفاده کرد }
با این وجود در این مقاله من دو روش برای کار با پایگاه داده‌ی چند بعدی را توضیح دادم. ابتدا زمانی که یک پایگاه داده وجود دارد و ما می‌خواهیم در ساختار Cubeها و ... در آن تغییر ایجاد کنیم (تغییرات باید درون محیط BIMS انجام شود) در چنین شرایطی با توجه به مواردی که توضیح داده شده پایگاه داده را در محیط BIMS باز کرده و تغییرات را انجام داده و سپس Process می‌کنیم.
اما گاهی برای شروع کار نیاز می‌باشد طراحی MDB را از روی DW از ابتدا انجام دهیم. خوب در اینحالت چیزی برای باز کردن در BIMS وجود ندارد و باید خودمان از ابتدا آن را پیاده سازی کنیم.
راحت‌تر بگم تقریبا مثل اینه که شما بخواهید یک برنامه با C# از ابتدا بنویسید یا اینکه یک برنامه‌ی موجود را باز کنید و تغییراتی در آن ایجاد کنید و سپس ذخیره کنید.
امید وارم پاسخ سوال شما داده شده باشد.

عنوان: آموزش MDX Query - قسمت هفتم – استفاده از Pivot ، به کارگیری از ساختار های سلسله مراتبی و به کارگیری Cross Join در کوئری ها
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۲ ۴:۳۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

انجام عملیات Pivot توسط MDX :

برای این منظور کافی است فقط جای سطر و ستون را با هم عوض کنیم.برای مثال، کوئری‌های زیر را اجرا نمایید.

```
Select
[Date].[Calendar].[Calendar Year] on columns,
[Product].[Product Categories].[Category] on rows
From [Adventure Works]

GO

Select
[Product].[Product Categories].[Category] on columns,
[Date].[Calendar].[Calendar Year] on rows
From [Adventure Works]
```

خروجی به صورت زیر می‌باشد.

Messages | Results

	CY 2005	CY 2006	CY 2007	CY 2008	CY 2010
Accessories	$20,235.36	$92,735.35	$296,532.88	$161,794.33	(null)
Bikes	$7,395,348.63	$19,956,014.67	$25,551,775.07	$13,399,243.18	(null)
Clothing	$34,376.34	$485,587.15	$871,864.19	$386,013.16	(null)
Components	$615,474.98	$3,610,092.47	$5,482,497.29	$2,091,011.92	(null)

	Accessories	Bikes	Clothing	Components
CY 2005	$20,235.36	$7,395,348.63	$34,376.34	$615,474.98
CY 2006	$92,735.35	$19,956,014.67	$485,587.15	$3,610,092.47
CY 2007	$296,532.88	$25,551,775.07	$871,864.19	$5,482,497.29
CY 2008	$161,794.33	$13,399,243.18	$386,013.16	$2,091,011.92
CY 2010	(null)	(null)	(null)	(null)

چگونگی استفاده از ساختارهای سلسله مراتبی در محورهای مختلف :

برای روشن شدن مطلب چندین نمونه کوئری زیر را باهم اجرا می‌کنیم.

در ابتدا به خاطر داشته باشید که امکان استفاده از یک ساختار سلسله مراتبی در دو محور (Axis) مجزا وجود ندارد. برای روشن‌تر شدن این مطلب کوئری زیر را اجرا کنید:

```
Select
[Date].[Calendar].[Calendar Year] on columns,
[Date].[Calendar].[Month] on rows
From [Adventure Works]
```

در توضیح مثال بالا دقت داشته باشید که واکشی [Calendar Year] و [Month] از یک ساختار سلسله مراتبی یکسان به نام [Date].[Calendar] انجام شده است و این کار در MDX ها غیر مجاز می‌باشد.

البته استفاده از دو ساختار سلسله مراتبی متفاوت از یک دایمنشن در دو محور مجزا امکان پذیر می‌باشد برای مثال :

```
Select
[Date].[Calendar].[Calendar Year] on columns,
[Date].[Month of Year].[Month of Year] on rows
From [Adventure Works]
```

در مثال بالا واکشی از دو ساختار سلسله مراتبی مختلف اما از یک دایمنشن در دو محور مجزا صورت گرفته است.

Messages | Results

	CY 2005	CY 2006	CY 2007	CY 2008	CY 2010
January	(null)	$713,116.69	$1,317,541.83	$1,662,547.32	(null)
February	(null)	$1,900,788.93	$2,384,846.59	$2,700,766.80	(null)
March	(null)	$1,455,280.41	$1,563,955.08	$2,739,370.98	(null)
April	(null)	$882,899.94	$1,865,278.43	$2,204,623.41	(null)
May	(null)	$2,269,116.71	$2,880,752.68	$3,315,275.00	(null)
June	(null)	$1,001,803.77	$1,987,872.71	$3,415,479.07	(null)
July	$489,328.58	$2,393,689.53	$2,665,650.54	(null)	(null)
August	$1,538,408.31	$3,601,190.71	$4,212,971.51	(null)	(null)
September	$1,165,897.08	$2,885,359.20	$4,047,574.04	(null)	(null)
October	$844,721.00	$1,802,154.21	$2,282,115.88	(null)	(null)
November	$2,324,135.80	$3,053,816.33	$3,483,161.40	(null)	(null)
December	$1,702,944.54	$2,185,213.21	$3,510,948.73	(null)	(null)

امکان استفاده از دو ساختار سلسله مراتبی یکسان در یک محور مجاز می‌باشد.

```
Select
[Measures].[Internet Sales Amount] on columns,
{[Date].[Calendar].[Calendar Year],[Date].[Calendar].[Month]} on rows
From [Adventure Works]
```

Messages | Results

	Internet Sales Amount
CY 2005	$3,266,373.66
CY 2006	$6,530,343.53
CY 2007	$9,791,060.30
CY 2008	$9,770,899.74
CY 2010	(null)
July 2005	$473,388.16
August 2005	$506,191.69
September 2005	$473,943.03
October 2005	$513,329.47
November 2005	$543,993.41
December 2005	$755,527.89
January 2006	$596,746.56
February 2006	$550,816.69

همچنین استفاده از دو ساختار سلسله مراتبی مختلف از یک دایمنشن در یک محور مجاز نمی‌باشد.

```
Select
[Measures].[Internet Sales Amount] on columns,
{[Date].[Calendar].[Calendar Year],[Date].[Month of Year].[Month of Year]} on rows
From [Adventure Works]
```

Messages

```
Executing the query ...
Query (3, 2) Members, tuples or sets must use the same hierarchies in the  function.
Execution complete
```

در مثال بالا از دو ساختار سلسله مراتبی [Date].[Calendar] و [Date].[Month of Year] استفاده شده است.

در صورتی می‌توانیم از دو ساختار سلسله مراتبی متفاوت از یک دایمنشن در یک محور استفاده کنیم که آنها را باهم Cross Join کنیم.

```
Select
[Measures].[Internet Sales Amount]on columns,
crossjoin(
[Date].[Calendar].[Calendar Year],
[Date].[Month of Year].[Month of Year]
)on rows
From [Adventure Works]
```

Messages | Results

		Internet Sales Amount
CY 2005	July	$473,388.16
CY 2005	August	$506,191.69
CY 2005	September	$473,943.03
CY 2005	October	$513,329.47
CY 2005	November	$543,993.41
CY 2005	December	$755,527.89
CY 2006	January	$596,746.56
CY 2006	February	$550,816.69
CY 2006	March	$644,135.20
CY 2006	April	$663,692.29
CY 2006	May	$673,556.20
CY 2006	June	$676,763.65

دقت داشته باشید که امکان استفاده از دو ساختار سلسله مراتبی یکسان از یک دایمنشن در Cross Join وجود ندارد همان گونه که در دو مثال قبل مشاهده نمودید. (و البته نیازی هم به استفاده از Cross Join وجود ندارد)

```
Select
[Measures].[Internet Sales Amount] on columns,
crossjoin(
[Date].[Calendar].[Calendar Year],
[Date].[Calendar].[Month]
)on rows
From [Adventure Works]
```

به طور کلی استفاده از Cross Join مانند استفاده از () می باشد . بنابر این می توان کلمه ی Cross Join را ننوشت .

```
Select
{[Measures].[Internet Sales Amount],[Measures].[Reseller Sales Amount]} on columns,
crossjoin(
[Date].[Calendar].[Calendar Year],
[Date].[Month of Year].[Month of Year]
)on rows
From [Adventure Works]
GO
Select
{[Measures].[Internet Sales Amount],[Measures].[Reseller Sales Amount]} on columns,
(
  [Date].[Calendar].[Calendar Year],
  [Date].[Month of Year].[Month of Year]
)on rows
From [Adventure Works]
```

دو کوئری بالا مشابه هم می‌باشند.

می توان از دو Cross Join در دو محور مختلف نیز استفاده کرد.

```
Select
crossjoin(
[Product].[Product Categories].[Category],
{[Measures].[Internet Sales Amount],[Measures].[Reseller Sales Amount]}
)on columns,
crossjoin(
[Date].[Calendar].[Calendar Year],
[Date].[Month of Year].[Month of Year]
)on rows
From [Adventure Works]
```

Messages | Results

		Accessories	Accessories	Bikes	Bikes	Clothing	Clothing
		Internet Sales Amount	Reseller Sales Amount	Internet Sales Amount	Reseller Sales Amount	Internet Sales Amount	Reseller Sales
CY 2005	July	(null)	$1,695.67	$473,388.16	$453,231.80	(null)	$2,875.1
CY 2005	August	(null)	$3,593.20	$506,191.69	$1,413,253.52	(null)	$7,038.5
CY 2005	September	(null)	$3,250.03	$473,943.03	$1,054,995.97	(null)	$5,138.3
CY 2005	October	(null)	$1,937.90	$513,329.47	$777,394.97	(null)	$3,583.7
CY 2005	November	(null)	$5,490.73	$543,993.41	$2,152,858.49	(null)	$8,551.7
CY 2005	December	(null)	$4,267.84	$755,527.89	$1,543,613.88	(null)	$7,188.8

دقت داشته باشید نوع محصول در ستون‌ها بالای سر ستون [Internet Sales Amount] و [Reseller Sales Amount] قرار گرفته است.

استفاده از Cross Join های تودر تو نیز وجود دارد.

```
Select
crossjoin(
[Sales Territory].[Sales Territory].[Country],
crossjoin(
[Product].[Product Categories].[Category],
{
[Measures].[Internet Order Count],
[Measures].[Reseller Order Count]
}
  )
)on columns ,
crossjoin(
[Date].[Calendar].[Calendar Year],
[Date].[Month of Year].[Month of Year]
) on rows
From [Adventure Works]
```

Messages | Results

		France	France	France	France	France	France
		Accessories	Accessories	Bikes	Bikes	Clothing	Clothing
		Internet Order Count	Reseller Order Count	Internet Order Count	Reseller Order Count	Internet Order Count	Reseller Order Count
CY 2005	July	(null)	(null)	13	(null)	(null)	(null)
CY 2005	August	(null)	(null)	6	(null)	(null)	(null)
CY 2005	September	(null)	(null)	11	(null)	(null)	(null)
CY 2005	October	(null)	(null)	4	(null)	(null)	(null)
CY 2005	November	(null)	(null)	12	(null)	(null)	(null)
CY 2005	December	(null)	(null)	13	(null)	(null)	(null)
CY 2006	January	(null)	(null)	11	(null)	(null)	(null)
CY 2006	February	(null)	(null)	11	(null)	(null)	(null)

در کوئری زیر به جای Cross Join از () استفاده شده است

```
Select
crossjoin(
[Sales Territory].[Sales Territory].[Country],
[Product].[Product Categories].[Category],
{
[Measures].[Internet Order Count],
[Measures].[Reseller Order Count]
}
 ) on columns,
([Date].[Calendar].[Calendar Year],[Date].[Month of Year].[Month of Year]) on rows
From [Adventure Works]
```

استفاده از عملگر * مانند استفاده از Cross Join می باشد.

```
Select
crossjoin(
[Sales Territory].[Sales Territory].[Country],
[Product].[Product Categories].[Category],
{
[Measures].[Internet Order Count],
[Measures].[Reseller Order Count]
}
) on columns,
[Date].[Calendar].[Calendar Year] * [Date].[Month of Year].[Month of Year] on rows
From [Adventure Works]
```

در مقالات بعدی آموزش MDX Query را ادامه خواهیم داد.

عنوان: **آموزش MDX Query - قسمت ششم - شروع کار با دستورات MDX**
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۲ ۹:۳۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

امروز اولین دستورات MDX را خواهیم نوشت . قبل از شروع کار فراموش نکنید موارد زیر را حتما انجام داده باشید :

نصب پایگاه داده ی Adventure Work DW 2008 و همچنین نصب پایگاه داده‌ی چند بعدی Adventure Work DW 2008 روی SSAS
مطاله قسمت‌های قبلی برای آشنایی با مفاهیم پایه .

در صورتیکه پیش شرایط فوق را نداشته باشید، احتمالا در ادامه با مشکلاتی مواجه خواهید شد؛ زیرا برای آموزش MDX Query ها از پایگاه داده‌ی Adventure Work DW 2008 استفاده شده است.

دقت داشته باشید که MDX Query ها تا حدودی شبیه T/SQL می‌باشند؛ اما مطلقا از نظر مفهومی با هم شباهت ندارند. به عبارت دیگر ما در T/SQL با یک مدل رابطه‌ای سرو کار داریم در حالیکه در MDX ها با یک پایگاه داده چند بعدی کار می‌کنیم. به بیان دیگر در پایگاه داده‌های رابطه‌ای صحبت از جداول، ردیف‌ها، ستون‌ها و ضرب دکارتی مجموعه‌ها می‌باشد، اما در پایگاه داده‌های چند بعدی در خصوص Dimension,Fact,Cube,Tuple و ... صحبت می‌کنیم. البته ماکروسافت تلاش کرده‌است تا حد زیادی Syntax ها شبیه به یکدیگر باشند.

نحوه‌ی نوشتن یک Select در MDX ها به صورت زیر می‌باشد :

```
Select
{} On Columns ,
{} On Rows
From <Cube_Name>
Where <Condition>
```

در ادامه با اجرای هر کوئری، توضیحات لازم در خصوص آن ارایه می‌گردد و با پیگیری این آموزش‌ها می‌توانید مفاهیم، توابع و ... را در MDX Query ها بیاموزید.

برای اجرای دستورات زیر باید Microsoft SQL Server Management Studio را باز نمایید و به سرویس SSAS متصل شوید. سپس پایگاه داده‌ی Adventure Works DW 2008R2 را انتخاب نمایید و از Cubes Adventure Works را انتخاب نمایید.

حال دکمه‌ی New Query را در بالای صفحه بزنید (Ctrl + N)

سپس در صفحه‌ی باز شده می‌توانید Cube یا SubCube های آن Cube را انتخاب کرده و کمی پایین‌تر Measure Group را خواهیم داشت و در انتها Measure ها و Dimension ها قرار گرفته‌اند. (در هنگام نوشتن Select می‌توان از عمل Drag&Drop برای آسان‌تر شدن نوشتن MDX Query ها نیز استفاده کنید)

متاسفانه هنوز در IDE مربوط به SQL Server کلیدی برای مرتب سازی دستورات MDX وجود ندارد و البته در نرم افزار هایی مانند SQL Toll Belt هم چنین چیزی قرار داده نشده است . بنابر این توصیه می‌شود در نوشتن دستورات MDX تمام تلاش خود را بکنید تا دستوراتی مرتب و خوانا را تولید کنید.

