

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 داده ی چند بعدی Adventure Work DW 2008 روی SSAS مطالعه 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 داده ی 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.

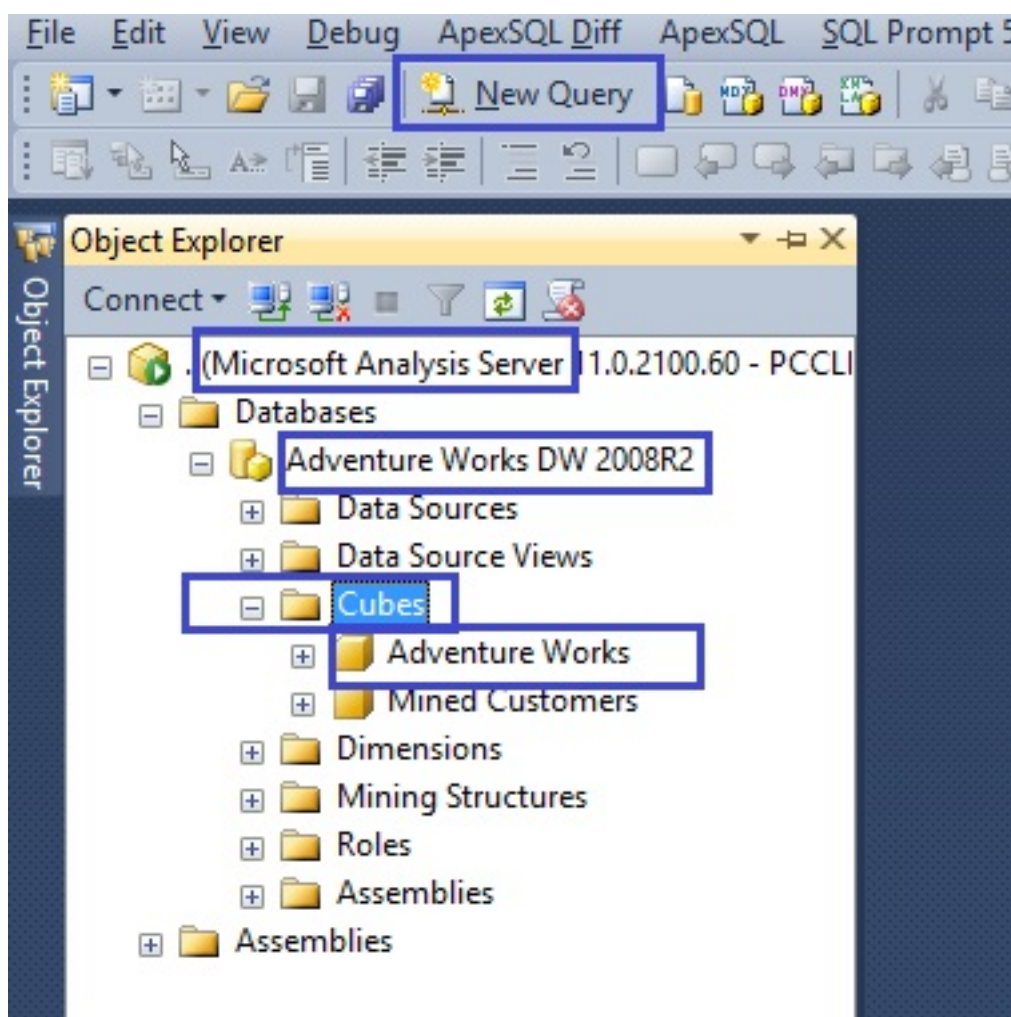
نحوه ی نوشتن یک Select در MDX ها به صورت زیر می‌باشد :

```
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 داده ی Adventure Works DW 2008R2 را انتخاب نمایید و از Cubes Adventure Works را انتخاب نمایید.

حال دکمه ی 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)

متأسفانه هنوز در IDE مربوط به SQL Server کلیدی برای مرتب‌سازی دستورات MDX وجود ندارد و البته در نرم‌افزارهایی مانند SQL To11 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.

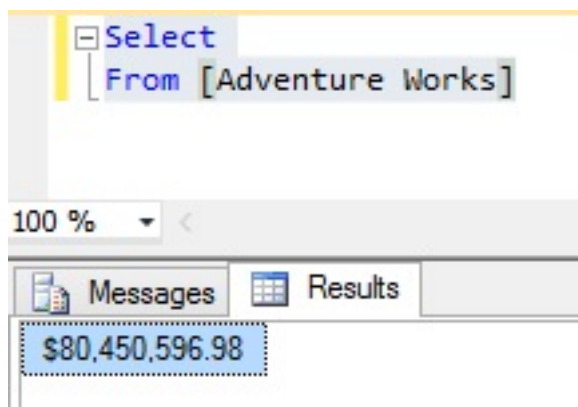
با اجرای دستور زیر اولین کوئری خود را در پایگاه داده‌ی چند بعدی بنویسید (برای اجرا کلید F5 مانند T/SQL کار خواهد کرد).

