

دوره آموزش Performance Tuning در SQL Server (گروه 15)

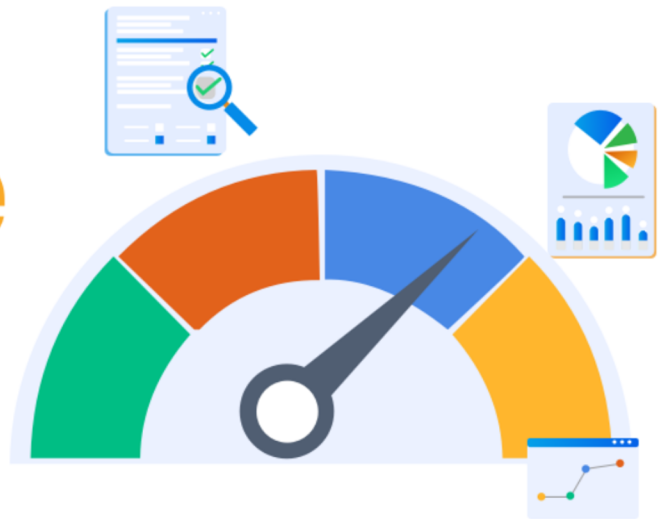
طول دوره: ۴۰ ساعت

مدرس: مسعود طاهری

Performance Tuning



Microsoft®
SQL Server®



عنوان دوره: آموزش Performance Tuning در SQL Server

مخاطبین: برنامه‌نویسان و مدیران بانک اطلاعاتی

پیش‌نیاز: دانش کافی در زمینه کوئری نویسی

طول دوره: ۴۰ ساعت به همراه حل تمرین

نحوه ارائه: به دو صورت حضوری و غیرحضوری

مدرس: مسعود طاهری

تاریخ شروع دوره: ۲۷ تیر ماه ۱۴۰۲

پشتیبانی: دارای گروه تلگرامی اختصاصی

قیمت ثبت‌نام به صورت حضوری: ۸,۷۰۰,۰۰۰ تومان

قیمت ثبت‌نام به صورت غیرحضوری: ۵,۷۰۰,۰۰۰ تومان

دسترسی از طریق پلیر اختصاصی اسپات پلیر

روش مشاهده دوره‌های آموزشی محافظت شده



ثبت‌نام به صورت حضوری:

این دوره آموزشی در تاریخ سه شنبه ۲۷ تیرماه ۱۴۰۲ به صورت حضوری در **کانون اسلامی انصار** به آدرس: «تهران، خیابان شریعتی، تقاطع خیابان طالقانی، پلاک ۱۷۳» برگزار می‌شود.

لوکیشن محل برگزاری



مدرس: مسعود طاهری

فیلم هر جلسه طی ۳ روز کاری بعدی برگزاری تدوین و در پلی اختصاصی اسپات پلیر قابل مشاهده می باشد. برای دسترسی به فیلم جلسات، به «**حساب کاربری، سفارش ها**» بروید. روی سفارش مربوطه «نمایش» را بزنید. راهنمای نصب اسپات پلیر را مطالعه کنید. (این آموزش بر روی دستگاه ویندوز یا مک، اندروید یا مرورگر وب قابل استفاده است) **مشاهده راهنما**

ثبت نام به صورت غیر حضوری:

این دوره آموزشی در تاریخ **۲۷ تیرماه ۱۴۰۲** شروع می گردد. فیلم هر جلسه **طی ۳ روز کاری بعد از برگزاری** تدوین و در پلی اختصاصی اسپات پلیر قابل مشاهده می باشد.

برای دسترسی به فیلم جلسات، به «**حساب کاربری، سفارش ها**» بروید. روی سفارش مربوطه «نمایش» را بزنید. راهنمای نصب اسپات پلیر را مطالعه کنید. (این آموزش بر روی دستگاه ویندوز یا مک، اندروید یا مرورگر وب قابل استفاده است) **مشاهده راهنما**



ثبت نام در دوره

مدرس این دوره کیست؟

مسعود طاهری مدرس و مشاور ارشد SQL Server & BI

- CTO شرکت نیک آموز
- مدرس دوره های SQL Server و هوش تجاری در شرکت نیک آموز
- رئیس واحد توسعه پایگاه داده پرداخت الکترونیک سداد
- نویسنده کتاب PolyBase در SQL Server



۱. آشنایی اولیه با معماری SQL Server:

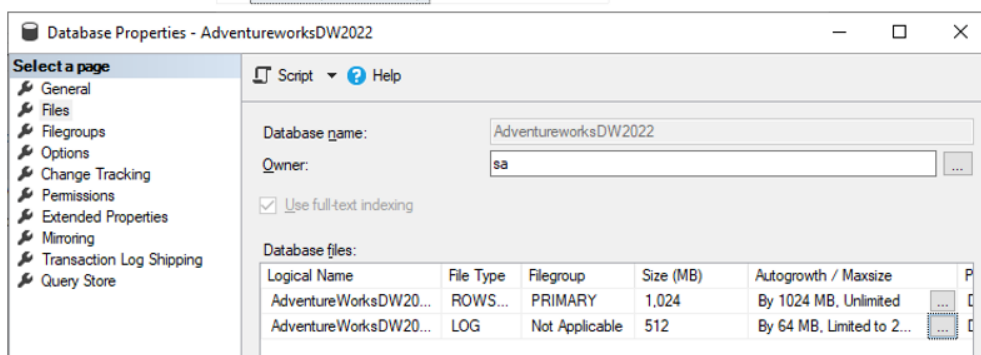
- آشنایی با مفهوم DMV و DMF در SQL Server
- بررسی مفهوم Database
- بررسی مفهوم Data File و Log File
- بررسی معماری Log File
- بررسی معماری منطقی Log File
- بررسی معماری فیزیکی Log File
- بررسی نحوه اعمال تنظیمات مناسب برای Data File و Log File
- بررسی تنظیمات مفید سیستم عامل برای فایل‌های بانک اطلاعاتی
- بررسی تنظیمات Database Instant File Initialization
- آشنایی با Buffer Pool در SQL Server
- بررسی تغییرات مربوط به Buffer Pool در SQL Server ۲۰۲۲
- بررسی مفهوم Checkpoint
- چه کارهایی در هنگام Checkpoint رخ می‌دهد؟
- بررسی اولیه مفهوم Lock
- بررسی اولیه مفهوم Latch
- آشنایی با DMV ها و DMF های مهم و پرکاربرد در این حوزه
- معرفی ابزارهایی کاربردی برای مشاهده محتوای Page ها

Log File و اعمال تنظیمات مناسب برای آن:

همه ما می‌دانیم که بانک‌های اطلاعاتی در SQL Server حداقل از یک Data File و یک Log File تشکیل شده است. چنانچه ما معماری ذخیره‌سازی داده در هر کدام از این فایل‌ها را بدانیم رفتارمان نسبت به این فایل‌ها عوض می‌شود. یکی از مواردی که من همیشه در بانک‌های اطلاعاتی بزرگ تنظیمات مناسبی بر روی آن اعمال می‌کنم، تنظیمات مربوط به Log File است.

```
SELECT
    servicename, instant_file_initialization_enabled
FROM sys.dm_server_services
WHERE
    filename LIKE '%sqlservr.exe%'
GO
```

	servicename	instant_file_initialization_enabled
1	SQL Server (SQLSERVER2022)	Y



نوع دسترسی در این فایل به صورت Sequential (دسترسی ترتیبی) می‌باشد و برای افزایش کارایی بانک اطلاعاتی باید تنظیمات مناسبی بر روی آن اعمال نمود. در حال حاضر در SQL Server 2022 با تنظیم اصولی و مناسب برای Initial Size و همچنین نحوه رشد می‌توانیم به عملکرد قابل قبولی در این زمینه رسید.

ما در این دوره با نحوه اعمال تنظیمات مناسب برای Log File و Data File آشنا خواهیم شد.

SQLQuery1.sql - (lo...TAHERI\TaHERi (67))*

```

1  USE ClinicDW
2  GO
3  SELECT * FROM SYS.database_files
4  GO
5

```

124 %

	file_id	file_guid	type	type_desc	data_space_id	name	physical_name	state	state_desc
1	1	C2BD46FD-4A...	0	ROWS	1	ClinicDW	D:\Program File...	0	ONLINE
2	2	72F69658-BC4...	1	LOG	0	ClinicDW_log	D:\Program File...	0	ONLINE

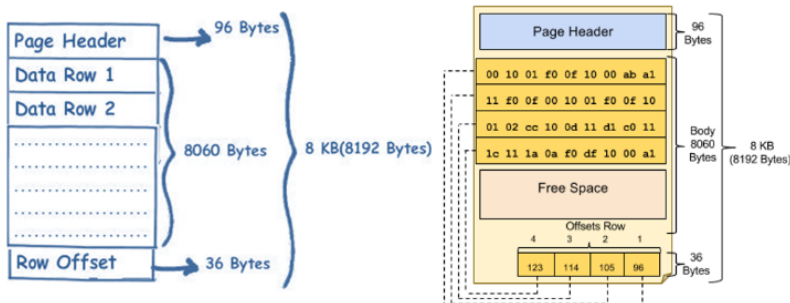
۲. آشنایی با Page ها و انواع آن در SQL Server:

- بررسی مفهوم Page در Data File
- بررسی معماری ذخیره‌سازی داده‌ها در Page
- معرفی اجمالی انواع Page ها و کاربرد هر کدام از آنها
- بررسی مفهوم Allocation Page
- بررسی مفهوم Row-Overflow
- بررسی تکنیک‌های استفاده مناسب از Data Type ها
- بررسی مفهوم Extent
- معرفی انواع Extent ها در SQL Server

بررسی مفهوم Page

۱. ذخیره داده در قالب Page ها

۲. انجام عملیات IO به ازای Page ها



چرا باید جزئیات نحوه ذخیره سازی داده‌ها در جداول را تا به این حد یاد بگیریم؟

این سوالی است که خیلی از دوستانی که در دوره SQL Server در Performance Tuning شرکت می‌کنند از من می‌پرسند. شما هر چقدر با جزئیات و ریزه کاری‌های مربوط به SQL Server آشنا شوید رفتارشان با آن عوض می‌شود و به خوبی می‌توانید از عهده مشکلات مربوط به کارایی بانک‌های اطلاعاتی برآید.

