

## مثال اول از طرح query

در این بخش با استفاده از دیتابیس AdventureWorksDW وارد یک query ساده می شویم.

### توضیحات

یک query ساده را اجرا کرده و طرح Actual Graphical Execution را مورد بررسی قرار خواهیم داد.

اجازه بدهید تولیدات جدول را براساس ProductSubcategoryKey فیلتر کرده و ثبت ها را در جایی که ProductSubcategoryKey برابر 1 است، لیست کنیم.

بنابراین query ما اینطور خواهد بود:

```
SELECT ProductKey, ProductSubcategoryKey
FROM AdventureWorksDW..DimProduct
WHERE ProductSubcategoryKey = 1
```

یک پنجره ی جدید باز کرده و CTRL+M را فشار دهید تا Actual Execution Plan را وارد کرده و سپس query بالا را اجرا کنید. طرح Actual Query برای این query در زیر نمایش داده شده است.

Query 1: Query cost (relative to the batch): 100%

```
SELECT [ProductKey], [ProductSubcategoryKey] FROM [AdventureWorksDW].[DimProduct] WHERE [ProductSubcategoryKey] = 1
```

Execution Plan:

- SELECT (Cost: 0%)
  - Index Seek (NonClustered) [AdventureWorksDW].[DimProduct].[IX\_DimProduct\_ProductKey] (Cost: 100%)

در تصویر بالا مشاهده می کنید که SQL Server به این روش query را پردازش می کند:

ایندکس Seek روی یک ایندکس NonClustered در جدول DimProduct (هزینه 100 درصد). بازگشت به مجموعه ی نتیجه را انتخاب کنید. (هزینه 0 درصد).

برای تجزیه ی طرح query باید به دنبال آیتم های زیر باشیم تا پیشرفت های ممکن را سرعت بخشیم.

منابع فشرده ی عملکردها که در آن درصد بالاست.

اسکن عملکردها.

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

عرض پیکان بین اپراتورها ( هرچه پیکان ضخیم تر داده بیشتر )

در این مثال 100 درصد منابع توسط عملکرد **NonClustered Index Seek** روی جدول **DimProduct** استفاده می شوند.

ما در **query** خود در حال فیلتر کردن روی **ProductSubcategoryKey** در عبارت **WHERE** هستیم. اما چون ایندکسی که استفاده می کنیم، حاوی **Key Product** ( کلید تولید ) و **Key Subcategory Product** (کلید زیرشاخه ی تولید) می باشد، یک ایندکس ساده ی عملکرد **seek** نیز می تواند درخواست را کامل کند.

اجازه بدهید جزئیات عملکرد **Index Seek** را بررسی کنیم.

ماوس خود را روی آیکن **Index Seek** در **Execution Plan** قرار دهید که یک **ToolTip** مانند زیر باید ظاهر شود:



آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>

Index Seek (NonClustered)	
Scan a particular range of rows from a nonclustered index.	
Physical Operation	Index Seek
Logical Operation	Index Seek
Actual Number of Rows	38
Estimated I/O Cost	0.003125
Estimated CPU Cost	0.0001988
Estimated Number of Executions	1
Number of Executions	1
Estimated Operator Cost	0.0033238 (100%)
Estimated Subtree Cost	0.0033238
Estimated Number of Rows	38
Estimated Row Size	15 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	True
Node ID	0
<b>Object</b>	
[AdventureWorksDW].[dbo].[DimProduct]. [IX_DimProduct_ProductSubcategoryKey]	
<b>Output List</b>	
[AdventureWorksDW].[dbo].[DimProduct].ProductKey, [AdventureWorksDW].[dbo]. [DimProduct].ProductSubcategoryKey	
<b>Seek Predicates</b>	
Seek Keys[1]: Prefix: [AdventureWorksDW].[dbo]. [DimProduct].ProductSubcategoryKey = Scalar Operator (CONVERT_IMPLICIT(int,[@1],0))	

در **TollTip** می بینید که فقط یک اجرا (**Number of Executions** تعداد اجراها) لازم بود که 38 (تعداد برآورد شده ی ردیف ها) ردیف درست را گزارش می دهد.

اجازه بدهید پیکان بین عملکردها را چک کنیم.

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>

ماوس خود را روی آیکن **SELECT** در **Execution Plan** قرار دهید که یک **ToolTip** مانند زیر باید ظاهر شود.

SELECT	
Cached plan size	16 B
Degree of Parallelism	1
Estimated Operator Cost	0 (0%)
Estimated Subtree Cost	0.0033238
Estimated Number of Rows	38
<b>Statement</b>	
SELECT [ProductKey], [ProductSubcategoryKey] FROM [AdventureWorksDW].[DimProduct] WHERE [ProductSubcategoryKey]=@1	

مجددا مشاهده می کنیم که 38 ردیف گزارش می شود.

موردی که باید ذکر شود، **Estimated Subtree Cost** (برآورد هزینه ی زیردرخت) می باشد. از آنجایی که بیشترین قسمت عملکرد رها می شود، این هزینه ی کلی برای طرح **query** می باشد. این هزینه به **SQL Server** نیز کمک می کند تا تعیین کند چه زمان از یک طرح اجرایی همگام استفاده کند.

چگونه می توانیم اجرا را بهبود ببخشیم؟

برای این **query** نمی توان کاری انجام داد. هزینه ی کلی برآورد شده ی زیرمجموعه (**Subtree**) خیلی پایین میباشد، همچنین ما در حال انجام یک **Index Seek** برای گرفتن داده می باشیم، که خیلی بهتر از آن نمی شود.