با اجرای دستور زیر اولین کوئری خود را در پایگاه داده‌ی چند بعدی بنویسید (برای اجرا کلید F5 مانند T/SQL کار خواهد کرد.)

```
Select
From [Adventure Works]
```

شاید تعجب کنید. کوئری فاقد قسمت Projection می‌باشد! در MDX ها می‌توان هیچ سطر یا ستونی را انتخاب نکرد. اما چگونه؟ و خروجی نمایش داده شده چیست؟

برای توضیح مطلب فوق باید در خصوص Default Measure کمی اطلاعات داشته باشید. در هنگام Deploy کردن پروژه در SSAS برای هر Cube یک Measure به عنوان Measure پیش فرض انتخاب شده. بنابر این در صورتیکه هیچ گونه Projection یا Where ایی اعمال نشده باشد، SQL Server به صورت پیش فرض مقدار Mesaure پیش فرض را بدون اعمال هیچ بعدی نمایش می‌دهد.

خروجی دستور بالا مشابه تصویر زیر می‌باشد.

حال دستور زیر را اجرا می‌کنیم :

```
Select
From [Adventure Works]
Where [Measures].[Reseller Sales Amount]
```

تصویر خروجی به صورت زیر می‌باشد :

شاید باز هم تعجب کنید. نوشتن نام یک شاخص به جای عبارت شرط؟! آیا خروجی عبارات شرطی نباید Boolean باشند؟

خیر. اگر چنین پرسش هایی در ذهن شما ایجاد شده باشد، به دلیل مقایسه‌ی MDX با T/SQL می‌باشد. در اینجا شرط Where بر روی ردیف‌های جدول مدل رابطه ای اعمال نمی‌شود و عملا بیانگر واکشی اطلاعات از مدل چند بعدی می‌باشد. با اعمال شرط فوق به SSAS اعلام کرده ایم که خروجی بر اساس شاخص [Measures].[Reseller Sales Amount] باشد. با توجه به این که شاخص انتخاب شده با شاخص پیش فرض یکی می‌باشد خروجی با حالت قبل تفاوتی نخواهد کرد.

برای درک بهتر، کوئری زیر را اجرا کنید :

```
Select
From [Adventure Works]
where [Measures].[Internet Sales Amount]
```

استفاده از این شرط سبب استفاده نشدن از شاخص پیش فرض می شود . به عبارت دیگر این کوئری دارای سرجمع مبلغ فروش اینترنتی می باشد.

دستور زیر را اجرا کنید :

```
Select
[Measures].[Reseller Sales Amount] on columns
From [Adventure Works]
```

با اعمال یک شاخص خاص در ستون ، عملا فیلترینگ انجام می شود

استفاده از یک دایمنشن در ستون :

دستور زیر را اجرا کنید

```
Select
[Date].[Calendar].[Calendar Year] on columns
From [Adventure Works]
```

خروجی به شکل زیر خواهد بود

```
Select
    [Date].[Calendar].[Calendar Year] on columns
From [Adventure Works]
```

100 %

Messages | Results

CY 2005	CY 2006	CY 2007	CY 2008	CY 2010
$8,065,435.31	$24,144,429.65	$32,202,669.43	$16,038,062.60	(null)

همان طور که مشاهده می‌کنید خروجی دارای چندین ستون می‌باشد و دارای مقادیری در هر ستون. اما این مقادیر از کجا آمده اند؟

همواره این نکته را به خاطر بسپارید که در صورت عدم ذکر نام یک Measure در کوئری ، SSAS از Measure پیش فرض استفاده می‌کند. حال کوئری فوق میزان فروش نمایندگان (Reseller Sales Amount) را در هر سال نمایش می‌دهد.

سوال بعدی این می‌باشد که این سال‌ها از کجا آمده اند؟ خوب برای درک بهتر این مورد می‌توانیم مانند تصویر زیر به دایمنشن Date رفته و در ساختار سلسله مراتبی ، اعضای سطح [Date].[Calendar].[Calendar Year] را مشاهده کنیم.

ایجاد سرجمع ستون‌ها :

کوئری زیر را اجرا نمایید

```
Select
{[Date].[Calendar].[Calendar Year],[Date].[Calendar]} on columns
From [Adventure Works]
```

بعد از اجرا تصویر زیر را خواهید دید :

100 %

Messages | Results

CY 2005	CY 2006	CY 2007	CY 2008	CY 2010	All Periods
$8,065,435.31	$24,144,429.65	$32,202,669.43	$16,038,062.60	(null)	$80,450,596.98

سوال اول این می‌باشد که کاربرد {} در انتخاب دایمنشن‌ها چیست؟ در پاسخ می‌توان گفت که اگر شاخص ها یا بعد ها ، مرتبط به یک سلسله مراتب باشند آنها را در یک {} قرار می دهیم ولی اگر سلسله مراتب متفاوت باشد، یا بعد و شاخص باشند باید در () قرار بگیرند .

خوب همان طور که مشخص است در ساختار سلسله مراتبی ابتدا سال و بعد یک سطح بالاتر را انتخاب کرده ایم این به معنی نمایش سرجمع در سطح بالاتر از سال می‌باشد(سرجمع تمامی سال ها).

استفاده از دایمنشن و Measure در سطر و ستون مجرا :

کوئری زیر را اجرا نمایید

```
Select
{[Date].[Calendar].[Calendar Year],[Date].[Calendar]} on columns,
[Product].[Product Categories].[Category] on rows
From [Adventure Works]
```

خروجی مشابه شکل زیر می‌باشد

```
Select
    {[Date].[Calendar].[Calendar Year],[Date].[Calendar]} on columns,
    [Product].[Product Categories].[Category] on rows
From [Adventure Works]
```

100 %

Messages | Results

	CY 2005	CY 2006	CY 2007	CY 2008	CY 2010	All Periods
Accessories	$20,235.36	$92,735.35	$296,532.88	$161,794.33	(null)	$571,297.93
Bikes	$7,395,348.63	$19,956,014.67	$25,551,775.07	$13,399,243.18	(null)	$66,302,381.56
Clothing	$34,376.34	$485,587.15	$871,864.19	$386,013.16	(null)	$1,777,840.84
Components	$615,474.98	$3,610,092.47	$5,482,497.29	$2,091,011.92	(null)	$11,799,076.66

در مثال فوق از بعدها در ستون و همزمان، نمایش نوع دسته بندی محصولات در ردیف‌ها استفاده شده است. به عبارت دیگر نتیجه عبارت است از فروش نمایندگان فروش (Reseller Sales Amount) براساس هر سال به تفکیک نوع دسته بندی محصول

فروخته شده.

(کسانی که چنین گزارشی را با استفاده از T/SQL نوشته اند، احتمالا از آسانی نوشتن این گزارش توسط MDX ها شگفت زده شده اند.)

قراردادن فیلد سرجمع در ردیف :

برای این منظور کوئری زیر را اجرا نمایید

```
Select
{[Date].[Calendar].[Calendar Year],[Date].[Calendar]} on columns,
{[Product].[Product Categories].[Category],[Product].[Product Categories]}on rows
From [Adventure Works]
```

خروجی به صورت زیر می‌باشد

```
Select
    {[Date].[Calendar].[Calendar Year],[Date].[Calendar]} on columns,
    {[Product].[Product Categories].[Category],[Product].[Product Categories]}on rows
From [Adventure Works]
```

100 %

Messages | Results

	CY 2005	CY 2006	CY 2007	CY 2008	CY 2010	All Periods
Accessories	$20,235.36	$92,735.35	$296,532.88	$161,794.33	(null)	$571,297.93
Bikes	$7,395,348.63	$19,956,014.67	$25,551,775.07	$13,399,243.18	(null)	$66,302,381.56
Clothing	$34,376.34	$485,587.15	$871,864.19	$386,013.16	(null)	$1,777,840.84
Components	$615,474.98	$3,610,092.47	$5,482,497.29	$2,091,011.92	(null)	$11,799,076.66
All Products	$8,065,435.31	$24,144,429.65	$32,202,669.43	$16,038,062.60	(null)	$80,450,596.98

نحوه‌ی نمایش سرجمع در ردیف، مشابه نمایش سرجمع در ستون می‌باشد.

استفاده از تابع non empty :

برای حذف ستون هایی که کاملا دارای مقدار null می‌باشند به صورت زیر عمل می‌کنیم :

```
Select
non empty {[Date].[Calendar].[Calendar Year],[Date].[Calendar]} on columns ,
{[Product].[Product Categories].[Category],[Product].[Product Categories]} on rows
From [Adventure Works]
```

خروجی به صورت زیر می‌باشد:

Messages | Results

	CY 2005	CY 2006	CY 2007	CY 2008	All Periods
Accessories	$20,235.36	$92,735.35	$296,532.88	$161,794.33	$571,297.93
Bikes	$7,395,348.63	$19,956,014.67	$25,551,775.07	$13,399,243.18	$66,302,381.56
Clothing	$34,376.34	$485,587.15	$871,864.19	$386,013.16	$1,777,840.84
Components	$615,474.98	$3,610,092.47	$5,482,497.29	$2,091,011.92	$11,799,076.66
All Products	$8,065,435.31	$24,144,429.65	$32,202,669.43	$16,038,062.60	$80,450,596.98

انتخاب دو دایمنشن در سطر و ستون و مشخص نمودن یک Measure خاص برای کوئری :

برای این کار به صورت زیر عمل خواهیم کرد:

```
Select
{[Date].[Calendar].[Calendar Year],[Date].[Calendar]} on columns,
{[Product].[Product Categories].[Category],[Product].[Product Categories]} on rows
From [Adventure Works]
Where [Measures].[Internet Sales Amount]
```

در اینجا با اعمال شرط Where عملا از SSAS خواسته‌ایم خروجی برای شاخص مشخص شده واکشی شود.

در بالا میزان فروش اینترنتی برای دسته بندی محصولات و در سال‌های مختلف ارائه و همچنین سرجمع ستون و سطر نیز نمایش داده شده است.

	CY 2005	CY 2006	CY 2007	CY 2008	CY 2010	All Periods
Accessories	(null)	(null)	$293,709.71	$407,050.25	(null)	$700,759.96
Bikes	$3,266,373.66	$6,530,343.53	$9,359,102.62	$9,162,324.85	(null)	$28,318,144.65
Clothing	(null)	(null)	$138,247.97	$201,524.64	(null)	$339,772.61
Components	(null)	(null)	(null)	(null)	(null)	(null)
All Products	$3,266,373.66	$6,530,343.53	$9,791,060.30	$9,770,899.74	(null)	$29,358,677.22

در صورتیکه بخواهیم ستون و سطرهایی را که دارای مقدار null در تمامی آن سطر یا ستون می‌باشند، حذف کنیم به صورت زیر عمل می‌کنیم:

```
Select
non empty {[Date].[Calendar].[Calendar Year],[Date].[Calendar]} on columns,
non empty {[Product].[Product Categories].[Category],[Product].[Product Categories]} on rows
From [Adventure Works]
Where [Measures].[Internet Sales Amount]
```

	CY 2005	CY 2006	CY 2007	CY 2008	All Periods
Accessories	(null)	(null)	$293,709.71	$407,050.25	$700,759.96
Bikes	$3,266,373.66	$6,530,343.53	$9,359,102.62	$9,162,324.85	$28,318,144.65
Clothing	(null)	(null)	$138,247.97	$201,524.64	$339,772.61
All Products	$3,266,373.66	$6,530,343.53	$9,791,060.30	$9,770,899.74	$29,358,677.22

اگر در یک دایمنشن فقط یک سلسله مراتب باشد یا اصلا سلسله مراتبی وجود نداشته باشد، می توان از نام خود دایمنشن استفاده کرد

```
Select
[Sales Channel] on columns
From [Adventure Works]
```

Messages | Results

All Sales Channels
$80,450,596.98

و دقت داشته باشید دایمنشنی که دارای بیش از یک سلسله مراتب باشد، حتما باید در Select مشخص شود که از کدام سلسله مراتب می خواهیم استفاده کنیم . در غیر این صورت با خطا مواجه خواهیم شد.

```
Select
[Product] on columns
From [Adventure Works]
```

Messages

```
Executing the query ...
Query (2, 2) The 'Product' dimension contains more than one hierarchy, therefore the hierarchy must be explicitly
Execution complete
```

استفاده از فیلدهای یک دایمنشن که دارای سلسه مراتب می باشد نیز جایز می باشد

```
Select
[Product].[Category] on columns
From [Adventure Works]
```

All Products
$80,450,596.98

```
Select
[Product].[Category].[all]   on columns
From [Adventure Works]
--
Select
[Product].[Category].[All] on columns
From [Adventure Works]
--
Select
[Product].[Category].[(all)] on columns
From [Adventure Works]
--
Select
[Product].[Category].[all products] on columns
From [Adventure Works]
```

برای به دست آوردن سرجمع کل روی یک صفت از دایمنشن، باید از سه حالت آخر استفاده کرد. حالت اول خطا دارد و خروجی خالی نمایش داده می شود .

در صورتی که بخواهیم از یک دایمنشن تمامی Member های آن را واکشی کنیم به صورت زیر عمل خواهیم کرد

```
Select
{[Product].[Category].members} on columns
From [Adventure Works]
```

استفاده از Members روی یک خصوصیت در دایمنشن به معنی دریافت سرجمع آن صفت و سپس تک تک اجزای آن صفت می‌باشد.

Messages | Results

All Products	Accessories	Bikes	Clothing	Components
$80,450,596.98	$571,297.93	$66,302,381.56	$1,777,840.84	$11,799,076.66

اگر از یک صفت واکشی اطلاعات انجام شود در سطح اعضای آن ، در آن صورت دیگر سرجمع نمایش داده نمی شود و فقط جمع هر عضو در آن صفت نمایش داده می شود .

```
Select
[Product].[Category].[Category].members
-- dimension.hierarchy.level.members
on columns
From [Adventure Works]
```

Accessories	Bikes	Clothing	Components
$571,297.93	$66,302,381.56	$1,777,840.84	$11,799,076.66

اگر بخواهیم دو ستون را داشته باشیم که هر دو برای یک دایمنشن می‌باشند باید از {} استفاده کرد . دستور اول خطا خواهد داشت.

```
Select
[Product].[Category].[Category].members,[Product].[Category].[All Products] on columns
From [Adventure Works]
```

در دستور دوم با استفاده از {} خروجی نمایش داده می‌شود که عبارت است از تمامی اعضای سطح [Product].[Category].[Category]. به همراه سرجمع تمامی محصولات.

```
Select
{[Product].[Category].[Category].members,[Product].[Category].[All Products]} on columns
From [Adventure Works]
```

Accessories	Bikes	Clothing	Components	All Products
$571,297.93	$66,302,381.56	$1,777,840.84	$11,799,076.66	$80,450,596.98

یک راه کوتاه‌تر برای انتخاب تمامی اعضا و سرجمع آنها

```
Select
{[Product].[Category].[Category],[Product].[Category]}
on columns
From [Adventure Works]
```

می توان از کلمات Members, All X استفاده نکرد.