۳. بررسی File Group ها

- بررسی مفهوم File Group
- بررسی کاربردهای File Group
- بررسی مزایای استفاده از File Group
- بررسی انواع File Group ها
- بررسی نحوه تعریف اصولی File Group ها
- بررسی نحوه رشد Data File های مربوط به File Group
- بررسی نحوه قرار دادن ایندکس در یک File Group
- بررسی نحوه ذخیره و بازیابی BLOB ها در File Group های جداگانه
- بررسی فایل گروه‌های Filestream
- بررسی نحوه استفاده از فایل گروه‌های Filestream در جهت افزایش کارایی بانک اطلاعاتی
- بررسی نحوه ایجاد و مدیریت فایل گروه‌های Filestream
- بررسی نحوه پی‌کربندی SQL Server برای کار با Filestream
- بررسی نحوه کار با Filestream در .NET.
- بررسی جداول FileTable
- بررسی نحوه استفاده از FileTable در .NET.

نحوه ارتباط دات نت با Filestream و File Table:

ما در این دوره شما را با نحوه استفاده از تکنولوژی Filestream و همچنین File Table در دات نت آشنا می‌کنیم. از این سبک کد نویسی خودم به شخص در اغلب پروژه‌ها استفاده کرده‌ام و کارایی بانک اطلاعاتی را افزایش داده‌ام. یادم ادم است در یکی از پروژه‌هایی که به تازگی درگیر آن بودیم یک سیستم با کاربری ملی وجود داشت که برنامه‌نویسان آن به اشتباه ذخیره فایل‌ها را خودشان در NTFS و همچنین داده‌های مربوط به آن را در MongoDB انجام داده بودن که به محض افزایش تعداد فایل‌ها و تعداد کاربران دچار مشکلات

زیادی شدن که با انجام تغییراتی در معماری برنامه و هدایت فایل‌ها و داده‌های مربوط به آن به سمت SQL Server و استفاده از تکنولوژی Filestream سیستم به راحتی توانست از عهده درخواست‌های کاربران بر بیاید.



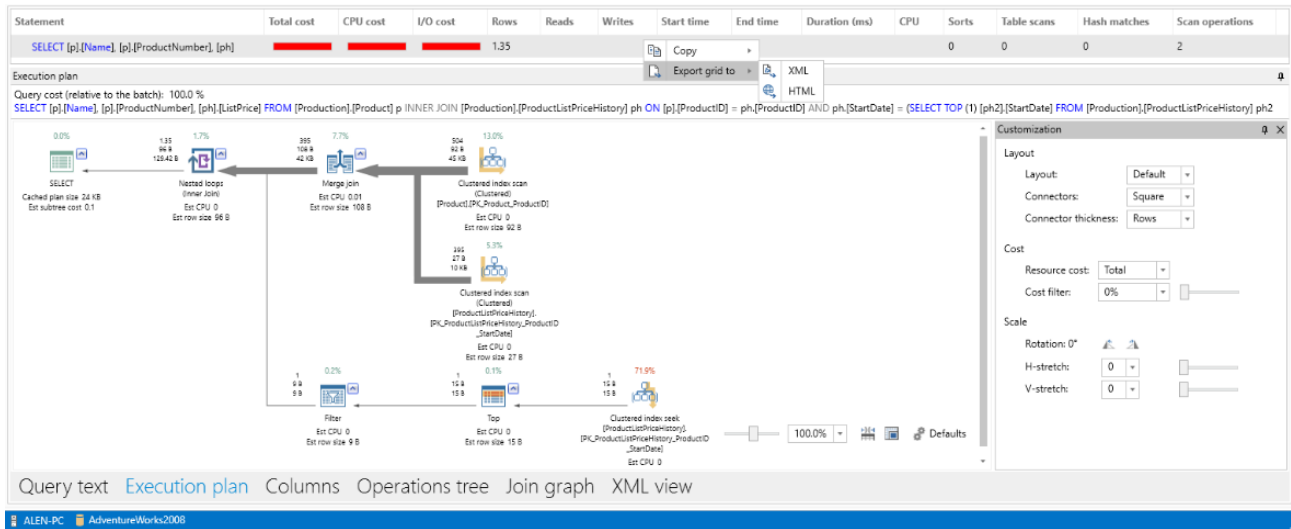
۴. معرفی Execution Plan و نحوه کار با آن:

- بررسی مفهوم Logical Read و Physical Read
- بررسی دستور Set Statistics IO
- بررسی دستور Set Statistics Time
- بررسی ترتیب اجرای منطقی و فیزیکی دستورات SQL Server
- بررسی مفهوم Execution Plan و انواع آن
- بررسی نحوه Compare کردن Execution Plan ها در SQL Server
- بررسی نحوه ذخیره و بازیابی Execution Plan

معرفی یک ابزار حرفه‌ای برای آنالیز Execution Plan:

ما در SSMS می‌توانیم Execution Plan ها را مشاهده و آنالیز کنیم، اما SSMS امکانات محدودی برای آنالیز Plan دارد.

ما در این دوره شما را با یک ابزار خیلی قوی برای آنالیز Execution Plan آشنا خواهیم کرد.

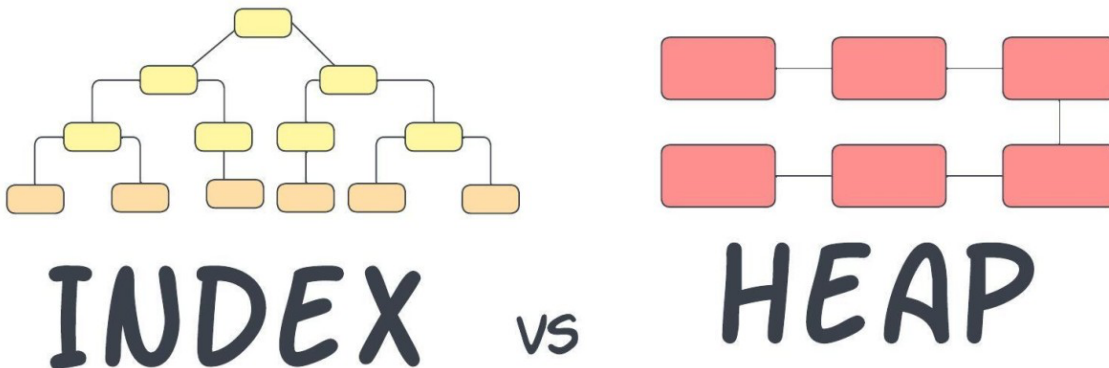


۵. بررسی ایندکس و معماری آن

- بررسی Heap Table
- بررسی معماری جداول Heap در SQL Server
- بررسی چند سناریوی کاربردی برای استفاده از Heap Table
- آشنایی با Clustered Index
- بررسی معماری Clustered Index
- بررسی تاثیر انتخاب درست کلید Clustered Index در کارایی بانک اطلاعاتی
- چگونه Where Condition جهت استفاده از Clustered Index بنویسیم؟
- مقایسه Heap و Clustered Index
- بررسی انتخاب کلید ایندکس به صورت ترکیبی و تاثیر آن بر کارایی بانک اطلاعاتی
- بررسی ویژگی‌های مهم و کاربردی برای انتخاب کلید ایندکس
- بررسی Non Clustered Index و معماری مربوط به آن
- بررسی مفهوم lookup هنگام استفاده از ایندکس‌های Non Clustered
- بررسی Key Lookup
- بررسی RID Lookup

یکی از پروژه‌هایی که چند وقت است در نیک آموز درگیر انجام عملیات Performance & Tuning آن بودیم شامل یک جدول Heap با بیش از چهار میلیون رکورد بود. زمانی که کاربران در ساعت اداری با این سیستم کار می‌کردند مصرف CPU سرور بالای 80 درصد بود این موضوع باعث شده بود که

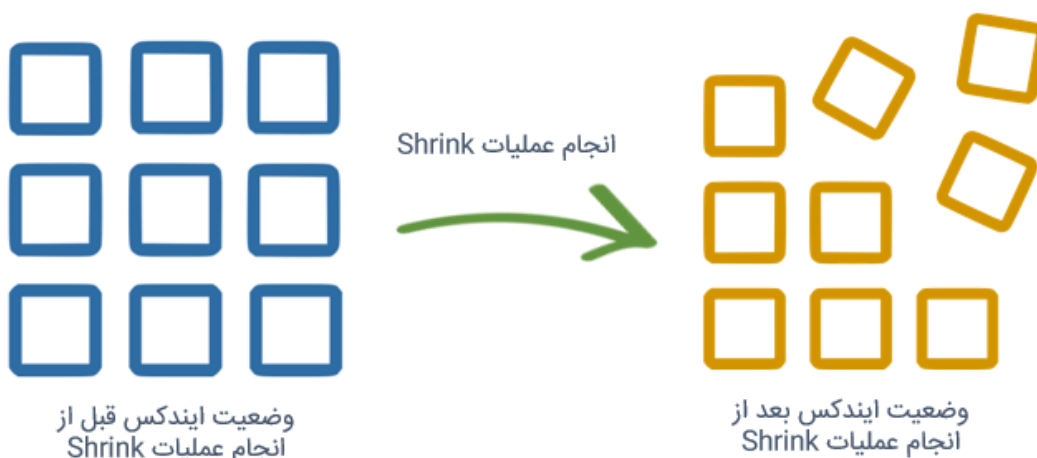
کاربران به شدت شاکی باشند و کُندی در سیستم محسوس باشد. اولین کاری که برای رفع مشکل انجام دادیم تبدیل جدول به Clustered و ایجاد تعدادی Non Clustered ایندکس بر روی آن بود.



