

Index Scans and Table Scans

روش های زیادی برای بهبود اجرا با قرار دادن سخت افزار وجود دارد، اما موثریت راه زمانی است که **query** های خود را هماهنگ سازی می کنید. یک مشکل متداول نبود ایندکس ها یا وجود ایندکس های نادرست می باشد، بنابراین **SQL Server** باید داده های بیشتری را پردازش کند تا رکوردهایی را که با معیارهای **query** ها مناسب هستند، پیدا کند. این موارد **Index Scans** و **Table Scans** نامیده می شوند.

در این بخش به چگونگی یافتن این موارد و چگونگی حل آنها خواهیم پرداخت.

توضیح

یک **index scan** یا اسکن جدول زمانیست که **SQL Server** باید داده یا صفحات ایندکس را برای یافتن رکوردهای مناسب اسکن کند. اسکن کردن مخالف جستجو (**seek**) می باشد، جستجو (**seek**) از ایندکس برای اشاره به رکوردهایی استفاده می کند که برای کامل کردن **query** لازم هستند. دلیل اینکه تمایل دارید اسکن های خود را یافته و اصلاح کنید به خاطر این است که نیاز به **I/O** بیشتر و همچنین زمان پردازش بیشتری دارند. این موردی است که در برنامه ای که در طول زمان رشد می کند، با آن مواجه خواهید شد. وقتی اولین باشد، اجرا بسیار خوب است، اما در طول زمان وقتی که داده ی بیشتری اضافه می شود، اسکن های ایندکس بیشتر طول خواهند کشید.

برای یافتن این موارد می توانید با اجرای **Profiler** و یا تنظیم سرور ردیابی (**server side trace**) آغاز کنید و به دنبال وضعیت هایی باشد که دارای مقادیر بالای خواندن هستند. وقتی که این موقعیت ها را شناسایی کردید، می توانید به برنامه ی **query** دقت کرده و اگر اسکنی در حال اجراست، مشاهده کنید.

در اینجا یک **query** ساده را مشاهده می کنید که اجرا می کنیم. ابتدا از **Ctrl+M** برای روشن کردن برنامه ی اجرایی استفاده کنید و سپس **query** را اجرا کنید.

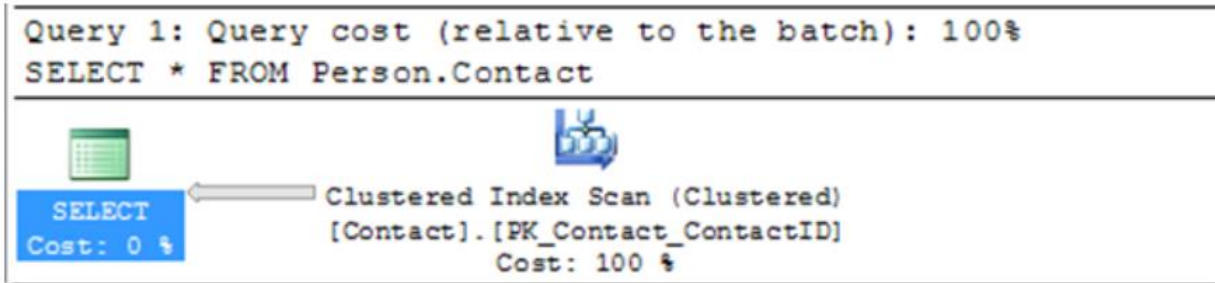
```
SELECT * FROM Person.Contact
```

در اینجا مشاهده می کنیم که این **query** در حال اجرای **Clustered Index Scan** می باشد. از آنجایی که این جدول دارای یک **clustered index** می باشد و عبارت **WHERE** وجود ندارد، **SQL Server** کل **clustered index** را برای بازگرداندن تمام ردیف ها اسکن می کند. بنابراین در این مثال هیچ موردی برای بهبود این **query** وجود ندارد.

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

88146323 - 88446780 - 88146330

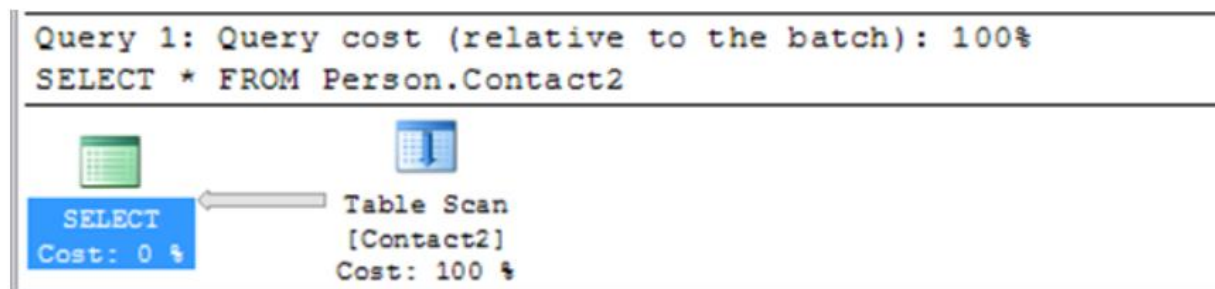
<http://www.tahlildadeh.com/>



در مثال بعدی یک کپی از جدول **Person.Contact** بدون **clustered index** ایجاد کرده و سپس **query** را اجرا می کنیم.