```
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 پیش فرض انتخاب شده. بنابر این در صورتیکه هیچ گونه Projection یا Where ای اعمال نشده باشد، SQL Server به صورت پیش فرض مقدار Measure پیش فرض را بدون اعمال هیچ بعدی نمایش می‌دهد.

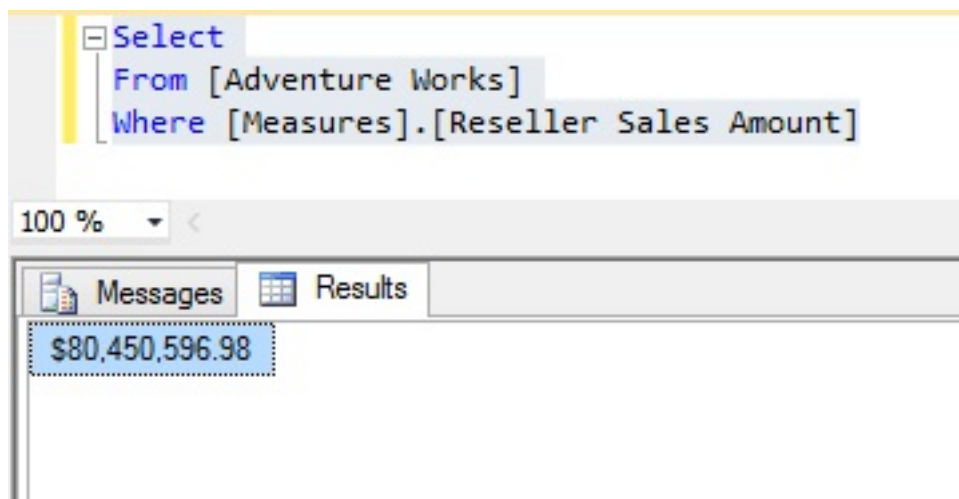
خروجی دستور بالا مشابه تصویر زیر می‌باشد.



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 پرسش‌هایی 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 عملاً 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 :

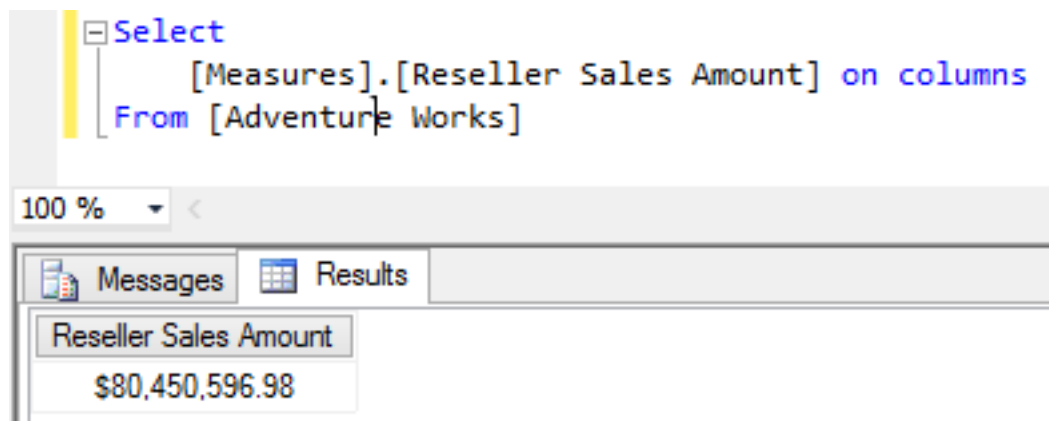
```
Select
From [Adventure Works]
where [Measures].[Internet Sales Amount]
```