۶. ایندکس و نکات پیشرفته مربوط به آن

- بررسی مفهوم Cover Index در SQL Server
- بررسی نحوه حذف هزینه Lookup با استفاده از Cover Index
- بررسی Filtered Index ها در SQL Server
- بررسی استفاده از GUID به عنوان کلید ایندکس و مشکلات مربوط به آن
- Computed Column و اعمال ایندکس بر روی آنها
- بررسی تاثیر عملیات Shrink بر روی ایندکس ها
- بررسی تاثیر منفی ایجاد بیش از حد ایندکس ها در کارایی بانک اطلاعاتی
- بررسی Option هایی که هنگام ایجاد ایندکس باید در نظر داشت

یکی از وحشتناک ترین کارهایی که یک DBA ممکن است خیلی زیاد آن را انجام دهد Shrink کردن Database است. این استراتژی را یکی از DBA های یک کارخانه مشهور دولتی انجام داده و به این موضوع به شدت افتخار می کرد که با انجام این کار Database دوباره جان می گیرد



به شکل دوستانه و با چند خط اسکریپت وضعیت بانک اطلاعاتی قبل از Shrink و بعد از آن را به او نشان دادم تا متوجه مشکلات مربوط به این کار شود. پس از حذف پروسه Shrink از Jobهای شبانه ایشان مشکلات مربوط به کندی برخی از کوئری‌ها رفع شد.

۷. فشرده‌سازی جداول و ایندکس

- بررسی مفهوم Data Compression در SQL Server
- بررسی فشرده سازی به روش Row Compression
- بررسی فشرده سازی به روش Page Compression
- بررسی تاثیر فشرده‌سازی بر عملیات Select
- بررسی تاثیر فشرده‌سازی بر عملیات Insert, Update, Delete
- بررسی تاثیر فشرده‌سازی بر کارایی بانک اطلاعاتی
- بررسی تاثیر فشرده‌سازی بر روی Clustered Index ها
- بررسی تاثیر فشرده‌سازی بر روی Non Clustered Index ها
- بررسی توابع مربوط به فشرده‌سازی در SQL Server
- بررسی تاثیر عملیات فشرده‌سازی بر روی کارایی عملیات تهیه و بازیابی نسخه پشتیبان

تاثیر فشرده‌سازی جدول بر کارایی بانک اطلاعاتی:

در یکی از پروژه‌هایی که در آن درگیر بودم یکی از جداول بانک اطلاعاتی دارای حجم زیادی از داده بود که با توجه به اینکه حجم Select های این جدول خیلی زیاد بود تصمیم گرفتیم Data Compression را بر روی این جدول راه‌اندازی کنیم. با انجام این کار حجم داده‌های موجود در ایندکس‌های این جدول کاهش پیدا کرد و این موضوع باعث شد که IO کوئری‌ها کمتر شود.

تصویر زیر نمونه‌ای از فضای یک جدول قبل از فشرده و پس از فشرده‌سازی به روش‌های مختلف را نمایش می‌دهد.

```
2
3 SP_SPACEUSED None_Compression
4 GO
5 SP_SPACEUSED Row_Level_Compression
6 GO
7 SP_SPACEUSED Page_Level_Compression
8 GO
```

Results						
	name	rows	reserved	data	index_size	unused
1	None_Compression	1000	2696 KB	2672 KB	16 KB	8 KB

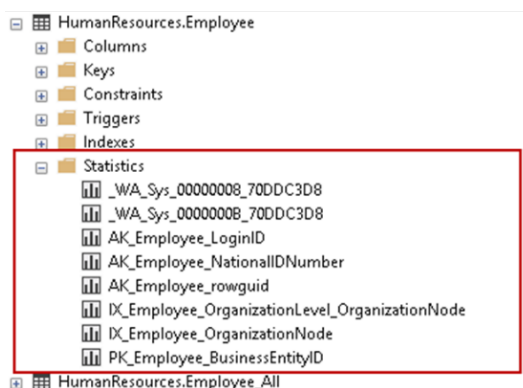
	name	rows	reserved	data	index_size	unused
1	Row_Level_Compression	1000	1672 KB	1600 KB	16 KB	56 KB

	name	rows	reserved	data	index_size	unused
1	Page_Level_Compression	1000	72 KB	16 KB	16 KB	40 KB

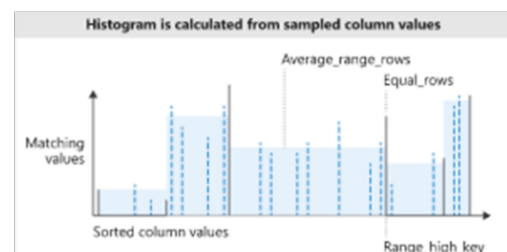
۸. آشنایی و کاربرد آن در SQL Server

- بررسی مفهوم Statistics
- بررسی کاربرد Statistics در SQL Server
- بررسی مفهوم Query Selectivity در SQL Server
- بررسی تاثیر Statistics در هنگام انتخاب یک ایندکس مناسب
- بررسی اجزاء مربوط به Statistics (Stat Header , Density Vector , Histogram)
- بررسی روش‌های ایجاد Statistics در SQL Server
- بررسی شرایط به روز رسانی Statistics در SQL Server
- بررسی روش‌های به روز رسانی Statistics در SQL Server
- بررسی تغییرات Statistics در نسخه ۲۰۱۴ (بازنویسی روال‌های مربوط به آن توسط مایکروسافت)
- بررسی تاثیر منفی تغییرات Statistics نسخه ۲۰۱۴ بر روی بانک‌های اطلاعاتی
- بررسی مشکلات به روز نبودن Statistics
- بررسی مشکل Spill to Disk
- بررسی Filtered Statistics ها
- بررسی تغییرات جدید Statistics در SQL Server ۲۰۲۲

یکی از سازمان‌های دولتی که با آن کار می‌کنم، چند روز پیش با من تماس گرفته بود و می‌گفت که پروسیجر مربوط به درج رکوردهای فایل X در بانک اطلاعاتی به شدت عمل می‌کند. هر عملیات درج بیش از 30 ثانیه زمان می‌برد. این پرویجر یکی از وحشتناک‌ترین SP‌های مربوط به سازمان مورد نظر بود که شامل بیش از هزار خط کد (SP و Function‌های تو در تو) بود. بررسی و Tune کردن این نوع SP‌ها خیلی خیلی زمان‌بر بوده و عموماً برای بهینه کردن هرچه بهتر نیاز است که شما به Business مربوط به آن هم کمی تسلط داشته باشید تا بتوانید در برخی از مواقع بازنویسی SP را با دستورات بهتر انجام دهید. خب اما من وقت کافی برای این کار نداشتم. بچه‌های سازمان به من می‌گفتند کل کاری که قبلاً به ازای این SP انجام می‌شد، 2 ثانیه زمان می‌برد. اما حالا بیش از 30 ثانیه، با کمی



انجام عملیات
Update Statistics



پرسش از بچه‌ها متوجه شدم به‌روزرسانی Statistics ها در سازمان جز دستور کار نبوده و با بروز کردن Statistics های سازمان به صورت Manual مشکل حل شد.

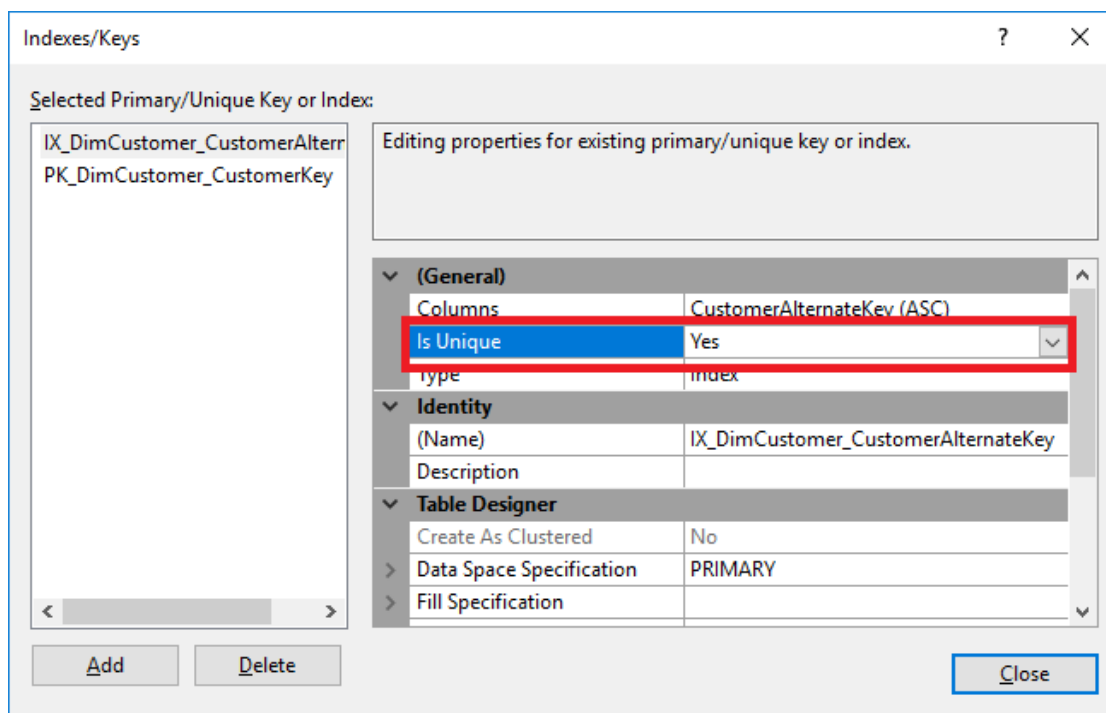
زمان اجرای Stored Procedure به کمتر از 2 ثانیه کاهش یافت. آیا شما با مباحث Statistics آشنا هستید؟ آیا می‌دانید تغییرات مربوط به حوزه Statistics در SQL Server 2014 چه چیزهایی بوده؟

۹. Constraint ها و تاثیر آن بر کارایی بانک اطلاعاتی

- بررسی مفهوم Primary Key و Unique Key
- نحوه انتخاب صحیح Primary Key در جداول
- Primary Key & Unique Key و ایندکس‌های مرتبط به آنها
- بررسی مفهوم Check Constraint و تاثیر آن بر کارایی بانک اطلاعاتی
- بررسی مفهوم Foreign Key و تاثیر آن بر کارایی بانک اطلاعاتی

خیلی از دوستانی که فیلد یا فیلدهایی را به عنوان Unique Key ایجاد می‌کنند اطلاعی ندارند که Unique Key توسط یک Unique Non Clustered Index کنترل می‌شود.