SELECT * FROM Person.Contact2

در اینجا مشاهده می کنیم که این **query** در حال انجام **Table Scan** می باشد. بنابراین وقتی یک جدول دارای **Clustered Index** باشد، یک **Clustered Index Scan** انجام خواهد داد و وقتی که یک جدول **clustered index** نداشته باشد، **Table Scan** انجام خواهد داد. از آنجایی که این جدول **Clustered Index** ندارد، و عبارت **WHERE** وجود ندارد، **SQL Server** کل جدول را برای بازگرداندن همه ی ردیف ها اسکن می کند. بنابراین در این مثال نیز موردی برای بهبود **query** وجود ندارد.



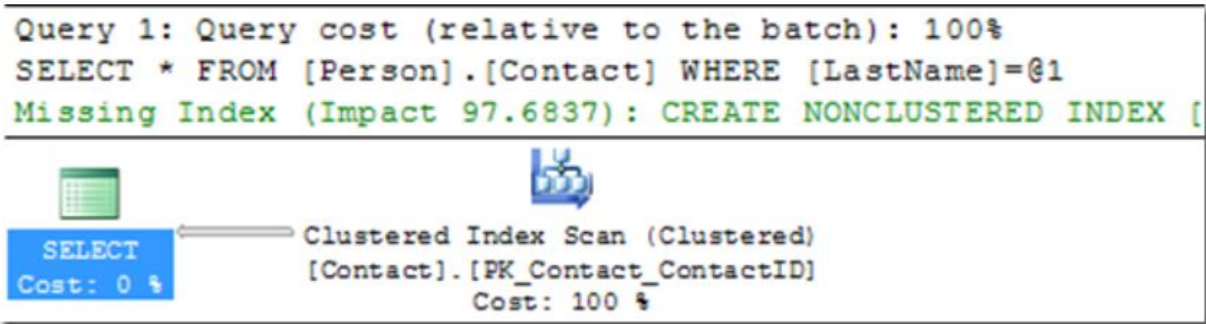
در مثال بعدی ما یک عبارت **WHERE** وارد **query** می کنیم.

SELECT * FROM Person.Contact WHERE LastName = 'Russell'

در اینجا مشاهده می کنیم که هنوز **Clustered Index Scan** را دریافت می کنیم، اما این بار **SQL Server** به ما اجازه می دهد تا متوجه یک ایندکس مفقود شویم. اگر روی برنامه ی **query** راست کلیک کرده و **Missing Index Details** را انتخاب کنید، پنجره ی جدیدی با یک اسکرینیت برای ایجاد یک **missing index** (ایندکس مفقود) دریافت خواهید کرد.

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

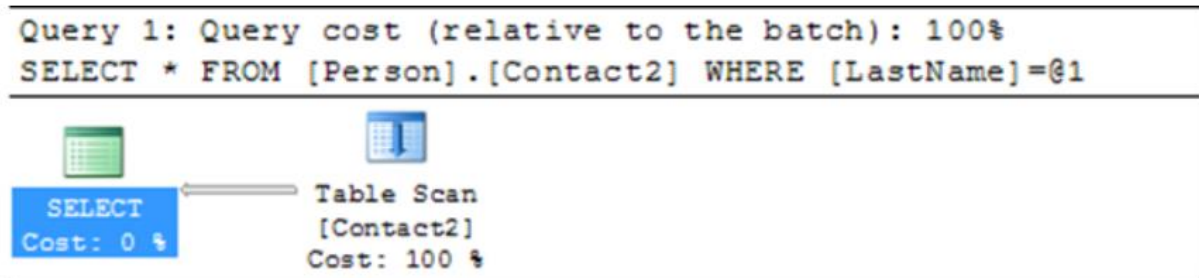
88146323 - 88446780 - 88146330



اجازه بدهید همین کار را برای جدول **Person.Contact2** خود انجام دهیم.

```
SELECT * FROM Person.Contact2 WHERE LastName = 'Russell'
```

مشاهده می کنیم که هنوز **Table Scan** را داریم، اما **SQL Sever** پیشنهادی در مورد چگونگی اصلاح آن ندارد.