Accessories	Bikes	Clothing	Components	All Products
$571,297.93	$66,302,381.56	$1,777,840.84	$11,799,076.66	$80,450,596.98

انتخاب اولین دسته بندی محصول البته این ترتیب بر اساس Key Columns در SSAS می باشد .

```
Select
[Product].[Category].&[1]
on columns
From [Adventure Works]
```

Bikes
$66,302,381.56

انتخاب دقیق یک عضو در خروجی

```
Select
[Product].[Category].[Bikes]
on columns
From [Adventure Works]
```

Bikes
$66,302,381.56

انتخاب دو عضو از یک دایمنشن

```
Select
{[Product].[Category].[Bikes],[Product].[Category].[Clothing]}
on columns
From [Adventure Works]
```

Bikes	Clothing
$66,302,381.56	$1,777,840.84

واکشی تمامی دسته بندی محصولات بر اساس Measure پیش فرض :

```
Select
[Product].[Product Categories].members
on columns
From [Adventure Works]
```

All Products	Accessories	Bike Racks	Hitch Rack - 4-Bike	Bike Stands	All-Purpose Bike Stand	Bottles and Cages	Mountain Bottle Cage	Road Bottl
$80,450,596.98	$571,297.93	$197,736.16	$197,736.16	(null)	(null)	$7,476.60	(null)	(null

در صورتیکه بخواهیم دو Dimension مختلف را در یک ستون یا سطر بیاوریم باید از Join استفاده کنیم. بنابر این دو دستور زیر با خطا روبرو می‌شوند

```
Select
```

```
[Product].[Product Categories],[Product].[Category]
on columns
From [Adventure Works]

Go

Select
{[Product].[Product Categories],[Product].[Category]}
on columns
From [Adventure Works]
```

```
Executing the query ... 01.
Parser: The statement dialect could not be resolved due to ambiguity.
Execution complete 01.

Executing the query ... 11.
Query (2, 2) Members, tuples or sets must use the same hierarchies in the  function.
Execution complete 11.
```

تعریف Axis : به هر کدام از ستون یا سطر یک محور یا Axis گفته می‌شود.

با بررسی مثال فوق به نتایج زیر خواهیم رسید.

1. امکان استفاده از دو سلسله مراتب مختلف از یک دایمنشن در یک Axis وجود ندارد . مگر اینکه آنها را باهمدیگر CrossJoin کنیم .

2. امکان استفاده از دو سلسله مراتب مختلف از یک دایمنشن در دو Axis مختلف وجود دارد .

ترتیب انتخاب Axis ها به صورت زیر می‌باشد:

1. Columns

2. Rows

برای مشخص شدن موضوع کوئری زیر را اجرا کنید

```
Select
[Product].[Product Categories].members
on rows
From [Adventure Works]
```

```
Executing the query ...
Query (2, 2) Axis numbers specified in a query must be sequentially specified, and cannot contain gaps.
Execution complete
```

نمی‌توانیم ردیفی را واکشی کنیم بدون اینکه ستونی برای کوئری مشخص کرده باشیم.

البته می‌توان ستون خالی ایجاد نماییم مانند مثال زیر :

```
Select
{} on columns,
[Product].[Product Categories].members
on rows
From [Adventure Works]
```

البته در این صورت خروجی فقط نام دسته بندی محصولات خواهد بود زیرا هیچ ستونی مشخص نشده .

در مقالات بعدی به ادامه‌ی مطالب MDX Query خواهیم پرداخت.

نظرات خوانندگان

نویسنده: علاقه مند
تاریخ: ۱۳۹۲/۱۰/۱۰ ۰:۵۵

با سلام خدمت شما، بسیار سپاسگذارم به خاطر این سری مقالات آموزشی که باعث شد با زبان جذاب و فوق العاده‌ی MDX آشنا شوم.
بنده با گزارشات آماری cross tab آشنا هستم، خروجی مثال هایتان دقیقا مشابه با خروجی گزارشات cross tab است (مو نمی‌زنه!)
نوشتن Queryهای pivoting/cross tabulation با این زبان نسبت به زبان SQL واقعا ساده‌تر است. حقیقتا لذت بردم.
فقط خواهشی داشتم، اگر مقدور است به مثال هایی بپردازید که در آنها ستون‌ها یا سطرها در Rangeهای مختلف گروه بندی شوند.
مثال:

Columns: Age	-10	10 - 20	21 -30	31-40	+40
Rows: Watching TV (min)					
-20					
20-40					
41-60					
+60					

جدول فوق یک گزارش آماری cross tab است. که مشخص کرده افراد (مرد و زن) در رده‌های سنی مختلف (زیر 10 سال، بین 10 و 20 سال و ...) روزانه چند دقیقه تلوزیون تماشا می‌کنند (کمتر از 20 دقیقه، بین 20 تا 40 دقیقه..). این بازه‌ها را میشه برای داده هایی مثل تاریخ، روزها، هفته ها، ماه‌ها و ... نیز در نظر گرفت.
و سوالی نیز داشتم، آیا می‌توان Queryهای MDX را به T-SQL تبدیل کرد (منظور بصورت خودکار است نه بازنویسی آن) ؟
و درخواست پایانی، لطفا به بحث Reporting هم بپردازید که ببینیم این نتیجه در قالب گزارش تجاری چگونه ظاهر می‌شوند.

نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۱۰ ۱:۳۴

آقای علاقه مند :
در خصوص سوال اول شما (ستون‌ها یا سطرها در Rangeهای مختلف گروه بندی شوند) این کار را باید با طراحی مناسب جداول Fact,Dimension در DW انجام دهید. به عبارت دیگر جدول Fact در خصوص آمار داشته باشید که یکی از فیلدهای آن مشاهده تلویزیون می‌باشد و دو عدد Dimension که یکی گروه‌های سنی و دیگری ساعات می‌باشد.
حال باید در ETL واکشی مناسبی برای پر کردن جدول Fact داشته باشد.
در خصوص تبدیل MDX Query به T/SQL اطلاع دقیقی ندارم و با قاطعیت نمی‌توانم بگویم نه. اما استدلال بنده این می‌باشد که از آنجایی که MDX Queryها روی پایگاه داده‌ی MDB نوشته می‌شوند کاملا متفاوت از T\SQL می‌باشند که روی مدل رابطه ای کار می‌کند بنابر این تصور می‌کنم چنین امکانی نباشد و درضمن در جستجوی وب هم چیزی نیافتم.
در خصوص نمایش نهایی این چنین گزارشاتی چندین روش وجود دارد
واکشی اطلاعات در Excel
استفاده از Performance Point Service در Sharepoint (ترجیحا 2013)
استفاده از کامپوننت‌های OLAP در محیط Visual Studio مانند RadarSoft,Dundas,...

استفاده از نرم افزار هایی که کارشان ساختن گزارشات از MDBها می‌باشد (CubeWare)

نویسنده: علاقه مند
تاریخ: ۱۳۹۲/۱۰/۱۰ ۱۱:۲

سلام مجدد،
ممنونم از پاسختتون.

راستش شناختی نسبت به این دو نوع جدولی که فرمودین ندارم (Fact, Dimension). اشاره کردین که باید رده بندی‌های مربوطه یعنی رده سنی و بازه زمانی را باید در جداول Dimension پیاده کرد. آیا این پیاده سازی به معنای عدم داینامیک بودن است؟
منظورم اینه که فرضا اگر بخواهیم روزی به جای اینکه سن افراد را ده سال ده سال دسته بندی کنیم،تصمیم گرفتیم اینکار را پنج سال پنج سال انجام دهیم آیا باز نیاز هست مجدد یک جدول دیگر طراحی کنیم؟

آیا در این زبان امکان استفاده از عملگرهای ابتدایی جبری مثل تقسیم وجود دارد؟ فرضا بخواهیم نسبت مقادیر دو ستون را بدست آوریم (بطور نمونه مقایسه تورم شش ماه اول سال جاری نسبت به شش ماه اول سال گذشته)؟

نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۱۰ ۱۶:۲۵

در قسمت اول این مجموعه توضیحات لازم را در خصوص مقدمات کار و مفاهیم اولیه خدمتتان ارایه کردم. برای آشنایی با ساخت Data Warehouse لطفا مراجعه کنید به قسمت (**مدل داده‌ای رابطه‌ای (Relational) وچند بعدی (Multidimensional)**) در سری اول.
در خصوص ایجاد ابعاد ی مانند بعد تاریخ و بعد سن باید عرض کنم که این ابعاد امکان ارایه انواع گزارشات مختلف را به شما خواهند داد و مطلقا OLAP را از حالت دینامیک خارج نمیکند.
در خصوص نحوه‌ی تحلیل و طراحی DW ، امیدوارم به زودی بتوانم مقاله ای را انتشار بدهم.
در خصوص عملگرهای تقسیم و ... و توابع جمعی به زودی در قسمت‌های بعدی اطلاعات کاملی را خدمتتان ارایه خواهم کرد. به طور خلاصه اینکه OLAP برای ایجاد گزارشات مدیریتی می‌باشد و قطعا این موارد شما در ادامه پوشش داده خواهد شد.(شما به راحتی می‌توانید سرجمع یک بازه‌ی تاریخی یا سنی را برای یک Measure خاص بدست آورید.)

نویسنده: میم
تاریخ: ۱۳۹۳/۰۲/۲۸ ۱۱:۴۱

با سلام و سپاس فراوان از مطالب پر مغزتان
سوال من این است که آیا امکان انجام عملیات group by بر روی پایگاه داده چند بعدی وجود دارد؟

عنوان: آموزش MDX Query - قسمت هشتم - کار بر روی ساختار های سلسله مراتبی
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۲ ۱۴:۵۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX

اگر دایمنشنی بیشتر از یک ساختار سلسله مراتبی باشد ، باید حتما مشخص شود که اعضای کدام ساختار سلسله مراتبی مورد نظر ما می باشد . بنابراین کوئری‌های زیر دارای خطا می‌باشند.

```
Select
[Customer].members on columns
From [Adventure Works]
GO
Select
[Scenario].members on columns
From [Adventure Works]
```

دو کوئری زیر را در نظر بگیرید :

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].members on rows
From [Adventure Works]
```

و

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Country].members on rows
From [Adventure Works]
```

تفاوت این دو کوئری در واکشی اطلاعات از اعضای سطح ساختار سلسله مراتبی دایمنشن می‌باشد. برای توضیح بیشتر ابتدا به خروجی آن‌ها نگاهی بیندازید.

Messages | Results

	Internet Sales Amount
All Customers	$29,358,677.22
Australia	$9,061,000.58
New South Wales	$3,934,485.73
Alexandria	(null)
2015	(null)
Coffs Harbour	$235,454.97
2450	$235,454.97
Adriana Smith	$5,333.25
Aimee Guo	$77.27
Allison R. Young	$39.98
Ann A. Sara	$39.98
Antonio G. Patterson	$8,068.03
Ariana Stewart	$6,070.59
Arthur Kapoor	$23.97

Messages | Results

	Internet Sales Amount
Australia	$9,061,000.58
Canada	$1,977,844.86
France	$2,644,017.71
Germany	$2,894,312.34
United Kingdom	$3,391,712.21
United States	$9,389,789.51

در کوئری اول تمامی سطوح موجود در ساختار سلسله مراتبی دایمنشن مورد نظر واکشی می‌شود. در کوئری دوم فقط سطح کشور از ساختار سلسله مراتبی واکشی می‌شود.

ساختار سلسله مراتبی دایمنشن [Customer] در شکل زیر نمایش داده شده است.

امکان واکشی عضوهای یک عضو از ساختار سلسله مراتبی وجود ندارد

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Country].[France].members on rows
From [Adventure Works]
```

دقت کنید که France خودش عضو سطح Country در ساختار سلسله مراتبی Customer Geography در دایمنشن Customer می‌باشد و دیگر امکان واکشی اعضای این عضو وجود ندارد

بچه‌های یک عضو را به شکل زیر می‌توان واکشی نمود.

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Country].[France].children on rows
From [Adventure Works]
```

برای بدست آوردن بچه های یک عضو در یک ساختار سلسله مراتبی، از children استفاده می شود .

	Internet Sales Amount
Charente-Maritime	$34,441.73
Essonne	$279,297.18
Garonne (Haute)	$54,641.72
Gers	(null)
Hauts de Seine	$263,416.19
Loir et Cher	$21,473.74
Loiret	$91,562.91
Moselle	$94,046.23
Nord	$391,400.20
Pas de Calais	$11,342.92
Seine (Paris)	$539,725.80
Seine et Marne	$109,735.24
Seine Saint Denis	$379,479.75
Somme	$29,555.28

دقت داشته باشید، همانگونه که عضو، دارای Members نمی‌باشد، یک بچه هم دارای Children نمی‌باشد. بنابراین کوئری زیر با خطا مواجه می‌شود.

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Country].[France].children.children on rows
From [Adventure Works]
```

البته شاید تصور می‌شد که باید به عنوان یک ساختار سلسله مراتبی بعد از کشور، استان و سپس شهرها واکشی شوند. اما برای واکشی ساختار سلسله مراتبی باید به صورت زیر عمل کرد.

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[Country].[France],
[Customer].[Customer Geography].[City]
) on rows
From [Adventure Works]
```

عملا برای یافتن شهرهای کشور فرانسه، باید دو سطح در ساختار سلسله مراتبی پایین برویم. برای این منظور از تابع descendants () استفاده کرده‌ایم و به خاطر داشته باشید که نوشتن France].children.children] مجاز نمی‌باشد. همچنین در تابع فوق دو پارامتر ورودی داریم؛ اولی مشخص کننده‌ی یک عضو از سطح مبدا می‌باشد و دومین پارامتر مشخص کننده‌ی سطحی است که باید واکشی در آن صورت بگیرد.

Messages | Results

	Internet Sales Amount
Saint Ouen	$34,441.73
Les Ulis	$181,244.73
Morangis	$56,432.84
Verrieres Le Buisson	$41,619.61
Colomiers	$54,641.72
Aujan Mournede	(null)
Boulogne-Billancourt	$14,289.24
Colombes	$90,268.51
Courbevoie	$38,809.63
Paris La Defense	$45,350.86
Sèvres	$39,598.20
Suresnes	$35,099.73
Saint Ouen	$21,473.74
Orleans	$91,562.91

برای مشخص‌تر شدن موضوع به مثال زیر توجه کنید

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[Country].[France],
[Customer].[Customer Geography].[Customer]
) on rows
From [Adventure Works]
```

در این مثال واکشی در چهار سطح پایین‌تر انجام شده است و به این معنی می‌باشد که ما مشتریان کشور فرانسه را در ردیف‌ها واکشی کرده‌ایم.

	Internet Sales Amount
Brad Nath	$48.97
Bridget C. Nath	$1,560.43
Chelsea Jordan	$1,735.98
Clayton Zhang	$2,049.10
Colin R. Wang	$47.97
Craig R. Gill	$23.78
Darren A. Ruiz	$43.46
Dominic A. Garcia	$1,382.97
Edwin Zhou	$2,753.82
Jon Cai	$88.97
Julie Pal	$2,778.52
Katie Shan	$8.99
Levi W. Rodriguez	$4.99
Nancy Schmidt	$5,033.14

عدم وجود یک Member :

در صورت عدم وجود یک عضو در واکشی، خروجی خالی می‌باشد و خطا نمایش داده نمی‌شود.

```
Select
[Customer].[Customer Geography].[Customer].[Crystal Zhen] on columns
From [Adventure Works]
```

مشتری با نام Crystal Zhen وجود ندارد؛ بنابراین نتیجه خالی می‌باشد.

پیدا کردن یک عضو در مجموعه ایی از اعضا، البته اگر این عضو در آ ن مجموعه وجود نداشته باشد خروجی خالی می باشد.