چنانچه ما با این مفهوم و همچنین معماری داخلی این ایندکس آشنا باشیم می‌توانیم به خوبی از این ویژگی در پروژه‌های عملیاتی خود استفاده کنیم.



۱۰. Fragment شدن ایندکس‌ها

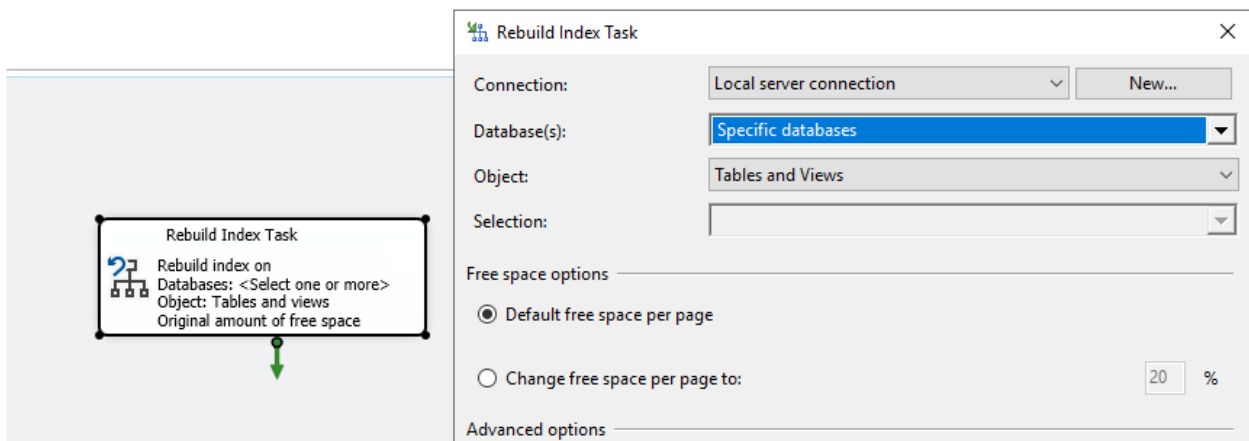
- بررسی مفهوم Fragmentation
- بررسی انواع Fragmentation در SQL Server
- بررسی تاثیر استفاده از FillFactor و Pad_Index بر روی Fragmentation
- بررسی روش‌های کاهش Fragmentation ایندکس‌ها در SQL Server
- بررسی نحوه تنظیم اصولی FillFactor برای ایندکس‌ها
- بررسی نحوه مانیتور کردن Fragmentation
- بررسی Rebuild کردن ایندکس‌ها و کاهش Fragmentation
- بررسی روش‌های مختلف Rebuild کردن ایندکس‌ها
- بررسی حالت‌های مختلف Rebuild کردن ایندکس‌ها به صورت آنلاین و حالت‌های مختلف آن

SP_HELPINDEX DimCustomer

	index_name	index_description	index_keys
1	IX_DimCustomer_CustomerAlternateKey	nonclustered, unique located on PRIMARY	CustomerAlternateKey
2	PK_DimCustomer_CustomerKey	clustered, unique, primary key located on PRIMARY	CustomerKey

شاید به بانک‌های اطلاعاتی برخورد کرده باشید که چندین سال است کار می‌کند و به یک باره کند می‌شود. بیشتر مشکلات کندی مربوط به این نوع برنامه‌ها با یک Rebuild ساده ایندکس ممکن است حل شود. از این نوع بانک‌های اطلاعاتی زیاد دیده‌ام؛ کارخانه X، کارخانه Y و... همه این‌ها نفرات که به طور تخصصی با SQL Server کار کنند را ندارند. یک Admin شبکه مسئول همه چیز است. حق هم دارد به این مسائل مسلط نباشد، چون حیطه تخصصی آن شبکه و مسائل مربوط به آن است ولی SQL Server را به او تحمیل کرده‌اند.

اما این وسط برنده کسی است که با این مباحث آشنایی داشته باشد و بتواند درآمد و سطح دانش خود را افزایش دهد.



۱۱. مانیتور کردن ایندکس‌ها

- معرفی DMVهای مربوط به مانیتور کردن ایندکس‌ها
- بررسی ویو مدیریتی Index Usage Stats
- بررسی ویو مدیریتی Index Operational Stats
- بررسی ویو مدیریتی Index Physical Stats
- پیدا کردن ایندکس‌ها تکراری و تصمیم‌گیری در خصوص وجود یا عدم وجود آن
- پیدا کردن ایندکس‌های بلا استفاده در SQL Server
- بررسی و آنالیز Missing Indexها در SQL Server

پیدا کردن ایندکس‌های بلا استفاده:

یکی از کارهایی که من همیشه بر روی بانک‌های اطلاعاتی انجام می‌دهم پیدا کردن ایندکس‌های بلا استفاده بر روی سرور است. وجود ایندکس‌های اضافی بر روی سرور باعث بالا رفتن افزایش Fragmentation ایندکس، افزایش مدت زمان Blocking، افزایش احتمال وجود Deadlock و.... می‌شود. ما در SQL Server می‌توانیم ایندکس‌های بلا استفاده را با استفاده از بررسی تعدادی DMV مدیریتی بدست آوریم.

Find unused
Indexes
Sql Server

۱۲. Triggerها و تاثیر آن بر کارایی بانک اطلاعاتی

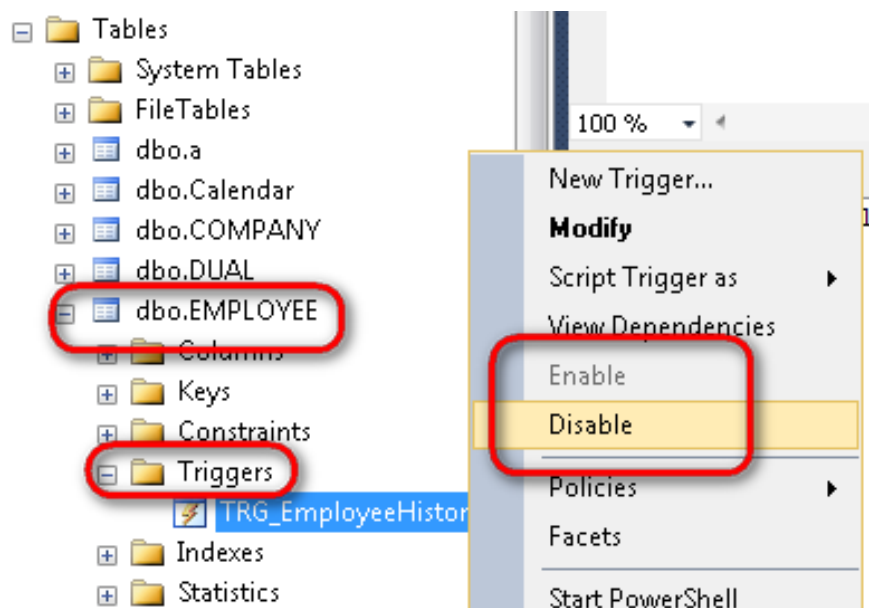
- بررسی مفهوم Trigger
- بررسی انواع DML Trigger
- بررسی کنترل Business Ruleها با استفاده از Trigger و تاثیر آن بر کارایی بانک اطلاعاتی

استفاده از تریگر و کاهش کارایی بانک اطلاعاتی:

چند سال پیش با مجموعه‌ای که با SQL Server 2008 کار می‌کرد همکاری داشتم. در یکی از جداول بزرگ این مجموعه در جهت ثبت سوابق تغییرات رکوردهای این جدول از Trigger After استفاده شده بود. پس از آنالیز و بررسی تریگر موجود در جدول متوجه دو موضوع مهم شدیم.

- ارتباط After Trigger با Fragment شدن جدول
- افزایش مدت درج و به روز رسانی رکوردهای موجود در جدول

برای رفع این مشکلات راهکاری خوب ارائه دادیم.



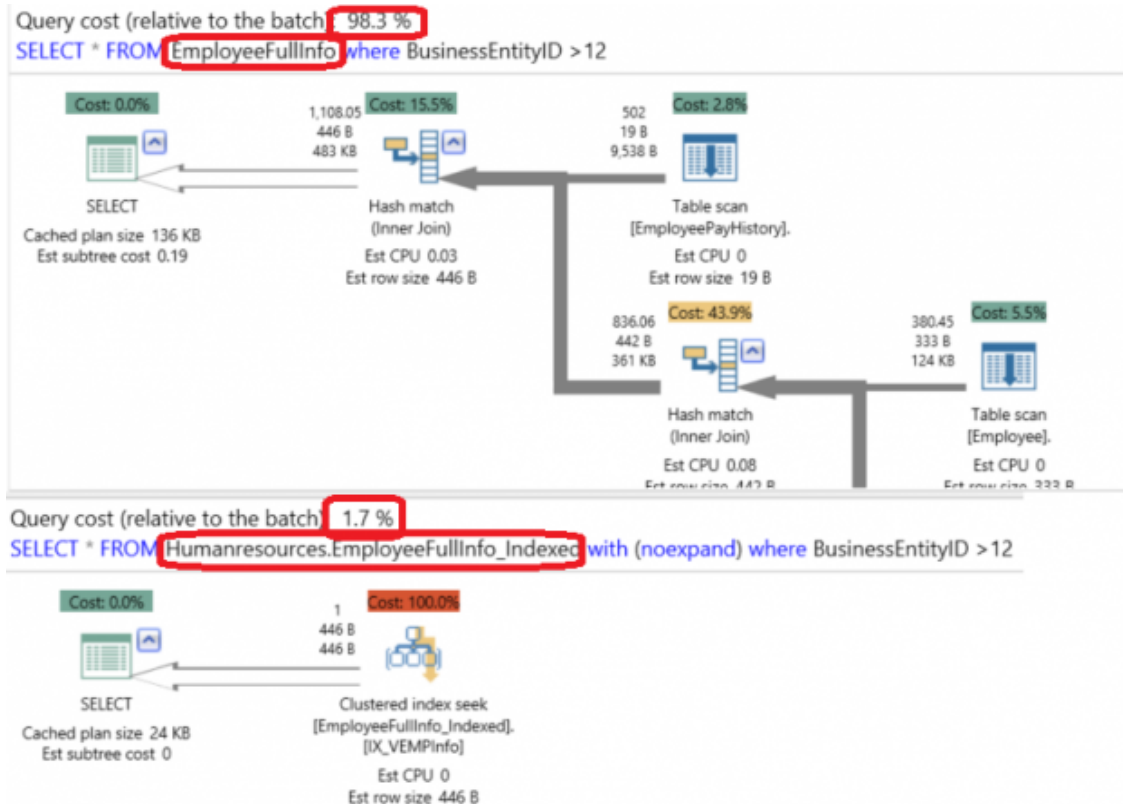
۱۳. View ها و تاثیر آن بر کارایی بانک اطلاعاتی