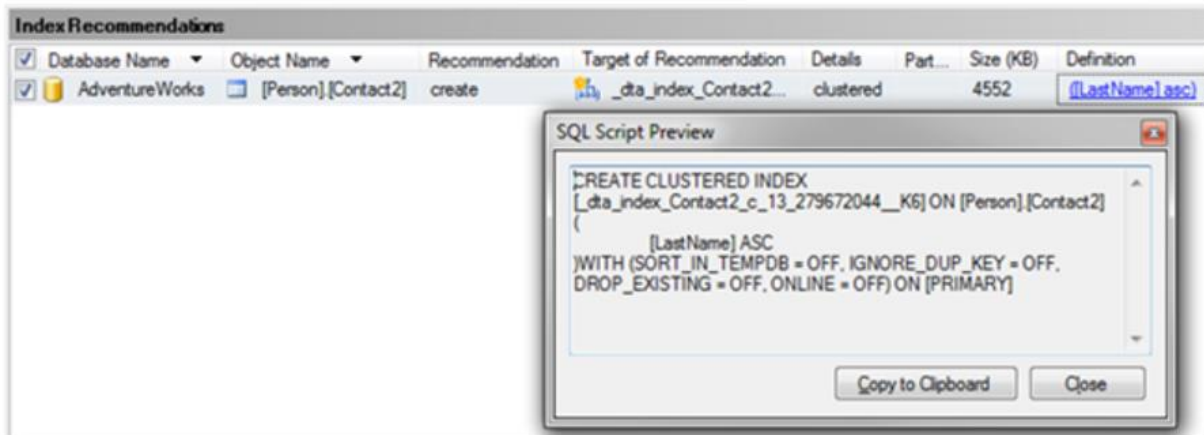


کار دیگری که می توان انجام داد، استفاده از **Database Engine Tuning Advisor** برای مشاهده ی پیشنهادات می باشد. اگر

query را در **SSMS** انتخاب کنم، راست کلیک کرده و **Analyze Query in Database Engine Tuning Advisor** را

انتخاب کنید، ابزار شروع به کار می کند و من می توانم گزینه ها را انتخاب کرده و تجزیه را آغاز کنم.

در زیر پیشنهادی است که این ابزار ارائه می دهد و ما می توانیم ببینیم که پیشنهادات یک ایندکس جدید ایجاد می کند، بنابراین مشاهده می کنید که استفاده از هر دو ابزار می تواند موثر باشد.



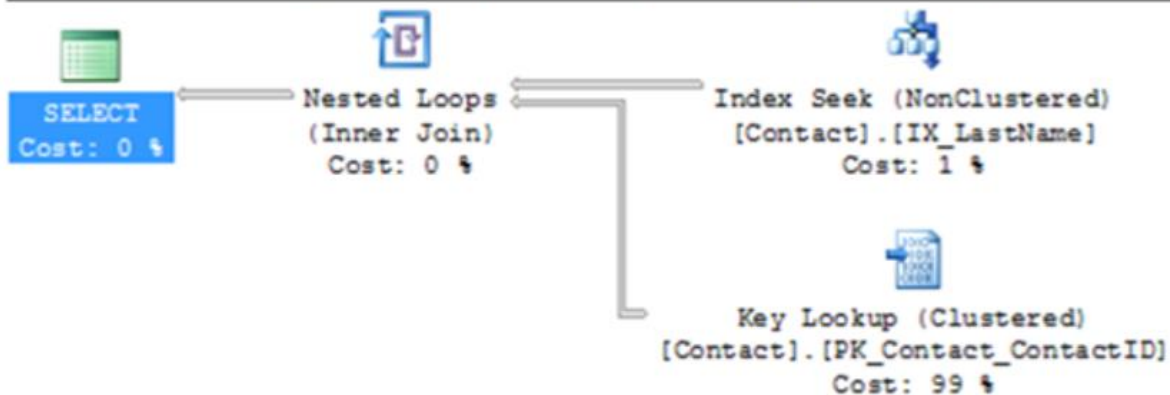
Create New Index

اجازه بدهید ایندکس پیشنهاد شده را روی **Person.Contact** ایجاد کرده و **query** را مجدداً اجرا کنیم.

```
USE [AdventureWorks]
GO
CREATE NONCLUSTERED INDEX [IX_LastName]
ON [Person].[Contact] ([LastName])
GO
SELECT * FROM Person.Contact WHERE LastName = 'Russell'
```

مشاهده می کنید که برنامه ی **query** تغییر یافته و به جای یک **Clustered Index Scan** اکنون دارای یک **Index Seek** هستیم که خیلی بهتر است. همچنین مشاهده می کنید که یک عملکرد **Key Lookup** وجود دارد که راجع به آن در بخش بعد صحبت خواهیم کرد.

```
Query 1: Query cost (relative to the batch): 100%
SELECT * FROM [Person].[Contact] WHERE [LastName]=@1
```



خلاصه:

با شناسایی و تعمیر **Index Scans** و **Table Scans** می توانید اجرا را به شدت به ویژه برای جدول های بزرگتر بهبود ببخشید. بنابراین زمانی را به شناسایی مکانی که ممکن است اسکن ها اتفاق بیفتند، اختصاص دهید و ایندکس های لازم را برای برطرف کردن مشکل ایجاد کنید. موردی که باید مد نظر داشته باشید این است که ایندکس زیاد نیز مشکل آفرین است، بنابراین مطمئن شوید که تعادلی در تعداد ایندکس های ایجاد شده برای یک جدول خاص وجود دارد.

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/>