

## مثال دوم query plan

اجازه دهید با استفاده از دیتابیس AdventureWorksDW وارد یک query پیچیده تر بشویم.

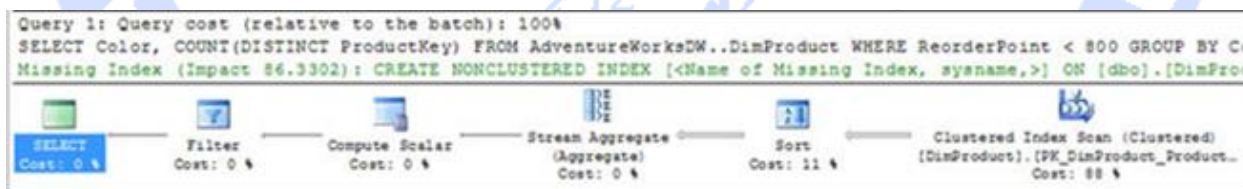
توضیحات

اجازه بدهید جدول تولیدات را بر اساس  $ReorderPoint < 800$  فیلتر کرده و با رنگ گروه بندی کنیم و تنها رکوردهایی را نشان دهیم که بیشتر از 10 عدد دارند. بنابراین query ما به شکل زیر خواهد بود:

```
SELECT Color, COUNT(DISTINCT ProductKey)
FROM AdventureWorksDW..DimProduct
WHERE ReorderPoint < 800
GROUP BY Color
HAVING COUNT(DISTINCT ProductKey) > 10
ORDER BY Color
```

یک پنجره ی query جدید باز کرده و CTRL+M را فشار دهید تا Actual Execution Plan ( طرح اجرای حقیقی) را وارد کنید و سپس query بالا را اجرا کنید.

Actual Execution Plan برای این query در زیر نمایش داده شده است:



در تصویر بالا مشاهده می کنید که SQL Server ، query را به یک روش خطی پردازش کرده و مقدار درصد هزینه را زیر هر آیکن اپراتور ذکر کرده است.

بنابراین مراحل برای این طرح عبارتند از:

Clustered Index Scan روی جدول DimProduct (هزینه 88 درصد) ( داده را در حالیکه  $ReorderPoint < 800$  به دست می آورد).

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

88146323 - 88446780 - 88146330

با استفاده از عملکرد مرتب سازی (sort)، مرتب سازی براساس رنگ انجام می شود (هزینه 11 درصد)  
عملکرد جریان مصالح (Stream Aggregate operation) (هزینه 0 درصد) (گروه بندی شده براساس رنگ یا شماره)

عملکرد محاسبه ی اسکالر (Compute Scalar operation) (شمارش)

فیلتر کردن نتایج (هزینه 0) (دارای شمار < 10)

انتخاب بازگشت مجموعه نتایج (هزینه 0) (گزارش داده)

برای وارد کردن بهبودهای ممکن برای این query، باید آیتم های زیر را جستجو کنیم:

عملکردهای منابع فشرده (resource intensive operations) که در آنها درصد بالا می باشد.

اسکن عملکردها و

نوارهای ضخیم بین اپراتورها (هرچه نوار ضخیم تر باشد، تعداد داده ها بیشتر می باشد).

واضح است که فشرده ترین منبع عملکرد Clustered Index Scan روی جدول DimProduct می باشد که از 88 درصد همه ی منابع استفاده کرده و تعداد زیادی از ردیف ها را ه عملکرد بعدی منتقل می کند.

Clustered Index Scan (Clustered)	
Scanning a clustered index, entirely or only a range.	
Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Actual Number of Rows	606
Estimated I/O Cost	0.156458
Estimated CPU Cost	0.0008236
Number of Executions	1
Estimated Number of Executions	1
Estimated Operator Cost	0.157282 (88%)
Estimated Subtree Cost	0.157282
Estimated Number of Rows	606
Estimated Row Size	24 B
Actual Rebinds	0
Actual Rewinds	0
Ordered	False
Node ID	4
<b>Predicate</b>	
[AdventureWorksDW].[dbo].[DimProduct].[ReorderPoint] < (800)	
<b>Object</b>	
[AdventureWorksDW].[dbo].[DimProduct].[PK_DimProduct_ProductKey]	
<b>Output List</b>	
[AdventureWorksDW].[dbo].[DimProduct].ProductKey, [AdventureWorksDW].[dbo].[DimProduct].Color	

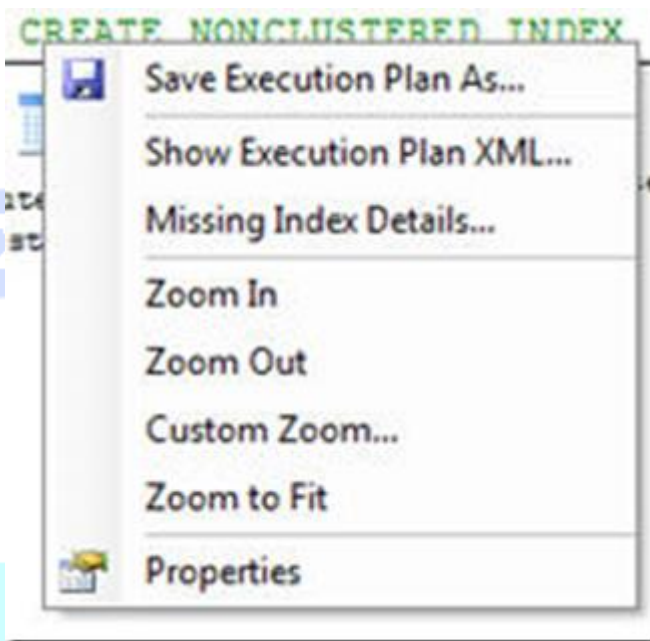
چگونه می توانیم اجرا را بهبود ببخشیم؟

این کار بسیار ساده می باشد، زیرا **SSMS** نشان یک ایندکس گم شده را به رنگ سبز به ما نشان می دهد. اگر روی این بخش سبز رنگ راست کلیک کنید، یک منو نمایش داده خواهد شد، **Missing Index Details...** را انتخاب کنید.

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/>



یک پنجره ی **query** جدید با کد زیر باز خواهد شد. این کد پیشنهاد می دهد که یک ایندکس جدید روی ستون **the** **ReorderPoint** وارد کرده و ستون **Color** را در ایندکس وارد کنیم. براساس ایندکس جدید **query** باید یک عملکرد **Index Seek** انجام دهد و می توانیم براساس برآورد **Query Optimizer**، سرعت **query** را تا **86** درصد بالا ببریم.

آموزشگاه تحلیگر داده ها

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/>