```
Select
exists(
[Customer].[Customer Geography].[Customer].[Crystal Zheng],
[Customer].[Customer Geography].members
  )on columns
From [Adventure Works]
```

در نوشتن کوئری‌ها دقت کنید که حتما بین دایمنشن‌ها و شاخص‌ها در SSAS ارتباط ایجاد شده باشد. در غیر این صورت خروجی نامعتبر خواهد بود. به عنوان مثال در کوئری زیر بین شاخص [Measures].[Reseller Sales Amount] و مشتری ارتباطی وجود ندارد (این ارتباط باید در هنگام ساخت Cube در SSAS ایجاد می‌گردید تا در هنگام Deploy کردن محاسبات لازم صورت بگیرد) بنابر این در خروجی شاهد نمایش شاخص پیش فرض Cube می‌باشیم.

```
Select
[Measures].[Reseller Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].members on rows
From [Adventure Works]
```

	Reseller Sales Amount
Adriana Smith	$80,450,596.98
Aimee Guo	$80,450,596.98
Allison R. Young	$80,450,596.98
Ann A. Sara	$80,450,596.98
Antonio G. Patterson	$80,450,596.98
Ariana Stewart	$80,450,596.98
Arthur Kapoor	$80,450,596.98
Barbara W. Lal	$80,450,596.98
Bobby D. Saunders	$80,450,596.98
Brianna J. Johnson	$80,450,596.98
Bruce G. Madan	$80,450,596.98
Bryant L. Perez	$80,450,596.98
Carla D. Madan	$80,450,596.98
Carlos Edwards	$80,450,596.98

قبلا گفته شد که هر سطح دارای Members می‌باشد و میتوان برای هر عضو Childrenهای آن عضو را نمایش داد، با این وجود دقت کنید که اعضای آخرین سطح ساختار سلسله مراتبی دارای Children نمی‌باشند.

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].children on rows
From [Adventure Works]
```

خروجی خالی می‌باشد.

بدست آوردن پدر یک عضو در ساختار سلسله مراتبی :

کوئری زیر را اجرا کنید

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].parent on rows
From [Adventure Works]
```

با توجه به ساختار سلسله مراتبی تعریف شده، پدر این مشتری یک کد پستی می باشد .

برای بدست آوردن پدربزرگ یک ساختار سلسله مراتبی داریم :

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].parent.parent on rows
From [Adventure Works]
```

و برای بدست آوردن پدر در 4 سطح بالاتر داریم

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].parent.parent.parent.parent
on rows
From [Adventure Works]
```

در صورتیکه با بالا رفتن از سطوح یک ساختار سلسله مراتبی از ریشه رد شویم ، خروجی خالی بر می‌گردد .

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].parent
.parent.parent.parent.parent.parent
on rows
From [Adventure Works]
```

برای بدست آوردن نیا می‌توانیم از تابع ancestor() استفاده کنیم.

بدست آوردن نیا برای یک مشتری در سطح شهر

```
Select
[Measures].[Internet Sales Amount] on columns,
ancestor(
[Customer].[Customer Geography].[Customer].[Crystal Zheng],
[Customer].[Customer Geography].[City]
)on rows
From [Adventure Works]
```

بدست آوردن نیا برای یک مشتری در سطح شهر

```
Select
[Measures].[Internet Sales Amount] on columns,
ancestor(
[Customer].[Customer Geography].[Customer].[Crystal Zheng],
[Customer].[Customer Geography].[City]
)on rows
From [Adventure Works]
```

پارامتر دوم در تابع ancestor می‌تواند هم مشخص کننده‌ی یک سطح باشد توسط نام آن سطح و هم می‌تواند یک عدد باشد که مشخص کننده‌ی تعداد سطحی است که باید در واکشی اطلاعات از ساختار سلسله مراتبی بالا برویم.

```
Select
[Measures].[Internet Sales Amount] on columns,
ancestor(
[Customer].[Customer Geography].[Customer].[Crystal Zheng],
2
) on rows
From [Adventure Works]
```

بدست آوردن نیا برای یک مشتری در دو سطح بالاتر

در صورتی که بخواهیم تمامی سطوح بالاتر از یک سطح را داشته باشیم و بعد هم سر جمع داشته باشیم، به صورت زیر عمل می‌کنیم و از تابع ascendants استفاده می‌کنیم.

```
Select
[Measures].[Internet Sales Amount] on columns,
ascendants(
[Customer].[Customer Geography].[Customer].[Crystal Zheng]
  ) on rows
From [Adventure Works]
```

Messages | Results

	Internet Sales Amount
Crystal Zheng	$60.47
78000	$102,657.25
Versailles	$102,657.25
Yveline	$268,664.80
France	$2,644,017.71
All Customers	$29,358,677.22

دقت داشته باشید که در این تابع تمامی سطوح از سطح مشخص شده با سمت ریشه باز گردانده می‌شوند. در حالیکه در تابع ancestor فقط سطح مشخص شده در ساختار سلسله مراتبی به سمت ریشه باز گردانده می‌شد.

مرتب سازی اعضای یک ساختار سلسله مراتبی از بالا به پایین به صورت نزولی

```
select
[Measures].[Internet Sales Amount] on columns,
hierarchize(
ascendants(
[Customer].[Customer Geography].[Customer].[Crystal Zheng]
)
) on rows
From [Adventure Works]
```

Messages | Results

	Internet Sales Amount
All Customers	$29,358,677.22
France	$2,644,017.71
Yveline	$268,664.80
Versailles	$102,657.25
78000	$102,657.25
Crystal Zheng	$60.47

برای مرتب سازی یک ساختار سلسله مراتبی از تابع hierarchize() استفاده می‌کنیم. دقت داشته باشید که کار برروی ساختارهای سلسله مراتبی یکی از قسمت‌های اصلی در MDX Query ها می‌باشد.

ادامه‌ی این مطالب و کار با ساختارهای سلسله مراتبی را در مقالات بعدی دنبال خواهیم کرد.

عنوان: **آموزش MDX Query - قسمت نهم - واکشی اعضا در ساختار سلسله مراتبی دایمنشن ها**
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۲ ۲۳:۳۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

در این قسمت می‌خواهیم بیشتر در خصوص توابع مرتبط با ساختار سلسله مراتبی صحبت کنیم. برای آشنایی با این توابع و امکانات MDX Query ، مقاله را با بررسی چندین Query دنبال خواهیم کرد.

بدست آوردن تمامی برادران یک سطح خاص :

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].parent.children on rows
From [Adventure Works]
```

در کوئری بالا تمامی مشتریانی را که دارای کد پستی مشابه با کد پستی [Crystal Zheng]. می‌باشند، واکشی کرده ایم.

به عبارت دیگر با اعمال [Crystal Zheng].parent ، به کد پستی مشتری دسترسی پیدا کرده ایم (برای درک بیشتر در زیر ساختار سلسله مراتبی موقعیت جغرافیایی مشتریان را ببینید) و سپس با اعمال .children به تمامی مشتریان موجود در آن کد پستی رسیده ایم؛ که عملا همان برادران [Crystal Zheng] می باشند.

نتیجه کوئری بالا در زیر نمایش داده شده است

	Internet Sales Amount
Abby Sandberg	$5,948.23
Adam Hayes	$597.95
Alexa Peterson	$3,350.42
Blake Roberts	$8.99
Caleb Washington	$1,145.48
Carolyn Sanchez	$2,419.06
Chloe L. Ross	$2,049.10
Christian C. Davis	$7.28
Clayton C. Gao	$89.97
Colin M. Zheng	$38.98
Crystal Zheng	$60.47
Dana Alvarez	$2,803.52
Darren E. Sai	$2,420.34
David B. Martin	$42.97

راه بهتر برای بدست آوردن تمامی برادران یک سطح، استفاده از تابع siblings می‌باشد.

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].siblings on rows
From [Adventure Works]
```

کوئری‌های بالا جواب یکسانی را بر می‌گردانند. به عبارت دیگر تابع siblings عملا کار دو تابع parent.children را انجام میدهد

	Internet Sales Amount
Abby Sandberg	$5,948.23
Adam Hayes	$597.95
Alexa Peterson	$3,350.42
Blake Roberts	$8.99
Caleb Washington	$1,145.48
Carolyn Sanchez	$2,419.06
Chloe L. Ross	$2,049.10
Christian C. Davis	$7.28
Clayton C. Gao	$89.97
Colin M. Zheng	$38.98
Crystal Zheng	$60.47
Dana Alvarez	$2,803.52
Darren E. Sai	$2,420.34
David B. Martin	$42.97

برای بدست آوردن برادر ارشد به صورت زیر عمل می‌کنیم (اولین بچه در ساختار سلسله مراتبی)

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].parent.firstchild on rows
From [Adventure Works]
```

	Internet Sales Amount
Abby Sandberg	$5,948.23

و یا از تابع زیر استفاده می‌کنیم

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].firstsibling on rows
From [Adventure Works]
```

هر دو کوئری به جواب یکسان خواهند رسید.

و برای بدست آوردن آخرین برادر در ساختار سلسله مراتبی (برادر ته تغاری) از دو روش زیر می‌توان استفاده کرد.

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].parent.lastchild on rows
From [Adventure Works]
```

یا

```
Select
[Measures].[Internet Sales Amount] on columns,
[Customer].[Customer Geography].[Customer].[Crystal Zheng].lastsibling on rows
From [Adventure Works]
```

	Internet Sales Amount
Xavier White	$3.99

برای توضیح بیشتر می‌توان اضافه کرد که در کوئری بالا میزان فروش اینترنتی را برای آخرین مشتری در موقعیت جغرافیایی مشتری با نام [Crystal Zheng] واکشی شده است.

حال تصور کنید بخواهیم میزان فروش اینترنتی را برای تمامی مشتریان ایالت [Yveline] بدست بیاوریم. در این صورت MDX Query به شکل زیر خواهد بود

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[State-Province].[Yveline]
,[Customer].[Customer Geography].[Customer]
)on rows
From [Adventure Works]
```

تابع descendants دارای دو پارامتر می‌باشد. اولی برای مشخص نمودن شروع و مبدا در ساختار سلسله مراتبی و دومین برای مشخص کردن سطح واکشی در ساختار سلسله مراتبی می‌باشد. به عبارت دیگر در کوئری بالا تمامی زاد و رود ایالت [Yveline] در سطح شهر واکشی شده است و میزان فروش اینترنتی آن نمایش داده شده است.

در زیر یک کوئری ترکیبی با استفاده از دو تابع ancestor و descendants نوشته شده است.

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
ancestor(
[Customer].[Customer Geography].[Customer].[Crystal Zheng],
[Customer].[Customer Geography].[State-Province]
)
,[Customer].[Customer Geography].[Customer]
)on rows
From [Adventure Works]
```

در اینجا ابتدا جد یک مشتری در سطح ایالت بدست آمده سپس زاد و رود آن در سطح مشتری بدست می آید .

	Internet Sales Amount
Abby P. Rana	$36.96
Aidan Washington	$548.98
Alexa Morris	$553.97
Alison Sharma	$60.47
Anne Alonso	$597.95
Beth R. Alonso	$5,923.23
Billy L. Jiménez	$39.98
Bryan E. Bailey	$3,010.03
Charles J. Bell	$7.28
Charles M. Smith	$36.27
Christopher R. Clark	$866.31

برای بدست آوردن فروش اینترنتی تمامی شهرهای کشور فرانسه می‌توانیم به صورت زیر عمل کنیم.

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[Country].[France],
[Customer].[Customer Geography].[City]
) on rows
From [Adventure Works]
```

	Internet Sales Amount
Saint Ouen	$34,441.73
Les Ulis	$181,244.73
Morangis	$56,432.84
Verrieres Le Buisson	$41,619.61
Colomiers	$54,641.72
Aujan Mourede	(null)
Boulogne-Billancourt	$14,289.24
Colombes	$90,268.51
Courbevoie	$38,809.63
Paris La Defense	$45,350.86
Sèvres	$39,598.20

تابع descendants دارای یک پارامتر سوم هم می‌باشد که مشخص کننده‌ی میزان واکشی سطوح می‌باشد و به صورت پیش فرض Self می‌باشد. بنابر این کوئری بالا و پایین ، نتیجه یکسان خواهند داشت

```
Select
```

```
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[Country].[France],
[Customer].[Customer Geography].[City],
self
)on rows
From [Adventure Works]
```

	Internet Sales Amount
Saint Ouen	$34,441.73
Les Ulis	$181,244.73
Morangis	$56,432.84
Verrieres Le Buisson	$41,619.61
Colomiers	$54,641.72
Aujan Mourede	(null)
Boulogne-Billancourt	$14,289.24
Colombes	$90,268.51
Courbevoie	$38,809.63
Paris La Defense	$45,350.86
Sèvres	$39,598.20

حال اگر بخواهیم فروش اینترنتی را برای تمامی زاد و رود کشور فرانسه از سطح شهر به پایین واکشی کنیم داریم :

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[Country].[France],
[Customer].[Customer Geography].[City],
self_and_after
) on rows
From [Adventure Works]
```

در این حالات تمامی زاد و رود کشور فرانسه از سطح شهر به پایین در خروجی قرار می گیرد . به این صورت که ابتدا اولین شهر می آید؛ سپس اولین کد پستی در آن شهر و بعد تمامی مشتری های آن کد پستی و بعد کد پستی بعدی و ...

	Internet Sales Amount
Saint Ouen	$34,441.73
17490	$34,441.73
Brad Nath	$48.97
Bridget C. Nath	$1,560.43
Chelsea Jordan	$1,735.98
Clayton Zhang	$2,049.10
Colin R. Wang	$47.97
Craig R. Gill	$23.78
Darren A. Ruiz	$43.46
Dominic A. Garcia	$1,382.97
Edwin Zhou	$2,753.82
Jon Cai	$88.97
Julie Pal	$2,778.52
Katie Shan	$8.99
Levi W. Rodriguez	$4.99
Nancy Schmidt	$5,033.14
Peter D. She	$691.91
Russell A. Sharma	$6,999.57
Sabrina Serrano	$3,216.90
Shaun N. Andersen	$13.98
Theresa H. Alvarez	$5,923.23
Tony K. Raji	$7.28
Wesley Sun	$27.77
Les Ulis	$181,244.73
91940	$181,244.73
Alicia D. Shen	$9,458.18
Alvin K. Liu	$786.33

به دست آوردن تمامی زاد و رود فرانسه از سطح بعد از شهر .

به عبارت دیگر ، خروجی باتوجه به ساختار سلسله مراتبی تعریف شده عبارت است از کد پستی و تمام مشتریان آن کد پستی و سپس کد پستی بعدی .

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[Country].[France],
[Customer].[Customer Geography].[City],
after
)on rows
From [Adventure Works]
```

در کوئری فوق، خود شهر در خروجی نمایش داده نمی‌شود.

	Internet Sales Amount
17490	$34,441.73
Brad Nath	$48.97
Bridget C. Nath	$1,560.43
Chelsea Jordan	$1,735.98
Clayton Zhang	$2,049.10
Colin R. Wang	$47.97
Craig R. Gill	$23.78
Darren A. Ruiz	$43.46
Dominic A. Garcia	$1,382.97
Edwin Zhou	$2,753.82
Jon Cai	$88.97
Julio Bal	$2,778.52

به دست آوردن زاد و رود فرانسه تا یک سطح قبل از شهر .

در این حالت فرانسه و تمامی ایالت های آن در خروجی آورده می شود .