- بررسی مفهوم View
- بررسی نحوه ایجاد Partition View و تاثیر آن بر کارایی بانک اطلاعاتی
- بررسی نحوه ایجاد ایندکس بر روی View ها
- بررسی نحوه اعمال Clustered Index بر روی View ها
- بررسی نحوه اعمال Non Clustered Index بر روی View ها

استفاده به جا از Index View:

در یکی از پروژه‌هایی که برای افزایش کارایی آن درگیر بودم با تعداد زیادی از جداول اطلاعات پایه درگیر بودم که هزینه Join آنها بالا بود با بررسی نوع جداول و فیلدهای مورد استفاده در کوئری‌ها

تصمیم به ایجاد View Index در بانک اطلاعاتی مربوط به پروژه گرفتیم. استفاده از این ویژگی باعث شد که هزینه مربوط به Join کوئری‌ها حذف گردد.



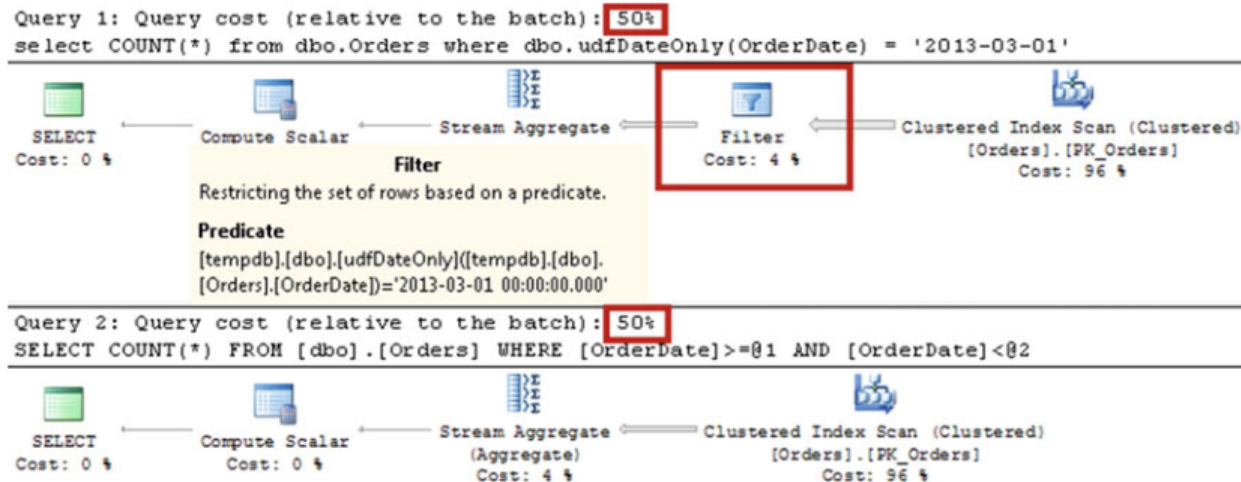
۱۴. Function ها و تاثیر آن بر کارایی بانک اطلاعاتی

- بررسی مفهوم Function
- بررسی انواع Function ها
- بررسی Scalar Function و تاثیر آن بر کارایی بانک اطلاعاتی
- بررسی تغییرات جدید مربوط به Scaler Function ها در حوزه کارایی در SQL Server ۲۰۱۹، ۲۰۲۲
- بررسی In Line Function و تاثیر آن بر کارایی بانک اطلاعاتی
- بررسی Multi Statement Function و تاثیر آن بر کارایی بانک اطلاعاتی
- بررسی مشکلات مربوط به Cardinality Estimation هنگام کار با Function ها

تاریخ شمسی و چالش‌های مربوط به آن:

یکی از روش‌های ذخیره تاریخ در بانک‌های اطلاعاتی ذخیره آن به صورت میلادی است اما مشکل زمانی است بخواهیم این تاریخ را به صورت شمسی نمایش دهیم. برای انجام این کار در SQL

Server ما می‌توانیم Scaler Function استفاده کنیم. این روش توسط اکثر شرکت‌های نرم‌افزاری استفاده می‌شود، اما زمانی که حجم داده‌های موجود در جداول شما زیاد باشد دیگر این روش جوابگوی کار نخواهد بود و باید از روش‌های بهینه و مناسبی استفاده کنید.



۱۵. پارتیشن‌بندی جداول

- پارتیشن‌بندی چیست؟
- چه زمانی باید سراغ پارتیشن‌بندی جداول برویم.
- چگونه یک جدول را پارتیشن‌بندی کنیم؟
- چگونه ایندکس‌های خود را پارتیشن‌بندی کنیم؟
- معرفی انواع تکنیک‌های پارتیشن‌بندی
- آشنایی با تابع Partition\$
- بررسی نحوه استفاده از File Group ها هنگام کار پارتیشن‌بندی جداول
- پارتیشن‌بندی و فشرده‌سازی داده‌ها
- انجام عملیات Merge و Split به ازای پارتیشن‌ها در SQL Server
- استفاده از ویژگی Incremental Statistics هنگام پارتیشن‌بندی
- معرفی Best Practices برای پارتیشن‌بندی جداول

پارتیشن‌بندی یک جدول با حجم 900GB:

در یکی از پروژه‌های بزرگی که درگیر بودم جدولی در حدود 900 گیگابایت حجم داشت. برای افزایش سرعت عملکرد این جدول تصمیم به پارتیشن‌بندی جدول بر اساس ماه گرفتیم.

مهمترین مزیتی که این تکنیک به ما داد این بود که توانستیم Active Partition ها یا پارتیشن‌هایی که مراجعه به آنها زیاد می‌باشد را در دیسک‌های پرسرعت قرار دهیم. همچنین ما ویژگی Incremental Statistics را در این جدول فعال کردیم که تاثیر بسیار مناسبی در عملکرد کوئری‌هایمان داشت.

رفتن به سمت Data Partitioning اصولی دارد که ما در این دوره این اصولی دارد که در این دوره این اصول را به خوبی یاد می‌گیریم.

Execution plan.xml partition.sql - jos...-PC\JosephSack (51)* x Properties

Query 1: Query cost (relative to the batch): 100%
SELECT [ProductID] FROM [Production].[ProductPro...]

Clustered Index Seek (Clustered)
[ProductProductPhoto].[Clust...]
Cost: 100 %

Clustered Index Seek (Clustered)

Misc

Property	Value
Actual Execution Mode	Row
Actual Number of Batches	0
Actual Number of Rows	85
Actual Partition Count	1
Actual Partitions Accessed	50
Actual Rebinds	0
Actual Rewinds	0

۱۶. جداول موقت و تنظیم TempDB و تاثیر آن بر کارایی بانک اطلاعاتی

- Temp Table یا جداول موقت چیست؟
- تاثیر استفاده از Temp Table بر کارایی بانک اطلاعاتی
- Table Variable چیست؟
- تاثیر استفاده از Table Variable بر کارایی بانک اطلاعاتی
- TVP چیست ؟
- تاثیر استفاده از TVP بر کارایی بانک اطلاعاتی
- بررسی بانک اطلاعاتی TempDB
- بررسی نحوه پیکربندی صحیح بانک اطلاعاتی TempDB
- بررسی مفهوم Latch Contention و نحوه کاهش آن در بانک اطلاعاتی TempDB
- بررسی Trace Flag های وابسته به Tempdb

تنظیم صحیح بانک اطلاعاتی TempDB:

بانک اطلاعاتی TempDB یکی از مهمترین بانک‌های اطلاعاتی سیستمی موجود در SQL Server است که تنظیم صحیح آن به طور مستقیم در کارایی بانک اطلاعاتی تاثیر دارد. یکی از راه‌های بهینه‌سازی TempDB افزایش تعداد Data File های آن به نسبت CPU Core های سرور است. اما این افزایش تعداد Data File ها همراه با آیتم‌های دیگر باید انجام شود. خود من این موضوع را در اکثر پروژه‌های Tuning ی که درگیر می‌شوم می‌انجام می‌دهم.

```
3 USE tempdb
4 GO
5 SP_HELPFILE
```

50 %

Results Messages

	name	fileid	filename	filegroup	size	maxsize	growth	usage
1	tempdev	1	C:\Program Files\Microsoft...	PRIMARY	51200 KB	Unlimited	10240 KB	data only
2	templog	2	C:\Program Files\Microsoft...	NULL	8192 KB	Unlimited	65536 KB	log only
3	tempdev2	3	C:\Program Files\Microsoft...	PRIMARY	51200 KB	Unlimited	10240 KB	data only
4	tempdev3	4	C:\Program Files\Microsoft...	PRIMARY	51200 KB	Unlimited	10240 KB	data only
5	tempdev4	5	C:\Program Files\Microsoft...	PRIMARY	51200 KB	Unlimited	10240 KB	data only
6	tempdev5	6	C:\Program Files\Microsoft...	PRIMARY	51200 KB	Unlimited	10240 KB	data only
7	tempdev6	7	C:\Program Files\Microsoft...	PRIMARY	51200 KB	Unlimited	10240 KB	data only
8	tempdev7	8	C:\Program Files\Microsoft...	PRIMARY	51200 KB	Unlimited	10240 KB	data only
9	tempdev8	9	C:\Program Files\Microsoft...	PRIMARY	51200 KB	Unlimited	10240 KB	data only

