

برای بدست آوردن بزرگترین عدد در یک ستون در جدول SQL Server کدام یک از روش های زیر عملکرد بهتری از نظر سرعت و پرفورمنس دارند؟

**Select Max(Column)**

یا

**Select Top(1) Column + Order by Column Desc**

! این موضوع دقیقا بستگی به ساختار جدول دارد. برای اثبات این موضوع دو جدول را زمانی که ایندکس دارند و زمانی که ایندکس ندارند با استفاده از نتیجه **Execution Plan** در **SQL Server Management Studio** بررسی نموده و نتیجه را مقایسه می کنیم.

لطفا دو جدول زیر را در نظر بگیرید:

```
CREATE TABLE [dbo].[tbl1] (  
    [RowID] [int] IDENTITY(1,1) NOT NULL  
    PRIMARY KEY Clustered,  
    [NameOf] [nvarchar](50) NOT NULL,  
    [MaxTotalNumes] [decimal](18, 0) NOT NULL  
)
```

و جدول

```
CREATE TABLE [dbo].[tbl2] (  
    [NameOf] [nvarchar](50) NOT NULL,  
    [MaxTotalNumes] [decimal](18, 0) NOT NULL  
)
```

همانطور که در ساختار جدول ها مشاهده می نمایید جدول اول دارای کلید اصلی و ایندکس می باشد ولی جدول دوم بدون هیچ کلید تعریف شده ای می باشد یا به عبارتی ، نوعی از یک جدول **Heap** است.

جدول های فوق را در **SQL Server** ایجاد می کنیم و تعدادی رکورد در آنها

درج می نماییم.

## نتیجه Execution Plan بر روی جدول دارای ایندکس Clustered

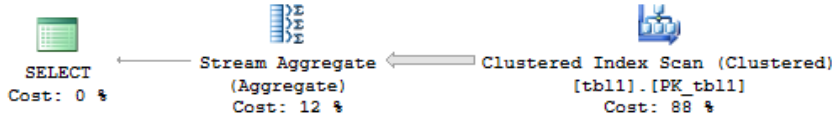
```
SELECT max([MaxTotalNumes]) as MaxTotalNumes
FROM [Studens].[dbo].[tbl1]

SELECT top(1) ([MaxTotalNumes]) as MaxTotalNumes
FROM [Studens].[dbo].[tbl1] order by MaxTotalNumes desc
```

Messages Execution plan

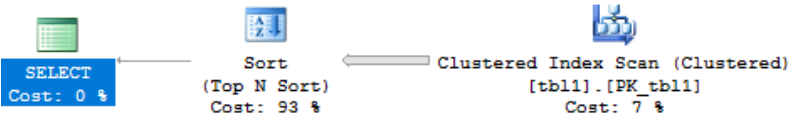
Query 1: Query cost (relative to the batch): 23%

```
SELECT max([MaxTotalNumes]) as MaxTotalNumes FROM [Studens].[dbo].[tbl1]
```



Query 2: Query cost (relative to the batch): 77%

```
SELECT top(1) ([MaxTotalNumes]) as MaxTotalNumes FROM [Studens].[dbo].[tbl1] order by
```



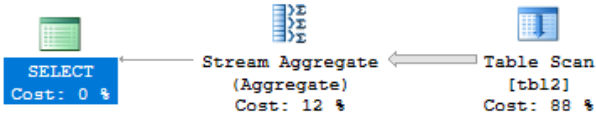
همانطور که در شکل مشخص شده در جدولی که دارای ایندکس از نوع Clustered است دستور Max نتیجه اجرای بهتری داشته است. (23 در مقابل 77). حالا برویم سر جدول دومی که از نوع Heap است...

## نتیجه Execution Plan بر روی جدول از نوع Heap

```
SELECT max([MaxTotalNumes]) as MaxTotalNumes
FROM [Studens].[dbo].[tbl2]
SELECT top(1) ([MaxTotalNumes]) as MaxTotalNumes
FROM [Studens].[dbo].[tbl2] order by MaxTotalNumes desc
```

Messages Execution plan

Query 1: Query cost (relative to the batch): 96%  
SELECT max([MaxTotalNumes]) as MaxTotalNumes FROM [Studens].[dbo].[tbl2]



Query 2: Query cost (relative to the batch): 4%  
SELECT top(1) ([MaxTotalNumes]) as MaxTotalNumes FROM [Studens].[dbo].[tbl2] order by



نتیجه واقعا جالب توجه بود. در جدول Heap استفاده از **Select Top(1)** نتیجه بسیار بهتری نسبت به استفاده از **Max** از نظر پرفورمنسی در اختیار قرار می دهد. (4 در مقابل 96) واقعا نتیجه جالبی است.

پس همیشه این نکته را مد نظر داشته باشید . . . بیشتر وقتها ایندکس گذاری مناسب می تواند باعث افزایش سرعت اجرای کوئری ها شود. توجه داشته باشید که برخی از جدولها ایندکس ندارند. ممکن است از قبل این جدول ها در ساختار پایگاه داده شما موجود باشند و شما هم اجازه ایندکس گذاری نداشته باشید. **اینجاست که اگر نکات پرفورمنسی** را بدانید می توانید کوئری هایی با سرعت اجرای بالاتر داشته باشید . . .

امیدوارم این مطلب به دانش شما افزوده باشه . . . اگر از مطلب خوشتون اومده لطفا مرا دریابید: (لطفا کلیک بفرمائید و Skill های مرا مورد عنایت قرار دهید . . . تا شاید مورد رحمت خداوند قرار گیریم. 😊)

#Blazor #C# #SQL Server #Codefirst #Asp.net core #Programming

[https://www.linkedin.com/in/ali-najafzadeh-gezgalee-](https://www.linkedin.com/in/ali-najafzadeh-gezgalee-12a226227/details/skills/)

[12a226227/details/skills/](https://www.linkedin.com/in/ali-najafzadeh-gezgalee-12a226227/details/skills/)