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[Country].[France],
[Customer].[Customer Geography].[City],
before
)on rows
From [Adventure Works]
```

	Internet Sales Amount
France	$2,644,017.71
Charente-Maritime	$34,441.73
Essonne	$279,297.18
Garonne (Haute)	$54,641.72
Gers	(null)
Hauts de Seine	$263,416.19
Loir et Cher	$21,473.74
Loiret	$91,562.91
Moselle	$94,046.23
Nord	$391,400.20
Pas de Calais	$11,342.92
Seine (Paris)	$539,725.80

همچنین می‌توان دومین پارامتر تابع را به صورت عدد وارد کرد و این عدد بیانگر تعداد سطح پایین‌تر از پارامتر اول در ساختار سلسله مراتبی می‌باشد.

به عنوان مثال :

```
Select
[Measures].[Internet Sales Amount] on columns,
descendants(
[Customer].[Customer Geography].[Country].[France],
2,
before
) on rows
From [Adventure Works]
```

در این حالت فرانسه و تمامی ایالت های آن در خروجی قرار می گیرد .

در ابتدا دو سطح ار کشور پایین می رویم و به شهر می رسیم و بعد زاد و رود فرانسه تا یکی قبل از شهر را بر می گرداند .

	Internet Sales Amount
France	$2,644,017.71
Charente-Maritime	$34,441.73
Essonne	$279,297.18
Garonne (Haute)	$54,641.72
Gers	(null)
Hauts de Seine	$263,416.19
Loir et Cher	$21,473.74
Loiret	$91,562.91
Moselle	$94,046.23
Nord	$391,400.20
Pas de Calais	$11,342.92
Seine (Paris)	$539,725.80

در قسمت‌های بعدی در خصوص دیگر توابع مرتبط با ساختارهای سلسله مراتبی، توضیحاتی را ارایه خواهم کرد.

عنوان: آموزش MDX Query - قسمت دهم – ادامه کار برروی ساختار های سلسله مراتبی و کار با تابع Cousin و ایجاد Range
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۶ ۱۶:۵
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

در این قسمت در خصوص توابع مرتبط با ساختار سلسله مراتبی صحبت خواهد شد.

```
Select
{
  [Date].[Calendar].[Calendar Quarter].[Q1 CY 2006],
  cousin(
[Date].[Calendar].[Calendar Quarter].[Q1 CY 2006],
[Date].[Calendar].[Calendar Year].[CY 2007]
)
} on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

تابع عمو زاده به این صورت کار می کند که دو پارامتر می گیرد . پارامتر اول سطح فعلی را مشخص می کند . پارامتر دوم سطح بالاتر از سطح اول را مشخص می کند در ساختار سلسله مراتبی و خروجی برابر است با سطحی برابر سطح پارامتر اول در زیر مجموعه ی پارامتر دوم و هم تراز پارامتر اول .

خروجی به صورت زیر می‌باشد:

خوب حالا به ساختار زیر دقت کنید (ساختار سلسله مراتبی Date)

Date
Calendar
Date.Calendar Quarter of Year
Date.Calendar Semester of Year
Date.Calendar Week of Year
Date.Calendar Year
Date.Calendar
Members
Calendar Year
Member Properties
CY 2005
CY 2006
CY 2007
CY 2008
CY 2010
Calendar Semester
Calendar Quarter
Member Properties
Q3 CY 2005
Q4 CY 2005
Q1 CY 2006
Q2 CY 2006
Q3 CY 2006
Q4 CY 2006
Q1 CY 2007
Q2 CY 2007
Q3 CY 2007
Q4 CY 2007
Q1 CY 2008
Q2 CY 2008
Q3 CY 2008
Q4 CY 2010
Month
Date

همانطور که مشخص می‌باشد تاریخ‌ها از 2005 تا 2008 و سال 2010 می‌باشند و فصول عبارتند از دو فصل پایانی سال 2005 و تمامی فصول سال 2006 و 2007 و سه فصل اول سال 2008 و فصل چهارم سال 2010 . حال دوباره به کوئری نوشته شده دقت کنید. در کوئری بالا فصل همسطح فصل اول سال 2006 در سال 2007 مورد واکشی قرار گرفته است که همان فصل اول در سال 2007 می‌باشد.

حال به بررس کوئری زیر خواهیم پرداخت:

```
Select
{
  [Date].[Calendar].[Calendar Quarter].[Q1 CY 2006],
  cousin(
[Date].[Calendar].[Calendar Quarter].[Q1 CY 2006],
[Date].[Calendar].[Calendar Semester].[H2 CY 2006]
)
} on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

در این کوئری ما ابتدا ستون فصل اول سال 2006 را بر می گردانیم . سپس در تابع پسر عمو در نیم فصل دوم سال 2006 به دنبال هم سطح فصل اول 2006 می گردیم .

نمودار درختی زیر توضیح کاملی به ما خواهد داد:

حال برای ادامه‌ی مطلب کار بر روی ساختارهای سلسله مراتبی، ابتدا باید در خصوص نحوه‌ی ایجاد Range توضیحاتی ارایه گردد. دو کوئری زیر را در نظر گرفته و خروجی آنها را با هم مقایسه نمایید

```
Select
{
  [Date].[Calendar].[Calendar Quarter].[Q1 CY 2006],
  [Date].[Calendar].[Calendar Quarter].[Q2 CY 2006],
  [Date].[Calendar].[Calendar Quarter].[Q3 CY 2006]
} on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

و

```
Select
[Date].[Calendar].[Calendar Quarter].[Q1 CY 2006]:
[Date].[Calendar].[Calendar Quarter].[Q3 CY 2006]
on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

خروجی‌ها به صورت زیر می‌باشد :

Messages | Results

	Q1 CY 2006	Q2 CY 2006	Q3 CY 2006
Reseller Sales Amount	$4,069,186.04	$4,153,820.42	$8,880,239.44

و

	Q1 CY 2006	Q2 CY 2006	Q3 CY 2006
Reseller Sales Amount	$4,069,186.04	$4,153,820.42	$8,880,239.44

مشخص می‌باشد که از علامت <:> برای ایجاد یک محدوده و جلوگیری از تولید کدهای بلند و طولانی استفاده می‌شود.

حال کوئری زیر را اجرا کنید:

```
Select
[Date].[Calendar].[Calendar Quarter].[Q1 CY 2006]
:
cousin(
[Date].[Calendar].[Calendar Quarter].[Q1 CY 2006],
[Date].[Calendar].[Calendar Semester].[H2 CY 2006]
    ) on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

در این کوئری در ابتدا تابع پسر عمو اجرا می گردد، سپس تابع رنج اجرا می گردد و در نتیجه، فاصله ی بین Q1 CY 2006 تا Q3 CY 2006 را بدست می‌آورد.

نمودار درختی زیر توضیح کاملی به ما خواهد داد :

خروجی به صورت زیر می‌باشد

Messages | Results

	Q1 CY 2006	Q2 CY 2006	Q3 CY 2006
Reseller Sales Amount	$4,069,186.04	$4,153,820.42	$8,880,239.44

در قسمت‌های بعدی دیگر توابع MDX Queryها را بررسی میکنیم.

عنوان: آموزش MDX Query - قسمت یازدهم – استفاده از توابع Lead و Lag
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۶ ۱۶:۳۵
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

از دو تابع Lead و Lag برای واکشی فرزندان، در همان سطح استفاده می‌شود. به عبارت دیگر این دو تابع می‌توانند برادران قبل و بعد را پیدا کنند.

استفاده از تابع Lead() :

```
Select
{
  [Date].[Calendar].[Calendar Quarter].[Q1 CY 2006],
  [Date].[Calendar].[Calendar Quarter].[Q1 CY 2006].lead(2)
}on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

این تابع، دو سطح بعد از نیم فصل اول 2006 را در خروجی می آ ورد. به عبارت دیگر نیم فصل سوم 2006 را بازگشت می‌دهد .

خروجی به صورت زیر می‌باشد.

استفاده از تابع Lag() :

```
Select
{
  [Date].[Calendar].[Calendar Quarter].[Q1 CY 2006],
  [Date].[Calendar].[Calendar Quarter].[Q1 CY 2006].lag(-2)
} on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

این تابع برعکس تابع Lead عمل می کند، ولی عدد منفی، کارآیی آن را مشابه تابع Lead می کند . به عبارت دیگر این تابع با عدد مثبت، در همان سطح ، فرزندان قبلی را پیدا کرده و با عدد منفی، فرزندان بعدی را پیدا می‌کند.
استفاده از تابع Lead با پارامتر منفی مشابه تابع Lag با پارامتر مثبت می‌باشد.

```
Select
{
  [Date].[Calendar].[Calendar Quarter].[Q3 CY 2006].lead(-2),
  [Date].[Calendar].[Calendar Quarter].[Q3 CY 2006]
}on columns,
[Measures].[Reseller Sales Amount]on rows
From [Adventure Works]
```

یا

```
Select
{
  [Date].[Calendar].[Calendar Quarter].[Q3 CY 2006].lag(2),
  [Date].[Calendar].[Calendar Quarter].[Q3 CY 2006]
}on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

در کوئری زیر یک حالت ترکیبی بین عملگر : و تابع حرکت در فرزندان در سطح ی مشابه را مشاهده می‌کنیم.

```
Select
[Date].[Calendar].[Calendar Quarter].[Q1 CY 2006]
:
[Date].[Calendar].[Calendar Quarter].[Q1 CY 2006].lead(2)
on columns,
[Measures].[Reseller Sales Amount] on rows
From [Adventure Works]
```

این نمودار به درک بهتر این مثال کمک خواهد کرد

خروجی به شکل زیر می‌باشد.

100 %

Messages | Results

	Q1 CY 2006	Q2 CY 2006	Q3 CY 2006
Reseller Sales Amount	$4,069,186.04	$4,153,820.42	$8,880,239.44

در قسمت‌های بعدی در خصوص توابع Head و Filter توضیح خواهم داد.

در ادامه به بررسی توابع Head , Filter , TopCount و tail می‌پردازیم

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} on columns,
head(
[Customer].[Customer Geography].[Country],
2
)on rows
From [Adventure Works]
```

تابع Head، تعداد مشخص شده بر اساس پارامتر اول از آن محور را بر اساس نحوه‌ی نمایش تنظیم شده در SSAS، واکشی می‌کند.

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Australia	$9,061,000.58	$724,880.07
Canada	$1,977,844.86	$158,227.59

حال تصور کنید بخواهیم شرط زیر را بر روی کوئری بالا اجرا کنیم

```
( [Measures].[Internet Sales Amount] >= '2500000' )
```

به عبارت دیگر ما می‌خواهیم دو کشوری را انتخاب کنیم که میزان فروش اینترنتی آنها بالای 2500000 باشد.

کوئری مشابه زیر می‌باشد

```
Select  {
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} on columns,
head(
[Customer].[Customer Geography].[Country],
2
)on rows
From [Adventure Works]
Where
( [Measures].[Internet Sales Amount] >= '2500000' )
```

البته خطای زیر را خواهیم داشت.

```
100 %
Messages
Executing the query ...
The WHERE clause function expects a tuple set expression for the  argument. A string or numeric expression was used.
Execution complete
```

به یاد داشته باشیم در صورتیکه بخواهیم ایجاد محدودیت در نمایش داده‌ها را در یک محور داشته باشیم، باید از تابع Filter استفاده کنیم؛ به صورت زیر:

```
Select
Filter(
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} ,
[Measures].[Internet Sales Amount] >= 2644017.71
  ) on columns,
head(
[Customer].[Customer Geography].[Country],
3
)on rows
From [Adventure Works]
```

تابع Filter دو پارامتر می گیرد. پارامتر اول نام ردیف یا ستونی می باشد که روی آن می خواهیم عمل فیلتر را انجام دهیم. پارامتر دوم شرط فیلترینگ می باشد که می بایست مانند T/SQL دارای یک خروجی Boolean باشد

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Australia	$9,061,000.58	$724,880.07
Canada	$1,977,844.86	$158,227.59
France	$2,644,017.71	$211,521.42

همچنان نتیجه درست نمی‌باشد ! چرا؟

اگر بخواهیم شرط روی Axis ردیف (کشور ها) اعمال گردد، باید عملیات فیلترینگ در این Axis انجام شود . بنابر این خروجی بدست آمده صحیح نمی باشد زیرا ما عملیات فیلترینگ را روی ستون ها انجام داده ایم.

کوئری زیر را اجرا نمایید

```
Select {
[Measures].[Internet Sales Amount]
,[Measures].[Internet Tax Amount]
  }
on columns,
head(
Filter(
```

```
  [Customer].[Customer Geography].[Country] ,
  [Measures].[Internet Sales Amount] >= 2644017.71
),
3)
on rows
From [Adventure Works]
```

البته توجه کنید که این کوئری، سه کشور اول که در شرط زیر قرار دارند را بر می گرداند و الزاما این سه کشور از تمام کشور های دیگر بیشتر نمی باشند .

	Internet Sales Amount	Internet Tax Amount
Australia	$9,061,000.58	$724,880.07
France	$2,644,017.71	$211,521.42
Germany	$2,894,312.34	$231,544.99

در این حالت سه کشور که بالاتر از مقدار ذکر شده، فروش اینترنتی دارند، در خروجی قرار می گیرند . البته این سه کشور دارای بالاترین فروش نمی باشند بلکه به ترتیب اسم، از بالا گزینش انجام شده است و بعد از پیدا کردن سه کشور که در شرط قرار بگیرند، جستجو تمام شده است .

اگر بخواهیم سه کشوری را که بالاترین میزان فروش را دارند پیدا کنیم و شرط هم همواره اعمال گردد، کوئری زیر درست می باشد:

```
Select {
[Measures].[Internet Sales Amount]
,[Measures].[Internet Tax Amount]
  }
on columns,
TopCount(
Filter(
  [Customer].[Customer Geography].[Country] ,
  [Measures].[Internet Sales Amount] >= 2644017.71
  ),
3, [Measures].[Internet Sales Amount])
on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
United States	$9,389,789.51	$751,183.18
Australia	$9,061,000.58	$724,880.07
United Kingdom	$3,391,712.21	$271,336.98

در این حالت به جای تابع Head از تابع TopCount استفاده گردیده است . این تابع سه کشوری را که بیشترین فروش اینترنتی را داشته اند و این فروش بالاتر از مقدار ذکر شده در شرط می باشد را بر می گرداند . البته در اینجا تابع topcount دارای سه پارامتر می باشد و در پارامتر سوم اعلام میکند که تعداد بالای مجموعه براساس چه شاخصی باید به دست بیاید.

حال اگر بخواهیم سه ردیف انتهایی جدول را واکشی کنیم داریم:

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
}on columns,
tail([Customer].[Customer Geography].[Country],
3)on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Germany	$2,894,312.34	$231,544.99
United Kingdom	$3,391,712.21	$271,336.98
United States	$9,389,789.51	$751,183.18

این تابع برعکس تابع Head کار میکند و N ردیف آ اخر مجموعه را بدست می آورد . البته در بالا فقط 3 ردیف انتهایی را در خروجی آورده ایم و هیچ شرطی اعمال نگردیده است.

عنوان: آموزش MDX Query - قسمت سیزدهم - برخی توابع برای کار روی ساختار های سلسله مراتبی (prevmember و nextmember)
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۰۸ ۱۸:۱۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

کار با توابع prevmember و nextmember

قبل از اجرای کوئری‌های زیر در ابتدا به ساختار سلسله مراتبی Customer دقت نمایید و ترتیب کشورها را در این ساختار بررسی نمایید.

کوئری زیر را اجرا نمایید :

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
}on columns,
{
[Customer].[Customer Geography].[Country].[Germany],
[Customer].[Customer Geography].[Country].[Germany].prevmember
}on rows
From [Adventure Works]
```

این تابع برادر قبلی را بدست می آ ورد

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Germany	$2,894,312.34	$231,544.99
France	$2,644,017.71	$211,521.42

حال کوئری زیر را اجرا نمایید :

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} on columns,
{
[Customer].[Customer Geography].[Country].[Germany],
[Customer].[Customer Geography].[Country].[Germany].lag(1)
} on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Germany	$2,894,312.34	$231,544.99
France	$2,644,017.71	$211,521.42

در اینجا میزان فروش اینترنتی و همچنین میزان مالیات اینترنتی برای مشتریان آلمان و کشور قبلی بدست می‌آید.