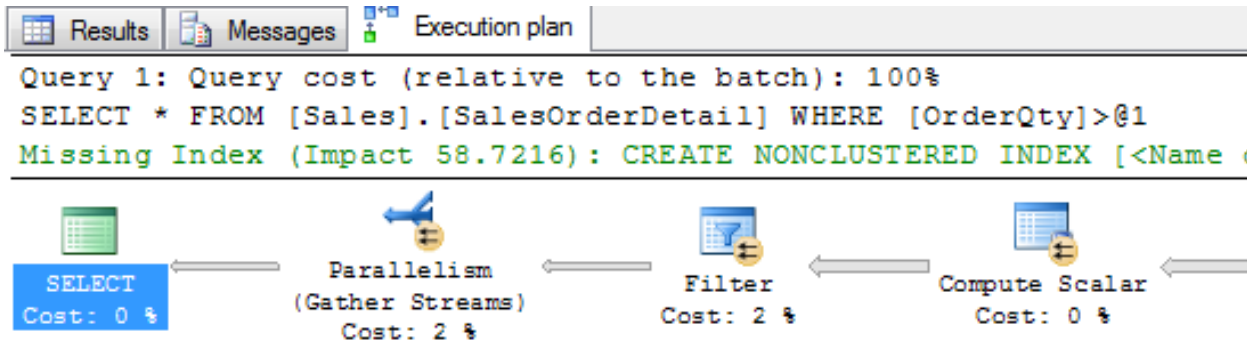
۱۷. Query Execution Plan (نقشه اجرایی کوئری‌ها)

- بررسی پیشرفته اپراتورها در Execution Plan
- بررسی اپراتورهای فیزیکی مربوط به (Join (Nested Loop, Merge Join , Hash Join)
- بررسی اپراتورهای فیزیکی (Aggregate (Stream Aggregate , Hash Aggregate)
- بررسی Table Spool
- بررسی Loop, Merge, Hash join Hint
- بررسی NoExpand Expand Hint

هدایت کوئری به سمت اجرا به صورت Parallel:

در کورس قبلی Performance & Tuning یکی از دوستان شرکت کننده در دوره کوئری داشت که پلن اجرایی آن به صورت Serial بوده و زمان اجرای آن بیش از 3 دقیقه بود ما با بازنویسی کوئری

مربوطه توانستیم روند اجرای کوئری را به حالت Parallel در بیاوریم. طی این حالت ردیف‌های بازگشتی به ازای هر کدام از اپراتورها توانایی اجرای بر روی CPU Core های مختلف دارند.



۱۸. بررسی Plan Caching

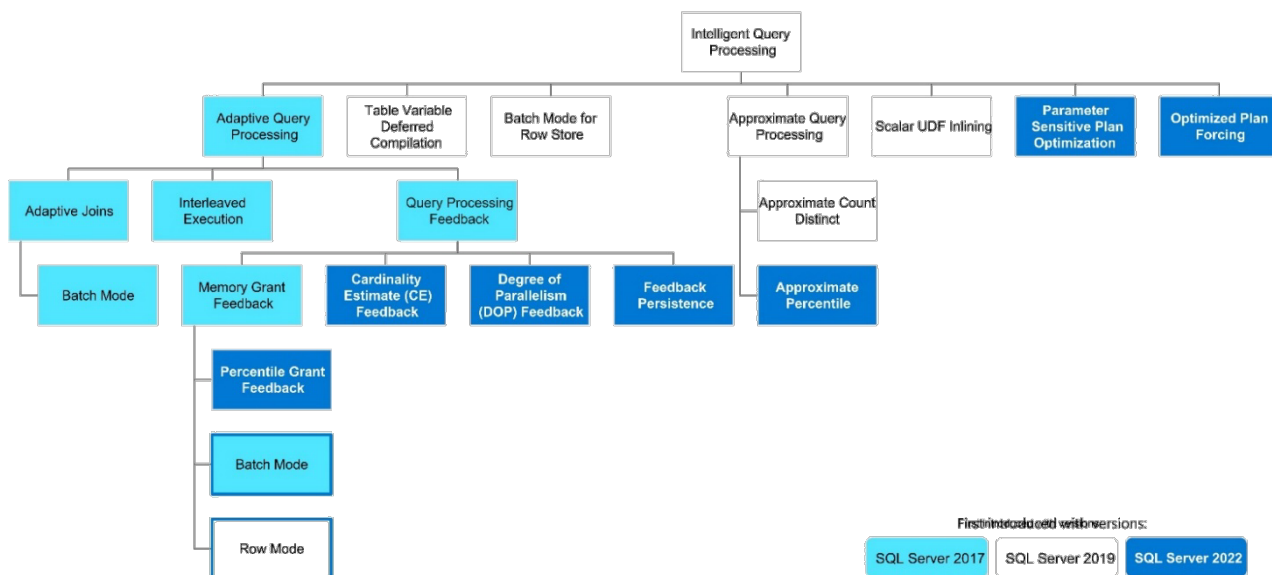
- مروری بر مکانیزم Cache کردن Execution Plan در SQL Server
- آشنایی با Plan Cache در SQL Server
- بررسی DMV های مربوط به مشاهده Plan Cache
- بررسی دستورات مدیریت Plan Cache
- بررسی استفاده مجدد از Execution Plan
- Parameter sniffing چیست و چگونه بوجود می‌آید؟
- راه حل رفع مشکل Parameter Sniffing چیست؟
- بررسی تاثیر استفاده از پروسیجرها در افزایش کارایی کوئری‌ها
- بررسی تکنیک‌هایی کاربردی برای ایجاد پروسیجرهای بهینه
- بررسی استفاده از Dynamic SQL در دستورات TSQL
- مشکلات Ad-Hoc Query ها چیست؟
- Auto Parameterization چیست؟
- Plan Guides چیست؟
- استفاده ویژگی Adaptive Query Processing در SQL Server
- استفاده از ویژگی Automatic Query Tuning در SQL Server
- بررسی تاثیر استفاده از Plan Regression در SQL Server

استفاده از ویژگی Adaptive Query Processing:

یکی از ویژگی‌های جالبی که در SQL Server 2017 ارائه و در SQL Server 2022 امکانات جالبی به آن اضافه شد Adaptive Query processing می‌باشد. با استفاده از این ویژگی‌های مصرف IO و Memory مربوط به کوئری‌ها کاهش می‌یابد که این موضوع باعث می‌شود زمان اجرای کوئری‌های کاهش یابد.

توجه داشته باشید زمانی که ما از Adaptive Query Processing در SQL Server استفاده می‌کنیم نیاز به تغییر در کدهای خود نداریم و SQL Server از ویژگی‌های مربوط به آن به صورت هوشمندانه استفاده خواهد کرد.

ما همیشه در پروژه‌های خودمان از این ویژگی استفاده می‌کنیم. شما چه طور؟



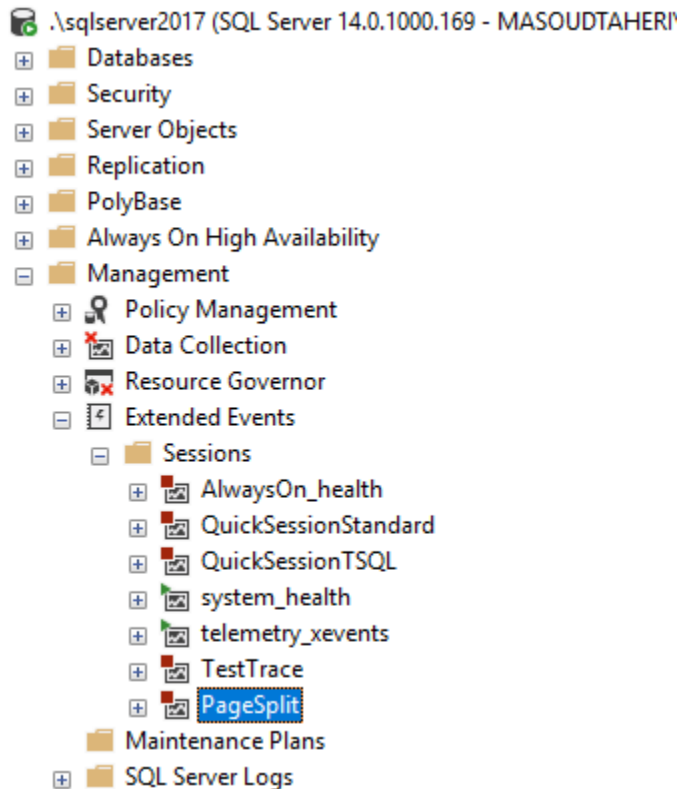
۱۹. استفاده از Extended Event در SQL Server

- آشنایی با Extended Event
- آشنایی با اصطلاحات موجود در Extended Event
- آشنایی با نحوه ایجاد Event Session ها در Extended Event
- نحوه اعمال Filter به ازای Event Session ها
- آشنایی با نحوه آنالیز خروجی Extended Event
- بررسی Event های مشهور

- بررسی سناریو Page Split
- بررسی سناریو استخراج Deadlock
- بررسی سناریو استخراج Blocking
- بررسی سناریو استخراج کوئری‌های کند به همراه Execution Plan

کدام یک از عملیات‌های موجود در سیستم موجب Page Split می‌شود؟

یکی از عوامل مهمی که باعث کاهش کارایی سیستم‌ها هنگام انجام عملیات Insert و Update می‌شود، عملیات Page Split می‌باشد. تیم ما معمولاً در پروژه‌های حرفه‌ای خود یکی از کارهایی که انجام می‌دهد آنالیز عملیات Page Split است برای این کار می‌توانیم از Extended Event ها استفاده کنیم.