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 ، عملاً 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]
```

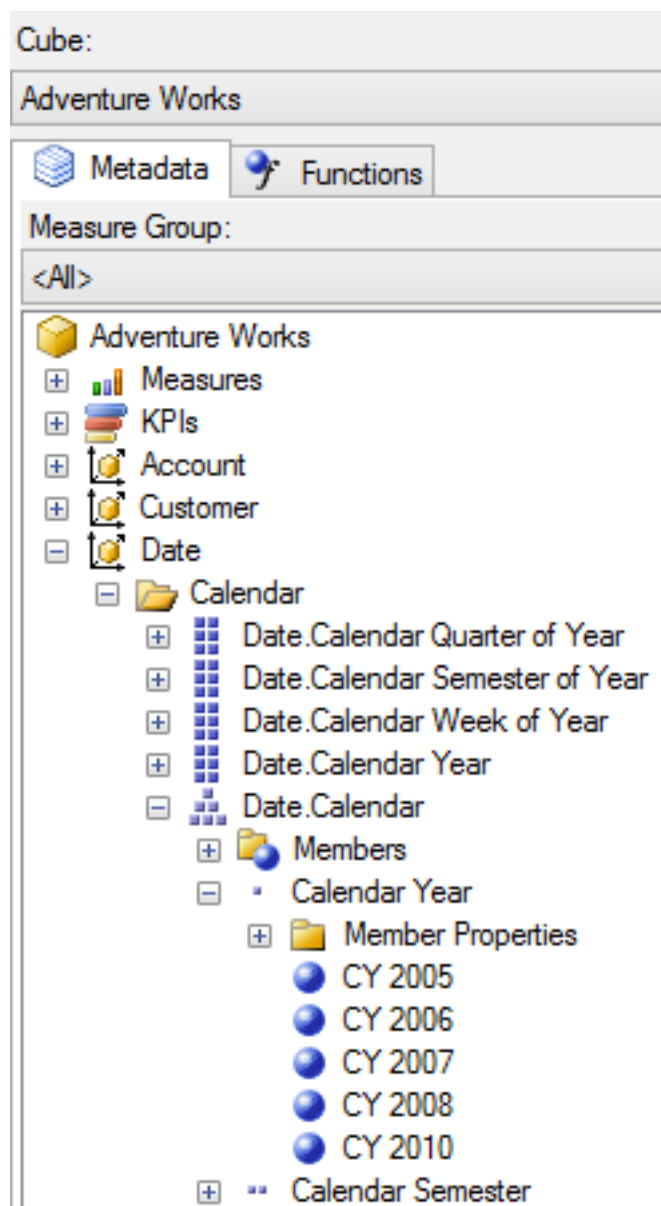
100 % <

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 % <

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 شاخص‌ها یا بعد‌ها، مرتبط به یک سلسله مراتب باشند آنها را در یک { } قرار می‌دهیم ولی اگر سلسله مراتب متفاوت باشد، یا بعد و شاخص باشند باید در () قرار بگیرند.

خوب همان‌طور 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 سال‌ها).

استفاده از دایمنشن و Measure در سطر و ستون مجزا :

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 % <

	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 می باشد

Select

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

100 %

	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 می باشد.

استفاده از تابع non empty :

برای حذف ستون هایی که کاملا دارای مقدار null می باشند به صورت زیر عمل می کنیم :

```
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 می باشد:

	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 سطری 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 اصلاً سلسله مراتبی وجود نداشته باشد، می‌توان 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، حتماً باید در Select مشخص شود که از کدام سلسله مراتب می‌خواهیم استفاده کنیم. در غیر این صورت با خطا مواجه خواهیم شد.

```
Select
[Product] on columns
From [Adventure Works]
```

Messages
Executing the query ... <u>Query (2, 2) The 'Product' dimension contains more than one hierarchy, therefore the hierarchy must be explicitly</u> Execution complete

استفاده از فیلدهای یک دایمنشن که دارای سلسله مراتب می‌باشد نیز جایز می‌باشد

```
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 می باشد.

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]
```

می‌توان از کلمات All X Members 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 می‌باشد.

```
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:

Columns .1

Rows .2

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 .

All Products
Accessories
Bike Racks
Hitch Rack - 4-Bike
Bike Stands
All-Purpose Bike Stand
Bottles and Cages
Mountain Bottle Cage
Road Bottle Cage
Water Bottle - 30 oz.
Cleaners
Bike Wash - Dissolver
Fenders
Fender Set - Mountain
Helmets
Sport-100 Helmet, Black
Sport-100 Helmet, Black
Sport-100 Helmet, Black

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 دقیقاً مشابه با خروجی گزارشات cross tab است (مونی‌زنه!)
نوشتن Queryهای pivoting/cross tabulation با این زبان نسبت به زبان SQL واقعا ساده‌تر است. حقیقتاً لذت بردم.
فقط خواهشی داشتم، اگر مقدور است به مثال‌هایی 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 تلویزیون تماشا می‌کنند (کمتر از 20 دقیقه، بین 20 تا 40 دقیقه..). این بازه‌ها را میشه برای داده‌هایی مثل تاریخ، روزها، هفته‌ها، 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، لطفاً 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 Query روی پایگاه داده‌ی MDB نوشته می‌شوند کاملاً متفاوت از T\SQL می‌باشند که روی مدل رابطه‌ای 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 MDB ها می باشد (Cubeware)

نویسنده: علاقه مند
تاریخ: ۱۱:۲ ۱۳۹۲/۱۰/۱۰

سلام مجدد،
ممنونم از پاسختون.

راستش شناختی نسبت به این دو نوع جدولی که فرمودین ندارم (Fact, Dimension). اشاره کردین که باید رده بندی های مربوطه یعنی رده سنی و بازه زمانی را باید در جداول Dimension پیاده کرد. آیا این پیاده سازی به معنای عدم دینامیک 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: ۱۶:۲۵ ۱۳۹۲/۱۰/۱۰

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 داده ای رابطه ای (Relational) و چند 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 نحوه ی تحلیل و طراحی DW ، امیدوارم به زودی بتوانم مقاله ای را انتشار بدهم.
در خصوص عملگرهای تقسیم و ... و توابع جمعی به زودی در قسمت های بعدی اطلاعات کاملی را خدمتتان ارایه خواهم کرد. به طور خلاصه اینکه OLAP برای ایجاد گزارشات مدیریتی می باشد و قطعا این موارد شما در ادامه پوشش داده خواهد شد. (شما به راحتی می توانید سرجمع یک بازه ی تاریخی یا سنی را برای یک Measure خاص بدست آورید).

نویسنده: میم
تاریخ: ۱۱:۴۱ ۱۳۹۳/۰۲/۲۸

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؟