با این تابع می توان برادر قبلی را با اعلام یک فاصله بدست آورد. مثلا 4 برادر قبلی یا .... را توسط این تابع بدست آورد .

کوئری زیر را اجرا نمایید :

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} on columns,
{
[Customer].[Customer Geography].[Country].[Germany],
[Customer].[Customer Geography].[Country].[Germany].prevmember.prevmember
} on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Germany	$2,894,312.34	$231,544.99
Canada	$1,977,844.86	$158,227.59

در این حالت مشابه تابع Lag(2) عمل کرده ایم. حال با استفاده از تابع Nextmember می توانیم برادر بعدی را بدست بیاوریم.

کوئری زیر را اجرا کنید :

```
select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} on columns,
{
[Customer].[Customer Geography].[Country].[Germany],
[Customer].[Customer Geography].[Country].[Germany].nextmember
} on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Germany	$2,894,312.34	$231,544.99
United Kingdom	$3,391,712.21	$271,336.98

و همچنین در کوئری زیر برادر بعد از برادر بعدی را بدست آورده ایم

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} on columns,
{
[Customer].[Customer Geography].[Country].[Germany],
[Customer].[Customer Geography].[Country].[Germany].nextmember.nextmember
} on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Germany	$2,894,312.34	$231,544.99
United States	$9,389,789.51	$751,183.18

یک ترکیب از عملگر Range , NextMember در کوئری زیر نوشته شده است.

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} on columns,
[Customer].[Customer Geography].[Country].[Germany]
:
[Customer].[Customer Geography].[Country].[Germany].nextmember.nextmember
on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Germany	$2,894,312.34	$231,544.99
United Kingdom	$3,391,712.21	$271,336.98
United States	$9,389,789.51	$751,183.18

کاربرد تابع Lead برای به دست آوردن برادر بعدی بر اساس عددی می باشد که به آن داده ایم .

در اینجا ترکیبی از رنج و به دست آوردن برادر برادر بعدی را داریم .

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Internet Tax Amount]
} on columns,
[Customer].[Customer Geography].[Country].[Germany]
:
[Customer].[Customer Geography].[Country].[Germany].lead(2)
on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Internet Tax Amount
Germany	$2,894,312.34	$231,544.99
United Kingdom	$3,391,712.21	$271,336.98
United States	$9,389,789.51	$751,183.18

در قسمت بعدی تابع Order را بررسی می‌کنیم.

در زیر در ابتدا یک کوئری می‌نویسیم و سپس بررسی می‌کنیم که چگونه می‌توان خروجی را مرتب کنیم.

```
Select
[Measures].[Internet Sales Amount]on columns,
[Product].[Product Categories].[Subcategory] on rows
From [Adventure Works]
```

کوئری بالا میزان فروش اینترنتی را بر اساس دسته بندی محصولات واکشی می‌کند

100 %

Messages | Results

	Internet Sales Amount
Bike Racks	$39,360.00
Bike Stands	$39,591.00
Bottles and Cages	$56,798.19
Cleaners	$7,218.60
Fenders	$46,619.58
Helmets	$225,335.60
Hydration Packs	$40,307.67
Lights	(null)
Locks	(null)
Panniers	(null)
Pumps	(null)

در کوئری زیر لیست فروش، براساس میزان فروش اینترنتی مرتب شده است :

```
Select
[Measures].[Internet Sales Amount] on columns,
order(
[Product].[Subcategory].[Subcategory],
[Measures].[Internet Sales Amount]
)on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Pedals	(null)
Pumps	(null)
Road Frames	(null)
Saddles	(null)
Tights	(null)
Touring Frames	(null)
Wheels	(null)
Socks	$5,106.32
Cleaners	$7,218.60
Caps	$19,688.10
Gloves	$35,020.70
Vests	$35,687.00
Bike Racks	$39,360.00
Bike Stands	$39,591.00
Hydration Packs	$40,307.67
Fenders	$46,619.58
Bottles and Cages	$56,798.19
Shorts	$71,319.81
Jerseys	$172,950.68
Helmets	$225,335.60

تابع Order، ستون یا ردیف خاصی را بر اساس ستون یا ردیف دیگری مرتب می کند . این تابع بر روی کل یک Axis اعمال میگردد و دو پارامتر دارد .

پارامتر دوم مشخص می کند مرتب سازی پارامتر اول بر اساس چه شاخصی باشد .

همچنین در کوئری زیر نحوه‌ی مرتب کردن به صورت نزولی را خواهیم دید.

```
Select
[Measures].[Internet Sales Amount] on columns,
order(
[Product].[Subcategory].[Subcategory],
[Measures].[Internet Sales Amount],
desc
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Road Bikes	$14,520,584.04
Mountain Bikes	$9,952,759.56
Touring Bikes	$3,844,801.05
Tires and Tubes	$245,529.32
Helmets	$225,335.60
Jerseys	$172,950.68
Shorts	$71,319.81
Bottles and Cages	$56,798.19
Fenders	$46,619.58
Hydration Packs	$40,307.67
Bike Stands	$39,591.00
Bike Racks	$39,360.00
Vests	$35,687.00
Gloves	$35,020.70
Caps	$19,688.10
Cleaners	$7,218.60
Socks	$5,106.32
Bib-Shorts	(null)
Bottom Brackets	(null)

مرتب سازی به صورت صعودی به شکل زیر می‌باشد

```
Select
[Measures].[Internet Sales Amount] on columns,
order(
[Product].[Subcategory].[Subcategory],
[Measures].[Internet Sales Amount],
asc
) on rows
From [Adventure Works]
```

Mountain Frames	(null)
Panniers	(null)
Pedals	(null)
Pumps	(null)
Road Frames	(null)
Saddles	(null)
Tights	(null)
Touring Frames	(null)
Wheels	(null)
Socks	$5,106.32
Cleaners	$7,218.60
Caps	$19,688.10
Gloves	$35,020.70

از آنجایی که مرتب سازی به صورت پیش فرض صعودی می باشد، ب کار بردن این کلید واژه ، خیلی لازم نمی باشد .

رویکرد تابع Order با ساختار سلسله مراتبی

برای شروع کوئری زیر را اجرا می‌کنیم.

```
Select
[Measures].[Internet Sales Amount] on columns,
order(
[Product].[Product Categories].[Subcategory],
[Measures].[Internet Sales Amount],
asc
) on rows
From [Adventure Works]
```

Bib-Shorts	(null)
Tights	(null)
Socks	$5,106.32
Caps	$19,688.10
Gloves	$35,020.70
Vests	$35,687.00
Shorts	$71,319.81
Jerseys	$172,950.68
Lights	(null)
Locks	(null)
Panniers	(null)
Pumps	(null)
Cleaners	$7,218.60
Bike Racks	$39,360.00
Bike Stands	$39,591.00
Hydration Packs	$40,307.67
Fenders	$46,619.58

البته در این حالت با نگاه کردن به خروجی متوجه میشوید که عملیات مرتب سازی به درستی انجام نشده است و دلیل این اتفاق رویکرد تابع مرتب سازی به ساختار سلسله مراتبی میباشد . عملا تابع مرتب سازی احترام ویژه ای به ساختار سلسله می گزارد .

حال همان کوئری را به صورت زیر اجرا می‌کنیم :

```
Select
[Measures].[Internet Sales Amount] on columns,
order(
[Product].[Product Categories].[Subcategory],
[Measures].[Internet Sales Amount],
basc
) on rows
From [Adventure Works]
```

Bib-Shorts	(null)
Tights	(null)
Bottom Brackets	(null)
Brakes	(null)
Chains	(null)
Cranksets	(null)
Derailleurs	(null)
Forks	(null)
Handlebars	(null)
Headsets	(null)
Mountain Frames	(null)
Pedals	(null)
Road Frames	(null)
Saddles	(null)
Touring Frames	(null)
Wheels	(null)
Socks	$5,106.32
Cleaners	$7,218.60
Caps	$19,688.10

حرف b در کلمه ی کلیدی basc به معنی شکستن ساختار سلسله مراتبی می باشد . در این حالت عملیات مرتب سازی بدون درنظر گرفتن ساختار سلسله مراتبی انجام می گردد .

کوئری‌های زیر تاثیر مرتب سازی نزولی و ساختار سلسله مراتبی را بررسی می‌کنند

```
Select
[Measures].[Internet Sales Amount] on columns,
order(
[Product].[Product Categories].[Subcategory],
[Measures].[Internet Sales Amount],
desc
) on rows
From [Adventure Works]
```

	Internet Sales Amount
Road Bikes	$14,520,584.04
Mountain Bikes	$9,952,759.56
Touring Bikes	$3,844,801.05
Tires and Tubes	$245,529.32
Helmets	$225,335.60
Bottles and Cages	$56,798.19
Fenders	$46,619.58
Hydration Packs	$40,307.67
Bike Stands	$39,591.00
Bike Racks	$39,360.00
Cleaners	$7,218.60
Lights	(null)
Locks	(null)
Panniers	(null)
Pumps	(null)
Jerseys	$172,950.68
Shorts	$71,319.81
Vests	$35,687.00
Gloves	$35,020.70

در این حالت مرتب سازی نزولی انجام می گردد اما با درنظر گرفتن ساختار سلسله مراتبی

```
Select
[Measures].[Internet Sales Amount] on columns,
order(
[Product].[Product Categories].[Subcategory],
[Measures].[Internet Sales Amount],
bdesc
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Road Bikes	$14,520,584.04
Mountain Bikes	$9,952,759.56
Touring Bikes	$3,844,801.05
Tires and Tubes	$245,529.32
Helmets	$225,335.60
Jerseys	$172,950.68
Shorts	$71,319.81
Bottles and Cages	$56,798.19
Fenders	$46,619.58
Hydration Packs	$40,307.67
Bike Stands	$39,591.00
Bike Racks	$39,360.00
Vests	$35,687.00
Gloves	$35,020.70
Caps	$19,688.10
Cleaners	$7,218.60
Socks	$5,106.32
Lights	(null)

در این حالت عمل مرتب سازی نزولی بدون درنظر گرفتن ساختار سلسله مراتبی انجام میگردد .

می توان مرتب سازی را بر اساس یک ستون که در کوئری ذکر نشده است انجام داد . در این حالت نتیجه در ظاهر مرتب نمی باشد .

```
Select
[Measures].[Internet Sales Amount] on columns,
order(
[Product].[Product Categories].[Subcategory],
[Measures].[Reseller Sales Amount],
bdesc
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Road Bikes	$14,520,584.04
Mountain Bikes	$9,952,759.56
Touring Bikes	$3,844,801.05
Mountain Frames	(null)
Road Frames	(null)
Touring Frames	(null)
Wheels	(null)
Jerseys	$172,950.68
Shorts	$71,319.81
Helmets	$225,335.60
Vests	$35,687.00
Gloves	$35,020.70
Cranksets	(null)
Tights	(null)
Bike Racks	$39,360.00
Handlebars	(null)

در این کوئری دو ستون وجود دارد اما مرتب سازی براساس یکی از ستون ها انجام می شود

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
order(
[Product].[Product Categories].[Subcategory],
[Measures].[Reseller Sales Amount],
bdesc
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Road Bikes	$14,520,584.04	$29,358,206.96
Mountain Bikes	$9,952,759.56	$26,492,684.38
Touring Bikes	$3,844,801.05	$10,451,490.22
Mountain Frames	(null)	$4,713,672.15
Road Frames	(null)	$3,849,853.34
Touring Frames	(null)	$1,642,327.69
Wheels	(null)	$679,070.07
Jerseys	$172,950.68	$579,308.71
Shorts	$71,319.81	$342,202.72
Helmets	$225,335.60	$258,712.93
Vests	$35,687.00	$223,801.37
Gloves	$35,020.70	$207,775.17
Cranksets	(null)	$203,942.62
Tights	(null)	$201,833.01
Bike Racks	$39,360.00	$197,736.16
Handlebars	(null)	$170,591.32

عملیات مرتب سازی می تواند بر روی هرکدام از Axis ها جداگانه انجام گردد .

```
Select
order(
[Product].[Product Categories].[Subcategory],
[Measures].[Reseller Sales Amount],
bdesc
 ) on columns,
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on rows
From [Adventure Works]
```

100 %

Messages | Results

	Road Bikes	Mountain Bikes	Touring Bikes	Mountain Frames	Road Frames	Touring Frames	Wheels	Jerseys	Shorts
Internet Sales Amount	$14,520,584.04	$9,952,759.56	$3,844,801.05	(null)	(null)	(null)	(null)	$172,950.68	$71,319.81
Reseller Sales Amount	$29,358,206.96	$26,492,684.38	$10,451,490.22	$4,713,672.15	$3,849,853.34	$1,642,327.69	$679,070.07	$579,308.71	$342,202.72

از تابع hierarchize هم می‌توان برای عملیات مرتب سازی استفاده کرد

```
Select
[Measures].[Internet Sales Amount] on columns,
hierarchize(
```

```
{
[Product].[Subcategory].[Touring Bikes],
[Product].[Subcategory],
[Product].[Subcategory].[Mountain Bikes]
}
)on rows
From [Adventure Works]
```

برای مرتب سازی ردیف هایی که عملا ارتباطی با هم ندارند از این تابع استفاده می شود . پیش فرض مرتب سازی نزولی می باشد . البته می توانستیم همچنان از تابع Order استفاده کنیم:

```
Select
[Measures].[Internet Sales Amount] on columns,
Order(
{
[Product].[Subcategory].[Touring Bikes],
[Product].[Subcategory],
[Product].[Subcategory].[Mountain Bikes]
} , [Measures].[Internet Sales Amount] , basc
)on rows
From [Adventure Works]
```

حال می‌خواهیم ببینیم چگونه می‌توان با استفاده از تابع hierarchize مرتب سازی صعودی را انجام دهیم :

```
Select
[Measures].[Internet Sales Amount] on columns,
hierarchize(
{
[Product].[Subcategory].[Touring Bikes],
[Product].[Subcategory],
[Product].[Subcategory].[Mountain Bikes]
 },
```

```
 post
) on rows
From [Adventure Works]
```

Messages | Results

	Internet Sales Amount
Mountain Bikes	$9,952,759.56
Touring Bikes	$3,844,801.05
All Products	$29,358,677.22

همان طور که مشخص می‌باشد مرتب سازی به درستی صورت نگرفته است؟! دلیل این اتفاق اولویت دادن به ساختار سلسله مراتبی می‌باشد.

برای بررسی بیشتر ابتدا کوئری زیر را فراخوانی نمایید :