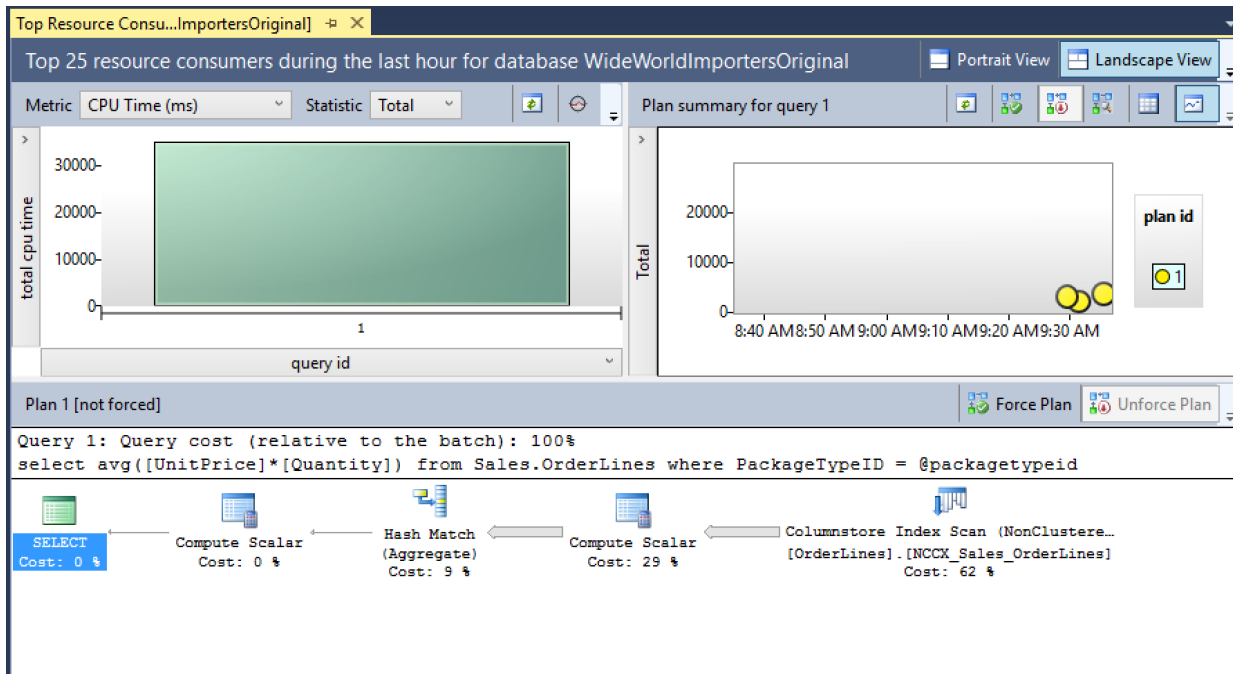


۲۰. استفاده از ویژگی Query Store

- بررسی معماری Query Store
- بررسی نحوه پیکربندی بانک اطلاعاتی برای استفاده از Query Store در SQL Server
- بررسی DMV های مرتبط با Query Store در SQL Server
- بررسی نحوه Force کردن Plan ها با استفاده از Query Store
- بررسی نحوه استفاده از گزارش‌های تکمیلی Query Store

آنالیز کندی سیستم بدون نصب ابزارهای جانبی:

بانک اطلاعاتی یکی از پروژه‌هایی که درگیر آن بودم SQL Server 2016 بود در جهت افزایش سرعت کوئری نیاز داشتیم به سریع‌ترین نحو ممکن از وضعیت اجرای کوئری‌ها و همچنین Execution Plan آنها اطلاع یابیم. یکی از بهترین ابزارهایی که از نسخه 2016 به بعد در SQL Server تعبیه شده است Query Store نام دارد این ابزار به راحتی آب خوردن امکان انجام این کار را به ما می‌دهد.



۲۱. معرفی ویژگی Buffer Pool Extension

- آشنایی با معماری Buffer Pool Extension
- بررسی دلیل استفاده از SSD به عنوان RAM
- آشنایی با نحوه خواندن و نوشتن Page ها در SSD
- آشنایی با Extended Buffer Pool در SQL Server بر روی SSD

غلبه بر مشکلات کمبود حافظه RAM:

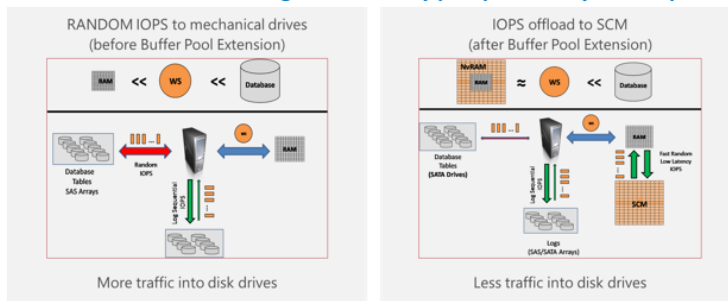
یکی از قابلیت‌های SQL Server 2014 به بالا امکان Extended کردن Buffer Pool در این روش چنانچه ما مشکل Memory داشته باشیم می‌توانیم به کمک SSD بخشی از داده‌هایی که قرار است در

Buffer Pool قرار گیرد را بر روی SSD منتقل کنیم تا کارایی بانک اطلاعاتی به نحوه مناسبی افزایش یابد.

تغییرات Buffer Pool در SQL Server 2014

Extend کردن Buffer Pool به SSD

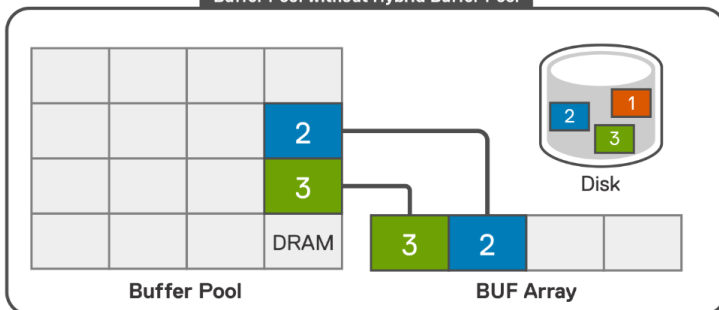
IOPS offload to Storage Class Memory (SCM) in memory hierarchy



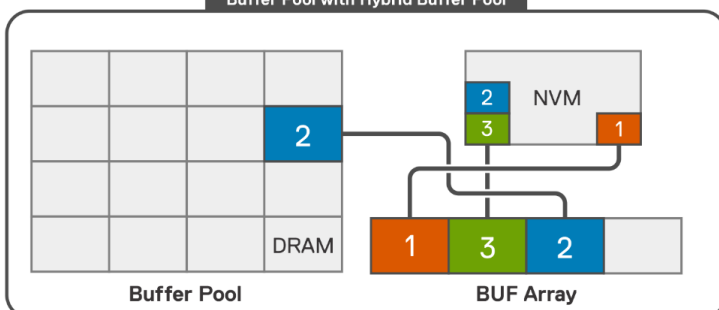
۲۲. بررسی Hybrid Buffer Pool در SQL Server

- معرفی PMEM
- بررسی نحوه استفاده از PMEM در ماشین‌های مجازی
- بررسی استفاده از PMEM به عنوان Hybrid Buffer Pool در SQL Server

Buffer Pool without Hybrid Buffer Pool



Buffer Pool with Hybrid Buffer Pool



استفاده از PMEM بر روی سرورها:

استفاده از سخت‌افزارهای جدید تاثیر زیادی در عملکرد و کارایی SQL Server دارد یکی از آنها PMEM می‌باشد که می‌توان از آن به صورت بهینه بر روی ماشین‌های مجازی و سرورهای SQL Server استفاده کرد.

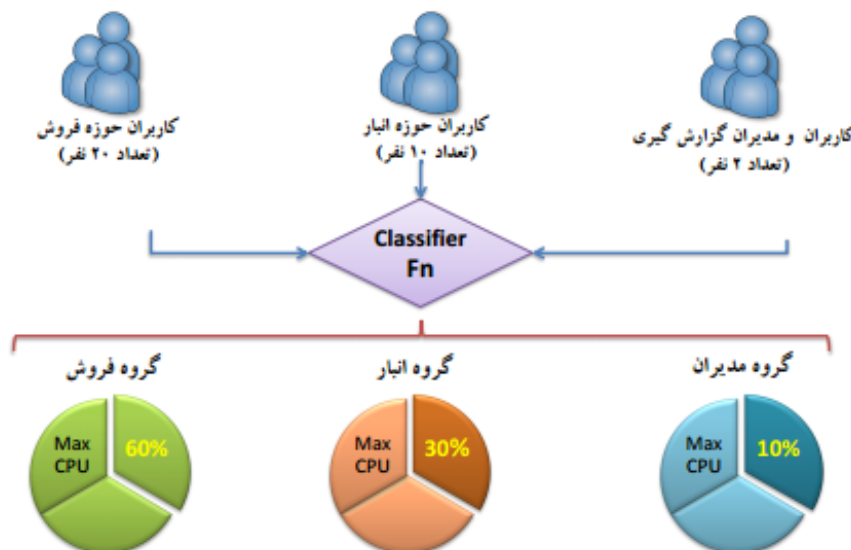
۲۳. معرفی ویژگی Resource Governor

- بررسی Resource Governor
- بررسی سناریوهای قابل استفاده برای Resource Governor در SQL Server
- بررسی نحوه محدود کردن CPU با استفاده از Resource Governor
- بررسی نحوه محدود کردن Memory با استفاده از Resource Governor
- بررسی نحوه محدود کردن IO با استفاده از Resource Governor
- بررسی نحوه ایجاد Classifier Function برای کار با Resource Governor

فراخوانی وب سرویس و افزایش Workload بانک اطلاعاتی:

چند سال پیش در یکی از سازمان‌های دولتی درگیر رفع مشکلات مربوط به کارایی بانک اطلاعاتی بودیم با بررسی که روی سرور انجام دادیم مشخص شد که یک کوئری بهینه (Index Seek) به تعداد زیادی روی سرور در حال اجرا می‌باشد به طوری که در طی ساعتی خاص امکان کار با سرور مهیا نبوده و سرور به علت فراخوانی مکرر این کوئری بهینه Busy می‌باشد. با بررسی دقیق‌تر متوجه شدیم که این کوئری توسط یک وب سرویس در حال فراخوانی است. برای اینکه این وب سرویس کلی Resource‌های سرور را در اختیار نگیرد Resource Governor را پی‌کرن‌دی کردیم.