```
Select
[Measures].[Internet Sales Amount] on columns,
non empty order(
crossjoin(
[Product].[Category].[Category],
[Product].[Subcategory].[Subcategory]
 ),
[Measures].[Internet Sales Amount],
desc
) on rows
From [Adventure Works]
```

Messages | Results

		Internet Sales Amount
Accessories	Tires and Tubes	$245,529.32
Accessories	Helmets	$225,335.60
Accessories	Bottles and Cages	$56,798.19
Accessories	Fenders	$46,619.58
Accessories	Hydration Packs	$40,307.67
Accessories	Bike Stands	$39,591.00
Accessories	Bike Racks	$39,360.00
Accessories	Cleaners	$7,218.60
Bikes	Road Bikes	$14,520,584.04
Bikes	Mountain Bikes	$9,952,759.56
Bikes	Touring Bikes	$3,844,801.05
Clothing	Jerseys	$172,950.68
Clothing	Shorts	$71,319.81
Clothing	Vests	$35,687.00
Clothing	Gloves	$35,020.70
Clothing	Caps	$19,688.10
Clothing	Socks	$5,106.32

همانطور که مشخص می‌باشد، مرتب سازی براساس فروش اینترنتی و با درنظر گرفتن ساختار سلسله مراتبی صورت گرفته است. برای مرتب سازی بدون در نظر گرفتن ساختار سلسله مراتبی کوئری زیر را بنویسید :

```
Select
[Measures].[Internet Sales Amount] on columns,
non empty order(
crossjoin(
[Product].[Category].[Category],
[Product].[Subcategory].[Subcategory]
),
[Measures].[Internet Sales Amount],
bdesc
) on rows
From [Adventure Works]
```

Messages | Results

		Internet Sales Amount
Bikes	Road Bikes	$14,520,584.04
Bikes	Mountain Bikes	$9,952,759.56
Bikes	Touring Bikes	$3,844,801.05
Accessories	Tires and Tubes	$245,529.32
Accessories	Helmets	$225,335.60
Clothing	Jerseys	$172,950.68
Clothing	Shorts	$71,319.81
Accessories	Bottles and Cages	$56,798.19
Accessories	Fenders	$46,619.58
Accessories	Hydration Packs	$40,307.67
Accessories	Bike Stands	$39,591.00
Accessories	Bike Racks	$39,360.00
Clothing	Vests	$35,687.00
Clothing	Gloves	$35,020.70
Clothing	Caps	$19,688.10
Accessories	Cleaners	$7,218.60
Clothing	Socks	$5,106.32

در این حالت مرتب سازی بدون در نظر گرفتن ساختار سلسله مراتب انجام می گردد .

در دو کوئری فوق از تابع Non empty برای حذف ردیف‌های Null استفاده شده است.

عنوان: آموزش MDX Query - قسمت پانزدهم - اعمال شرط بر روی خروجی عمل واکشی
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۱۳ ۱۰:۱۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

در این قسمت می‌خواهیم بیشتر روی مفاهیم اعمال شرط بر روی خروجی عمل واکشی کار کنیم. برای شروع کوئری ساده‌ی زیر را اجرا و خروجی آن را تفسیر می‌کنیم.

```
Select
From [Adventure Works]
```

همان طور که مشاهده می‌کنید، خروجی یک عدد می‌باشد. بدون نام ستون یا ردیف؟!

به‌خاطر بیاورید که هر Cube در SSAS دارای یک Measure پیش فرض بود که در صورت عدم اعلام نام یک Measure در کوئری، SSAS به صورت پیش فرض مقدار این Measure را بر می‌گرداند. خوب؛ نام ستون و سطر چرا ذکر نشده است؟ به دلیل عدم اعلام صریح نام سطر و ستون در کوئری بالا، SSAS نام ستون و سطر خاصی را نمی‌تواند نمایش دهد.

با بررسی کوئری زیر به درک بیشتری از شاخص (Measure) پیش فرض Cube دست پیدا خواهید کرد.

```
Select
From [Adventure Works]
Where
( [Measures].[Reseller Sales Amount] )
```

خروجی همچنان مانند بالا می‌باشد اما در این حالت اعلام شده است که از کدام شاخص باید واکشی انجام شود. دلیل خروجی مشابه، یکسان بودن شاخص پیش فرض و شرط اعلام شده می‌باشد. به بیان دیگر [Measures].[Reseller Sales Amount] در [Adventure Works] به عنوان شاخص پیش فرض معرفی شده است و با اجرای کوئری زیر عملا شرط واکشی برای یک شاخص متفاوت اعمال شده است.

```
Select
From [Adventure Works]
```

```
Where [Measures].[Internet Sales Amount]
```

کوئری زیر را اجرا کنید:

```
Select
[Measures].[Internet Sales Amount] on columns
From [Adventure Works]
```

تنها تفاوت دو کوئری بالا ، در آوردن نام ستون می باشد . زیرا در هر دو سرجمع ، یک شاخص واکشی می گردد .

کوئری زیر را اجرا کنید:

```
select
[Measures].[Internet Sales Amount] on columns
From [Adventure Works]
Where [Measures].[Internet Sales Amount]
```

این کوئری با خطا مواجه می شود . زیرا در آن کوئری در یک Axis و در شرط ، اعمال انتخاب شاخص شده است که این مورد فقط می بایستی در یکی از این دو قسمت رخ دهد .

و همچنین در صورت انتخاب دو شاخص متفاوت نیز با خطا برخورد خواهیم کرد.

```
Select
[Measures].[Internet Sales Amount] on columns
From [Adventure Works]
Where [Measures].[Reseller Sales Amount]
```

به عبارت دیگر نمی‌توان در خواست فیلتر کردن کوئری را برروی شاخص 1 داد؛ در صورتیکه می‌خواهیم شاخص 2 را واکشی کنیم. اعمال شرط برای واکشی اطلاعات از شاخص، پیش فرض نوشتن این شرط لازم نمی‌باشد؛ زیرا این شاخص به صورت پیش فرض انتخاب شده‌است.

```
select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
 }on columns
From [Adventure Works]
Where [Measures].[Reseller Sales Amount]
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$571,297.93	$66,302,381.56	$1,777,840.84	$11,799,076.66	$80,450,596.98

بنابراین کوئری بالا و کوئری زیر یکسان عمل خواهند کرد:

```
select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
 }on columns
From [Adventure Works]
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$571,297.93	$66,302,381.56	$1,777,840.84	$11,799,076.66	$80,450,596.98

حال می‌خواهیم سرجمع فروش نمایندگان فروش محصولات در کشور کانادا را بر اساس دسته بندی محصولات داشته باشیم .
برای این منظور کوئری زیر را می‌نویسیم:

```
Select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
} on columns
From [Adventure Works]
Where [Customer].[Customer Geography].[Country].[Canada]
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$571,297.93	$66,302,381.56	$1,777,840.84	$11,799,076.66	$80,450,596.98

با اعمال شرط کشور کانادا، عملا خروجی فروش نمایندگان فروش در کانادا بر اساس دسته بندی محصولات واکشی می‌گردد. کمی به خروجی دقت نمایید. مبلغ سرجمع برابر مبلغ کل فروش اینترنتی می‌باشد که در کوئری‌های قبلی بدست آوردیم؟!

خروجی این کوئری مشکوک به نظر می رسد . زیرا سرجمع مبالغ فروش نمایندگان فروش برای کانادایی ها برابر کل فروش نمایندگان فروش می باشد . آیا کانادایی ها تمام خرید را انجام داده اند؟ خیر .

دلیل این اشکال در این است که هیچ گونه ارتباطی بین بعد مشتری و شاخص پیش فرض در سیستم وجود ندارد .

مشکل کوئری بالا در این کوئری با تغییر بعد در قسمت اعمال شرط برطرف شده؛ اکنون خروجی حقیقی مشاهده می شود .

```
Select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
}on columns
From [Adventure Works]
Where [Sales Territory].[Sales Territory].[Country].[Canada]
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$118,127.35	$11,636,380.59	$378,947.63	$2,244,470.02	$14,377,925.60

حال اگر بخواهیم دو شرط را به صورت همزمان داشته باشیم به صورت زیر عمل خواهیم کرد :

```
Select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
} on columns
From [Adventure Works]
Where (
[Customer].[Customer Geography].[Country].[Canada],
[Measures].[Internet Sales Amount]
)
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$103,377.85	$1,821,302.39	$53,164.62	(null)	$1,977,844.86

در کوئری بالا سرجمع فروش اینترنتی توسط مشتریان کانادایی بدست آمده است.

البته می‌توان کوئری فوق را به صورت زیر هم نوشت و در این حالت نام ردیف هم در خروجی قابل مشاهده می‌باشد و البته دیگر نیازی به اعمال شرط، روی نام شاخص نمی‌باشد. زیرا اعمال شرط در ردیف انجام شده است.

```
Select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
} on columns,
[Measures].[Internet Sales Amount] On rows
From [Adventure Works]
Where (
[Customer].[Customer Geography].[Country].[Canada]
)
```

100 %

Messages | Results

	Accessories	Bikes	Clothing	Components	All Products
Internet Sales Amount	$103,377.85	$1,821,302.39	$53,164.62	(null)	$1,977,844.86

حال اگر بخواهیم فروش اینترنتی را برای استرالیا و کانادا داشته باشیم به صورت زیر عمل می‌کنیم .

```
Select
{
[Product].[Product Categories].[Category],[Product].[Product Categories]
} on columns
From [Adventure Works]
Where
(
[Customer].[Customer Geography].[Country].[Canada],
[Customer].[Customer Geography].[Country].[Australia],
[Measures].[Internet Sales Amount]
)
```

```
100 %
Messages
Executing the query ...
Query (6, 1) The Customer Geography hierarchy is used more than once in the Crossjoin function.
Execution complete
```

در اینجا ما نیاز داریم میزان فروش اینترنتی کانادا و استرالیا را برای انواع محصولات بدست آوریم ، اما نحوه استفاده از دو ساختار سلسله مراتبی مرتبط با یک دایمنشن را درست رعایت نکرده‌ایم . بنابر این کوئری زیر را اجرا خواهیم کرد :

```
Select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
} on columns
From [Adventure Works]
Where
(
{
[Customer].[Customer Geography].[Country].[Canada],
[Customer].[Customer Geography].[Country].[Australia]
},
[Measures].[Internet Sales Amount]
)
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$242,068.48	$10,673,352.40	$123,424.57	(null)	$11,038,845.45

که همان کوئری بالا می باشد با این تفاوت که از {} استفاده شده است .

درابتدا میزان فروش نمایندگان فروش در انگلستان را بدست می‌آوریم:

```
Select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
} on columns
From [Adventure Works]
Where
[Sales Territory].[Sales Territory].[Country].[United Kingdom]
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$42,593.03	$3,405,747.21	$118,828.80	$711,839.79	$4,279,008.83

و برای بدست آوردن فروش اینترنتی تمام کشور ها به جز انگلستان بر اساس دسته بندی محصولات کوئری زیر را خواهیم نوشت :

```
Select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
} on columns
From [Adventure Works]
Where
[Sales Territory].[Sales Territory].[Country]
-
[Sales Territory].[Sales Territory].[Country].[United Kingdom]
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$528,704.90	$62,896,634.35	$1,659,012.04	$11,087,236.86	$76,171,588.16

البته از تابع Except هم می‌توان به صورت زیر استفاده کرد

```
Select
{
[Product].[Product Categories].[Category],
[Product].[Product Categories]
} on columns
From [Adventure Works]
Where
except(
[Sales Territory].[Sales Territory].[Country],
[Sales Territory].[Sales Territory].[Country].[United Kingdom]
)
```

100 %

Messages | Results

Accessories	Bikes	Clothing	Components	All Products
$528,704.90	$62,896,634.35	$1,659,012.04	$11,087,236.86	$76,171,588.16

عملگر منها مشابه except کار میکند.

عنوان: آموزش MDX Query - قسمت شانزدهم - استفاده از تابع Filter در MDX Query ها
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۰/۱۳ ۱۱:۱۵
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

در این قسمت بر روی تابع Filter در MDX Query ها تمرکز خواهیم کرد. برای آشنایی با این تابع یک سری از کوئری‌ها را اجرا کرده و به بررسی آنها می‌پردازیم.

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
[Product].[Product Categories].[Category] on rows
From [Adventure Works]
```

دقت کنید که در واکشی، مقدار فروش اینترنتی Component برابر Null می‌باشد.

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Accessories	$700,759.96	$571,297.93
Bikes	$28,318,144.65	$66,302,381.56
Clothing	$339,772.61	$1,777,840.84
Components	(null)	$11,799,076.66

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
[Measures].[Internet Sales Amount] > 0
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Accessories	$700,759.96	$571,297.93
Bikes	$28,318,144.65	$66,302,381.56
Clothing	$339,772.61	$1,777,840.84

با توجه به شرطی که اعمال شده است، فقط دسته بندی محصولاتی در خروجی می آید که فروش اینترنتی آنها بیشتر از صفر یا برابر Null نباشند.

استفاده از کلید واژه ی Having در هر محور کاری شبیه به انجام عمل فیلترینگ می باشد .

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
[Product].[Product Categories].[Category]
having [Measures].[Internet Sales Amount] > 0 on rows
From [Adventure Works]
```

*/

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Accessories	$700,759.96	$571,297.93
Bikes	$28,318,144.65	$66,302,381.56
Clothing	$339,772.61	$1,777,840.84

اگر بخواهیم میزان فروش اینترنتی و میزان فروش نمایندگان فروش را برای محصولاتی واکشی کنیم که میزان فروش اینترنتی آنها بیش از 500000 دلار می‌باشد ، کوئری زیر را خواهیم داشت :

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
[Measures].[Internet Sales Amount] > 500000
  ) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Accessories	$700,759.96	$571,297.93
Bikes	$28,318,144.65	$66,302,381.56

و برای ایجاد شرط ترکیبی بر روی شاخص، به صورت زیر عمل خواهیم کرد :

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
[Measures].[Internet Sales Amount] > 500000
and
[Measures].[Internet Sales Amount] < 750000
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Accessories	$700,759.96	$571,297.93

در مثال بالا، دسته بندی محصولاتی در خروجی واکشی شده اند که میزان فروش اینترنتی آنها بیش از 500 هزار و کمتر از 750 هزار می‌باشد.

استفاده از And , Or در شروط ترکیبی مجاز می‌باشد

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
```

```
filter(
[Product].[Product Categories].[Category],
[Measures].[Internet Sales Amount] > 750000
or
[Measures].[Internet Sales Amount] < 500000
  ) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Bikes	$28,318,144.65	$66,302,381.56
Clothing	$339,772.61	$1,777,840.84
Components	(null)	$11,799,076.66

در زیر با استفاده از And، شرط برروی میزان فروش نمایندگان فروش نیز قرارداده شده است.

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
(
[Measures].[Internet Sales Amount] > 750000
or
[Measures].[Internet Sales Amount] < 500000
)
and
[Measures].[Reseller Sales Amount] < 15000000
) on rows
From [Adventure Works]
```

در هنگام ایجاد شروط ترکیبی حتما از () استفاده کنید .

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Bikes	$28,318,144.65	$66,302,381.56
Clothing	$339,772.61	$1,777,840.84
Components	(null)	$11,799,076.66

حال می‌خواهیم دو شاخص در یک ردیف، با یکدیگر مقایسه شوند و در صورت صحت شرط، آن ردیف در خروجی قرار گیرد

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
[Measures].[Internet Sales Amount]
>
[Measures].[Reseller Sales Amount]
) on rows
From [Adventure Works]
```

ایجاد فیلترینگ با استفاده از currentmember و عملگر Is .

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
[Product].[Product Categories].currentmember
is
[Product].[Product Categories].[Category].[Bikes]
) on rows
From [Adventure Works]
```

البته در مثال بالا می توانیم به جای استفاده از Is از = استفاده کنیم. تا اینجا عمل Filtering برروی شاخص‌ها انجام شده است. اما امکان اعمال Filter روی Dimension ها نیز وجود دارد.