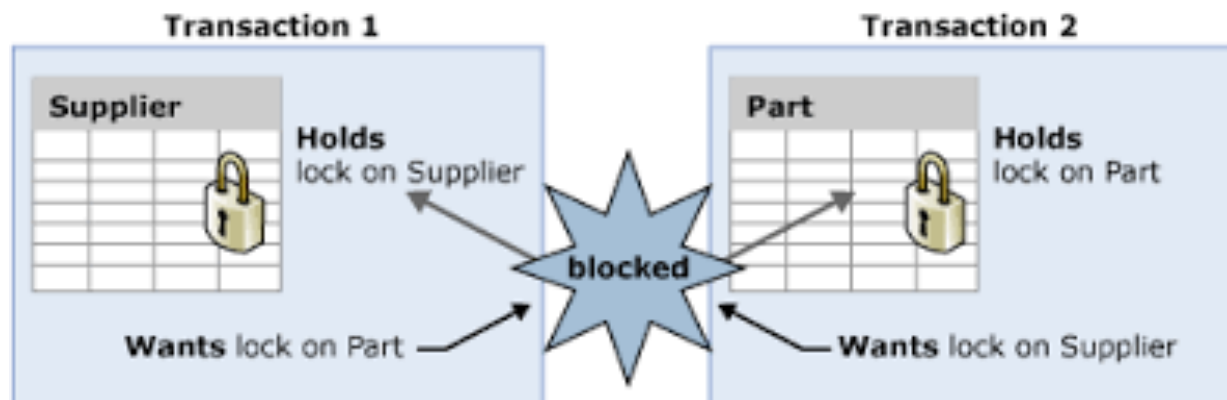
تقسیم بندی منابع با Resource Governor



۲۴. بررسی Lock & Block & Concurrency

- آشنایی با مفهوم Lock
- بررسی مفهوم Concurrency
- بررسی انواع مشکلات رایج هنگام کار با Resource ها به صورت همزمان
- بررسی انواع Lock
- بررسی مفهوم Blocking
- بررسی نحوه بوجود آمدن Blocking
- بررسی نحوه پیدا کردن Blocking
- بررسی مفهوم Deadlock
- بررسی حالت‌های مختلف بوجود آمدن Deadlock
- بررسی روش‌های کشف Deadlock
- بررسی نحوه رفع مشکلات Deadlock و کاهش آن
- بررسی مفهوم Lock Escalation
- بررسی مفهوم Isolation Level
- بررسی انواع Isolation Level
- بررسی Application Lock
- بررسی Schema Lock
- بررسی نحوه طراحی استفاده از Transaction در برنامه‌ها
- بررسی Delay Durability Transaction و Blocking و عوامل بوجود آمدن آن

خیلی از دوستان فکر می‌کنند که وقتی کوئری به کندی اجرای می‌شود عامل آن عدم وجود ایندکس مناسب است. در صورتی که یکی از عوامل کندی اجرای کوئری‌ها Blocking است. یکی از کارهایی که من همیشه در پروژه‌های خودم انجام می‌دهم کشف و بررسی عوامل Blocking است. برای انجام این کار باید حتماً با مفهوم Blocking آشنا باشید پس از آن می‌توان عوامل آن را با استفاده از ابزارهای موجود در SQL Server کشف و مشکلات مربوط به آن‌ها را رفع کرد.

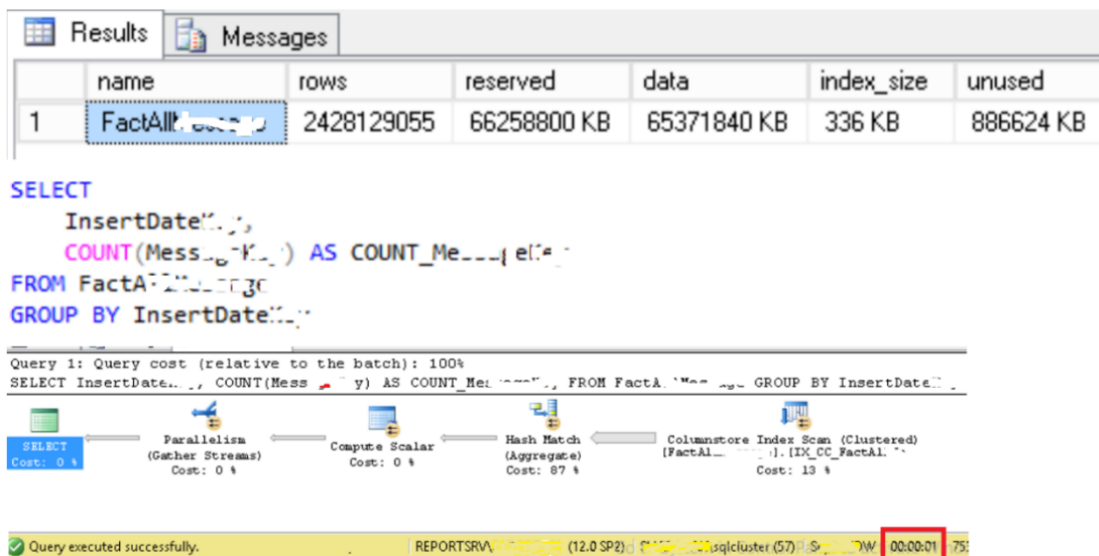


۲۵. Column Store Index

- آشنایی با برخی مباحث موجود در Data Warehouse ها
- آشنایی با Columnstore Index
- بررسی معماری مربوط به Columnstore Index
- آشنایی با مفهوم Batch Mode Processing
- بررسی مفهوم Row Group
- بررسی Delta store و Delete Bitmap در Columnstore Index
- آشنایی با روش‌های فشرده‌سازی مربوط به Columnstore Index
- آشنایی با DMV های مربوط به مانیتور کردن Columnstore Index
- بایدها و نبایدهای مربوط به Columnstore Index
- آشنایی با تغییرات مربوط به SQL Server ۲۰۱۷ به ازای Columnstore Index
- بررسی نحوه اعمال تنظیمات مناسب برای Bulk Insert در Columnstore Index ها
- بررسی تاثیر استفاده از Columnstore Index در Data Warehouse ها
- بررسی نحوه استفاده از Columnstore Index در OLTP ها
- بررسی استفاده از Columnstore Index هنگام کار با Memory Optimized Table

استفاده از Columnstore Index در Data Warehouse :

در یکی از پروژه‌هایی که درگیر بودم جدولی با حدود ۲ میلیارد و ۴۰۰ میلیون رکورد داشتم که روزانه بیش از ۱۰ میلیون رکورد به جدول مربوطه اضافه می‌شود. هدف ما در این پروژه افزایش سرعت کوئری‌ها بود. با توجه به اینکه اغلب کوئری‌های مربوط به این جدول از جنس تحلیلی می‌باشد، تصمیم گرفتیم از Columnstore Index در آن استفاده کنیم. همان طور که در تصویر دقت مشاهده می‌کنید، سرعت اجرای کوئری‌ها به طرز وحشتناکی افزایش پیدا کرد، کوئری که بیش از دو میلیارد و چهارصد میلیون رکورد را در خود درگیر می‌کند در کمتر از یک ثانیه (۳۰۰ میلی ثانیه) روی سرور اجرا شد.





نحوه مشاهده دوره چگونه است؟

این دوره آموزشی را می‌توانید در یک پلیر اختصاصی مشاهده فرمایید، به راحتی می‌توانید این نرم‌افزار را مناسب با سیستم عامل خود (ویندوز، مک، اندروید، لینوکس و یا وب) دانلود نصب کرده. و پس از کپی کلید لایسنس داخل نرم‌افزار محصول خریداری شده را تماشا کنید.

صدور فاکتور رسمی چگونه است؟

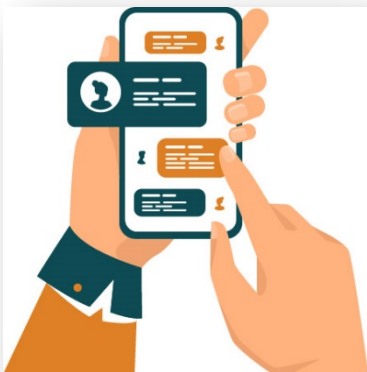
در صورت تمایل به دریافت فاکتور رسمی، پیش از خرید خود با واحد فروش مجموعه (۰۲۱ - ۹۱ ۰۷ ۰۰ ۱۷) تماس حاصل نمایید. شایان ذکر است، امکان صدور فاکتور رسمی پس از خرید آنلاین از سایت مجموعه به هیچ عنوان وجود نخواهد داشت



پشتیبانی بی نظیر

آموزش بدون پشتیبانی کاملاً بی‌معنی است، الان تمام دوره‌های نیک آموز دارای پشتیبانی از طریق سایت و تلگرام است.

پس ثبت و نهایی شدن سفارش شما در سایت نیک‌آموز، تیم پشتیبانی طی ۲۴ تا ۷۲ ساعت کاری با شما تماس خواهند گرفت تا فرایند عضو شدن شما در گروه پشتیبانی تلگرامی هر دوره انجام شود. در صورت وجود هر گونه سوال و ابهامی می‌توانید با شماره‌های شرکت تماس حاصل فرمایید.



آدرس: تهران، یوسف آباد، میدان فرهنگ، خیابان ۳۳، پلاک ۲۹، زنگ ۲، دفتر نیک آموز

شماره تماس: ۰۲۱ - ۹۱ ۰۷ ۰۰ ۱۷ | **موبایل فروش:** ۰۹ ۴۰۰۶ ۲۰۶