کوئری زیر را بررسی کنید :

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
[Product].[Product Categories].currentmember
is
[Product].[Product Categories].[Category].[Bikes]
or
[Product].[Product Categories].currentmember
is
[Product].[Product Categories].[Category].[Accessories]
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Accessories	$700,759.96	$571,297.93
Bikes	$28,318,144.65	$66,302,381.56

در کوئری بالا میزان فروش نمایندگان فروش و فروش اینترنتی برای دسته بندی‌های Bike و Accessories واکشی شده است.

امکان ایجاد شرایط ترکیبی از شاخص‌ها و بعدها در یک Filter نیز وجود دارد.

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
(
[Product].[Product Categories].currentmember
is
[Product].[Product Categories].[Category].[Bikes]
and
[Measures].[Reseller Sales Amount] > 1000000
)
or
(
[Product].[Product Categories].currentmember
is
[Product].[Product Categories].[Category].[Accessories]
)
and
```

```
[Measures].[Reseller Sales Amount] > 750000
)on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Bikes	$28,318,144.65	$66,302,381.56

همچنین می‌توان از Not درون شرط Filter استفاده کرد

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
filter(
[Product].[Product Categories].[Category],
not
(
[Product].[Product Categories].currentmember
is
[Product].[Product Categories].[Category].[Clothing]
)
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Accessories	$700,759.96	$571,297.93
Bikes	$28,318,144.65	$66,302,381.56
Components	(null)	$11,799,076.66

در زیر می‌خواهیم به بررسی تابع non Empty بپردازیم . برای این منظور در ابتدا کوئری زیر را اجرا کنید :

```
Select
{
```

```
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
[Product].[Product Categories].[Subcategory] on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Bike Racks	$39,360.00	$197,736.16
Bike Stands	$39,591.00	(null)
Bottles and Cages	$56,798.19	$7,476.60
Cleaners	$7,218.60	$11,188.37
Fenders	$46,619.58	(null)
Helmets	$225,335.60	$258,712.93
Hydration Packs	$40,307.67	$65,518.75
Lights	(null)	(null)
Locks	(null)	$16,225.22
Panniers	(null)	(null)
Pumps	(null)	$13,514.69
Tires and Tubes	$245,529.32	$925.21
Mountain Bikes	$9,952,759.56	$26,492,684.38
Road Bikes	$14,520,584.04	$29,358,206.96

با استفاده از تابع Non Empty، ردیف ها یا ستون هایی حذف می گردند که تمامی مقادیر آن‌ها ، در آن ردیف یا در آن ستون برابر Null باشند.

```
Select
{
[Measures].[Internet Sales Amount],
[Measures].[Reseller Sales Amount]
} on columns,
non empty [Product].[Product Categories].[Subcategory] on rows
From [Adventure Works]
```

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Bike Racks	$39,360.00	$197,736.16
Bike Stands	$39,591.00	(null)
Bottles and Cages	$56,798.19	$7,476.60
Cleaners	$7,218.60	$11,188.37
Fenders	$46,619.58	(null)
Helmets	$225,335.60	$258,712.93
Hydration Packs	$40,307.67	$65,518.75
Locks	(null)	$16,225.22
Pumps	(null)	$13,514.69
Tires and Tubes	$245,529.32	$925.21
Mountain Bikes	$9,952,759.56	$26,492,684.38
Road Bikes	$14,520,584.04	$29,358,206.96
Touring Bikes	$3,844,801.05	$10,451,490.22
Bib-Shorts	(null)	$166,739.71

در قسمت آینده برروی توابع Count , Top و ... تمرکز خواهیم کرد.

نظرات خوانندگان

نویسنده: سوسن
تاریخ: ۱۳۹۲/۱۰/۲۸ ۲۳:۳۸

سلام و خسته نباشید؛ یه سوال داشتم: کوئری‌های mdx پروژه‌ی SSAS م رو به جای اینکه با استفاده از reporting service بخوام نمایش بدم باید با استفاده از یه فرم توی C# نمایش بدم . یعنی با استفاده از ADOMD data adaptor یا CEllSEt و ... شبیه اینها . سوال من اینجاست که اگه بخوام روی فرمم که قراره کوئری هام توش نمایش پیدا کنه یه textbox بذارم و بخوام که مقداری که توی textbox وارد میشه رو بعنوان یه پارامتر توی دستور mdxم تاثیر بدم باید چیکار کنم؟ بطور مثال من یه تاریخ توی textbox روی فرمم وارد کنم و اونوقت کوئری بجای اینکه مثه قبل تمام محصولات فروخته شده رو نشون اون محصولاتی رو که توی اون تاریخ مشخص شده فروخته شده (توی textbox ذکرش شده) رو نشون بده. ممنون

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۰/۲۹ ۱۰:۲۴

با استفاده از ADO.NET نمی‌توان کوئری‌های MDX را مستقیما اجرا کرد. برای اینکار نیاز به Microsoft.AnalysisServices.AdomdClient.dll هست. برای دریافت آن به صفحه‌ی Microsoft® SQL Server® 2012 Feature Pack مراجعه و قسمت Microsoft® SQL Server® 2012 ADOMD.NET را در صفحه جستجو کنید. برای نگارش 2008 به صفحه‌ی Microsoft® SQL Server® 2008 R2 Feature Pack مراجعه و در آن Analysis Management Objects را دریافت کنید. برای SQL Server 2005 نیز در اینجا پروایدر ADOMD قابل دریافت است.

بعد برای استفاده از آن خواهید داشت:

```
using (AdomdConnection conn =
             new AdomdConnection("Data Source=tfsDB;Initial Catalog=Tfs_Analysis; MDX
Compatibility=1;"))
    {
        conn.Open();
        using (AdomdCommand cmd = new AdomdCommand(" ....... mdxQuery here ....... ", conn))
        {
            DataSet ds = new DataSet();
            ds.EnforceConstraints = false;
            ds.Tables.Add();
            DataTable dt = ds.Tables[0];
            dt.Load(cmd.ExecuteReader());
            return dt;
        }
    }
```

و یا پروژه MdxClient - ADO.NET data provider for MDX queries سعی کرده این روش را تبدیل به یک پروایدر جدید ADO.NET کند. نکته‌ی جالب این پروایدر امکان تعریف مقادیر دریافتی از کاربر به صورت پارامتر است در کوئری‌ها تعریف شده. فایل readme آن‌را برای مشاهده یک سری مثال در مورد نحوه‌ی تعریف پارامترها و دریافت داده‌های ارسالی از طریق کاربر، مطالعه کنید.

عنوان: آموزش MDX Query - قسمت هفدهم - توابع Topcount, bottomcount , toppercent, bottompercent, topsum, bottomsum
نویسنده: اردلان شاه قلی
تاریخ: ۱۳۹۲/۱۱/۰۱ ۱۰:۲۰
آدرس: www.dotnettips.info
گروه‌ها: SQL Server 2012, OLAP, MDX, SSAS

در این قسمت بر روی توابع Topcount, bottomcount , toppercent, bottompercent, topsum, bottomsum تمرکز خواهیم داشت.

در ابتدا تصور کنید بخواهیم میزان فروش اینترنتی را برای پنج ردیف از دسته بندی‌های محصولات واکشی کنیم.

```
Select
[Measures].[Internet Sales Amount] on columns,
non empty(
topcount([Product].[Product Categories].[Subcategory],5)
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Bike Racks	$39,360.00
Bike Stands	$39,591.00
Bottles and Cages	$56,798.19
Cleaners	$7,218.60
Fenders	$46,619.58

در تابع بالا پنج ردیف ابتدایی (به صورت فیزیکی) برگردانده می‌شوند.

در اینجا تابع topcount دارای دو پارامتر می باشد که پارامتر دوم آن مشخص کننده‌ی تعداد ردیف واکشی شده و پارامتر اول آن، مشخص کننده‌ی دایمنشنی می‌باشد که عمل واکشی برای آن صورت می‌گیرد. همچنین در بالا از تابع Non empty برای حذف ردیف‌های دارای مقدار Null استفاده شده است. حال تصور کنید بخواهیم پنج دسته بندی محصولی را دریافت کنیم که دارای بیشترین میزان فروش اینترنتی می‌باشند.

```
Select
[Measures].[Internet Sales Amount] on columns,
non empty(
topcount(
[Product].[Product Categories].[Subcategory],
5,
[Measures].[Internet Sales Amount]
)
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Road Bikes	$14,520,584.04
Mountain Bikes	$9,952,759.56
Touring Bikes	$3,844,801.05
Tires and Tubes	$245,529.32
Helmets	$225,335.60

خروجی بر اساس میزان فروش اینترنتی به صورت نزولی مرتب شده است.

تابع Topcount به عنوان پارامتر سوم می‌تواند نام یک Measure را دریافت کند و خروجی را براساس آن شاخص، برگرداند. امکان واکشی و مرتب سازی در تابع Topcount

برای یک شاخص متفاوت از شاخص واکشی شده در یک محور دیگر نیز وجود دارد به مثال زیر دقت کنید:

```
Select
[Measures].[Internet Sales Amount] on columns,
topcount(
[Product].[Product Categories].[Subcategory],
5,
[Measures].[Reseller Sales Amount]
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Road Bikes	$14,520,584.04
Mountain Bikes	$9,952,759.56
Touring Bikes	$3,844,801.05
Mountain Frames	(null)
Road Frames	(null)

همانطور که مشخص می‌باشد، پنج دسته بندی محصولاتی که دارای بیشترین میزان فروش نمایندگان فروش می‌باشند، در خروجی واکشی شده‌اند؛ در حالیکه در محور ستون میزان فروش اینترنتی واکشی شده است.

برای درک بیشتر همین کوئری را دوباره بازنویسی کرده اما اینبار در محور ستون هر دو شاخص [Measures].[Internet Sales Amount],[Measures].[Reseller Sales Amount] را واکشی می‌کنیم.

```
Select
{[Measures].[Internet Sales Amount],[Measures].[Reseller Sales Amount]} on columns,
topcount(
[Product].[Product Categories].[Subcategory],
5,
[Measures].[Reseller Sales Amount]
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount	Reseller Sales Amount
Road Bikes	$14,520,584.04	$29,358,206.96
Mountain Bikes	$9,952,759.56	$26,492,684.38
Touring Bikes	$3,844,801.05	$10,451,490.22
Mountain Frames	(null)	$4,713,672.15
Road Frames	(null)	$3,849,853.34

با بررسی خروجی دو کوئری بالا تفاوت واکشی را متوجه خواهید شد. در هر دو کوئری واکشی براساس شاخص [Measures].[Reseller Sales Amount] انجام شده است

اما واکشی در محور ستون متفاوت می‌باشد. (دقیقا مانند T/SQL که می‌توانستیم، مرتب سازی براساس فیلدی باشد که در قسمت Projection حاضر نبوده و در این حالت در برخی موارد ظاهرا خروجی مرتب نمی‌باشد)

حال تصور کنید بخواهیم 30 دسته بندی محصولاتی را داشته باشیم که دارای کمترین میزان فروش اینترنتی می‌باشند. برای این منظور از تابع bottomcount استفاده می‌کنیم

```
Select
[Measures].[Internet Sales Amount] on columns,
bottomcount(
[Product].[Product Categories].[Subcategory],
30,
[Measures].[Internet Sales Amount]
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Wheels	(null)
Touring Frames	(null)
Saddles	(null)
Road Frames	(null)
Pedals	(null)
Mountain Frames	(null)
Headsets	(null)
Handlebars	(null)
Forks	(null)
Derailleurs	(null)
Cranksets	(null)
Chains	(null)
Brakes	(null)
Bottom Brackets	(null)
Tights	(null)
Bib-Shorts	(null)
Pumps	(null)
Panniers	(null)
Locks	(null)
Lights	(null)
Socks	$5,106.32
Cleaners	$7,218.60
Caps	$19,688.10

ردیف هایی که دارای مقدار Null می باشند هم در خروجی قرار می گیرند

```
Select
[Measures].[Internet Sales Amount] on columns,
non empty bottomcount(
[Product].[Product Categories].[Subcategory],
30,
[Measures].[Internet Sales Amount]
)on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Socks	$5,106.32
Cleaners	$7,218.60
Caps	$19,688.10
Gloves	$35,020.70
Vests	$35,687.00
Bike Racks	$39,360.00
Bike Stands	$39,591.00
Hydration Packs	$40,307.67
Fenders	$46,619.58
Bottles and Cages	$56,798.19

در مثال بالا ردیف‌های دارای مقدار Null را از خروجی حذف کرده ایم.

گاهی نیاز می‌باشد که تعداد دسته بندی‌های محصولاتی را واکشی کنیم که دارای بیشترین یا کمترین میزان فروش اینترنتی می‌باشند و سرجمع فروش اینترنتی آنها بیشتر یا کمتر از X درصد از فروش اینترنتی کل می‌باشد را داشته باشند. به عنوان مثال می‌خواهیم ببینیم کدام دسته بندی محصولات شامل بیشترین میزان فروش اینترنتی می‌باشند و سرجمع فروش آنها 53 در صد از کل فروش اینترنتی می‌باشند.

```
Select
[Measures].[Internet Sales Amount] on columns,
{
 toppercent(
[Product].[Product Categories].[Subcategory],
53,
[Measures].[Internet Sales Amount]
),
 [Product].[Product Categories]
} on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Road Bikes	$14,520,584.04
Mountain Bikes	$9,952,759.56
All Products	$29,358,677.22

و یا واکشی دسته محصولاتی که دارای کمترین میزان فروش اینترنتی می‌باشند و سرجمع فروش اینترنتی آنها کمتر از 1 درصد کل میزان فروش اینترنتی می‌باشد.

```
Select
[Measures].[Internet Sales Amount] on columns,
non empty bottompercent(
[Product].[Product Categories].[Subcategory],
--0.01,
1,
[Measures].[Internet Sales Amount]
) on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Socks	$5,106.32
Cleaners	$7,218.60
Caps	$19,688.10
Gloves	$35,020.70
Vests	$35,687.00
Bike Racks	$39,360.00
Bike Stands	$39,591.00
Hydration Packs	$40,307.67
Fenders	$46,619.58
Bottles and Cages	$56,798.19

کاربرد تابع Topsum در کوئری زیر نمایش داده شده است

```
Select
[Measures].[Internet Sales Amount] on columns,
topsum(
[Product].[Product Categories].[Subcategory],
25000000,
[Measures].[Internet Sales Amount]
) on rows
From [Adventure Works]
```

در این کوئری از تابع TopSum استفاده شده است که عملا حداکثر تعداد دسته بندی محصولاتی را بازیابی می‌کند که دارای بیشترین میزان فروش بوده اند و همچنین در مجموع بیش از 25000000 فروش داشته باشند .

تابع bottomsum عملا تعداد دسته بندی محصولاتی را که دارای کمترین میزان فروش بوده اند و همچنین سرجمع میزان فروش اینترنتی آنها 100000 بوده است را بر می گرداند. البته خروجی توسط non empty ، فیلتر شده است و خروجی هایی که کاملا Null می باشند، حذف گردیده اند.

```
Select
[Measures].[Internet Sales Amount] on columns,
non empty bottomsum(
[Product].[Product Categories].[Subcategory],
100000,
[Measures].[Internet Sales Amount]
)on rows
From [Adventure Works]
```

100 %

Messages | Results

	Internet Sales Amount
Socks	$5,106.32
Cleaners	$7,218.60
Caps	$19,688.10
Gloves	$35,020.70
Vests	$35,687.00