

Laboratorio 7: Optimización de Consultas

Objetivo: Analizar el plan de ejecución de las consultas en SQL Server y reescribir las consultas o crear índices para mejorar su rendimiento.

Instrucciones generales: Este laboratorio lo deben trabajar en parejas. Hacen una sola entrega por pareja, indicando quiénes trabajaron juntos en el reporte.

Guía de Trabajo

Abra la aplicación SQL Server Management Studio y conéctese al servidor 172.16.202.209. Use la base de datos que tiene por nombre **LabOpt_XX** (donde XX es el número de su pareja), que contiene tres de las tablas de la base de datos AdventureWorks2017. Para su referencia, el esquema conceptual completo de la BD AdventureWorks2017 se encuentra en la [Figura 1](#).

Ejecute los siguientes comandos SQL:

```
SELECT * INTO dbo.SalesOrderHeader FROM Sales.SalesOrderHeader;  
SELECT * INTO dbo.SalesOrderDetail FROM Sales.SalesOrderDetail;  
SELECT * INTO dbo.SalesPerson FROM Sales.SalesPerson;
```

Estos comandos crean una copia de las tablas SalesOrderHeader, SalesOrderDetail y SalesOrderDetail del esquema Sales en el esquema dbo. Las nuevas tablas contienen un espejo de los datos de las tablas originales, pero sin sus restricciones (de llave primaria, llaves foráneas, valores por defecto, etc.). En particular, al no tener restricción de llave primaria, tampoco tienen creado ningún índice sobre este campo.

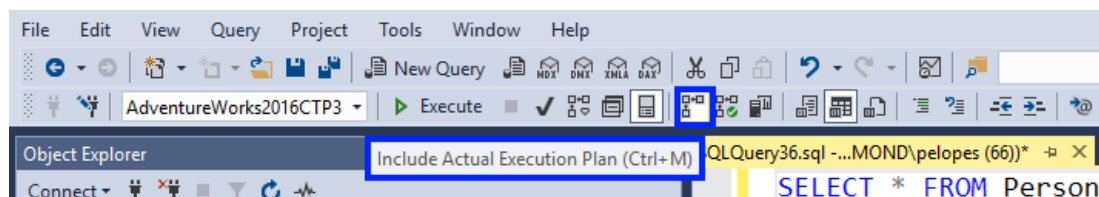
1. Ejercicio 1

a. Ejecute las siguientes consultas (marque ambas y ejecútelas juntas):

```
SELECT ProductID  
FROM Sales.SalesOrderDetail;  
  
SELECT ProductID  
FROM dbo.SalesOrderDetail;
```

(Tome una captura de pantalla del resultado de ambas consultas e inclúyala en el reporte de laboratorio).

b. Muestre el plan de ejecución real para las consultas del punto 1(a). Para ello, use el botón de la barra de herramientas que está resaltado en la siguiente imagen:



(Tome una captura de pantalla del plan de ejecución de ambas consultas e inclúyala en el reporte de laboratorio.)

- c. ¿Cuál es el costo porcentual (relativo al lote o *batch*) de cada una de las consultas? Esta información se despliega en el mismo plan de ejecución, donde dice “Query 1” y “Query 2”.
- d. ¿A qué atribuye la gran diferencia entre el costo de la consulta sobre el esquema Sales y la consulta sobre el esquema dbo? Para contestar esta pregunta, analice el plan de ejecución de cada consulta y revise si las tablas involucradas tienen algún índice que esté siendo usado en alguna de las consultas.
- e. Cree un índice sobre la tabla `dbo.SalesOrderDetail` que permita mejorar el rendimiento de la consulta:

```
SELECT ProductID
FROM dbo.SalesOrderDetail;
```

(Incluya en el reporte de laboratorio el código SQL de creación del índice)

- f. Para comprobar si el índice fue efectivo, vuelva a ejecutar las consultas del punto 1(a) y revise si el índice que creó está siendo utilizado en el plan de ejecución. (Tome una captura de pantalla del plan de ejecución de ambas consultas e inclúyala en el reporte de laboratorio.) Además, compare el costo porcentual (relativo al lote o *batch*) de cada consulta y note si hubo algún cambio con lo observado en el punto 1(c). Explique.
- g. Elimine el índice creado en el punto 1(e).

2. Ejercicio 2

- a. Ejecute la siguiente consulta:

```
SELECT count(*)
FROM dbo.SalesOrderHeader
WHERE DATEPART(YEAR, OrderDate) = '2014'
```

- b. Muestre el plan de ejecución real para la consulta del punto 2(a). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.)

- c. ¿Cómo resolvió SQL Server esta consulta? ¿Cuáles operadores físicos usó?
- d. Ahora ejecute esta consulta, que tiene su cláusula WHERE modificada pero da el mismo resultado que la consulta del punto 2(a):

```
SELECT count(*)  
FROM dbo.SalesOrderHeader  
WHERE OrderDate >= '20140101' AND OrderDate < '20150101';
```

- e. Muestre el plan de ejecución real para la consulta del punto 2(d). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.)
- f. ¿Cómo resolvió SQL Server esta consulta? ¿Cuáles operadores físicos usó? ¿Qué sugerencia de índice le da SQL Server para optimizar la consulta (i.e., *missing index*)?
- g. Cree un índice de acuerdo a la recomendación de SQL Server. (Incluya en el reporte de laboratorio el código SQL de creación del nuevo índice.)
- h. Vuelva a ejecutar la consulta del punto 2(d) y revise el plan de ejecución. (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Hay alguna diferencia entre los operadores que está usando este plan de ejecución y el del punto 2(e) –antes de crear el índice–? ¿Está SQL Server usando el índice creado en el punto 2(g)? Explique.
- i. Vuelva a ejecutar la consulta del punto 2(a) y revise el plan de ejecución. (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Hay alguna diferencia entre los operadores que está usando este plan de ejecución y el del punto 2(b)? ¿Por qué no está SQL Server usando el índice creado en el punto 2(g)? Explique.
- j. Elimine el índice creado en el punto 2(g).

3. Ejercicio 3

- a. Ejecute la siguiente consulta:

```
SELECT *  
FROM dbo.SalesOrderHeader  
WHERE TotalDue BETWEEN 500 AND 40000;
```

- b. Muestre el plan de ejecución real para la consulta del punto 3(a). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.)
- c. ¿Cómo resolvió SQL Server esta consulta? ¿Cuáles operadores físicos usó?
- d. Cree un índice sobre el atributo TotalDue de la tabla dbo.SalesOrderHeader para mejorar el rendimiento de la consulta. (Incluya en el reporte de laboratorio el código SQL de creación del índice.)

- e. Vuelva a ejecutar la consulta del punto 3(a) y revise el plan de ejecución. (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Por qué cree que SQL Server no está usando el índice que creó en el punto 3(d)?
- f. Ahora cambie la consulta 3(a) para que retorne TotalDue en lugar de todos los atributos de la tabla, y ejecútela:

```
SELECT TotalDue
FROM dbo.SalesOrderHeader
WHERE TotalDue BETWEEN 500 AND 40000;
```

- g. Revise el plan de ejecución para la consulta del punto 3(f). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Cuáles operadores físicos se están usando? ¿Se está usando el índice creado en el punto 3(d)?
- h. Cambie la consulta 2(f) para que retorne también SalesOrderID, y ejecútela:

```
SELECT SalesOrderID, TotalDue
FROM dbo.SalesOrderHeader
WHERE TotalDue BETWEEN 500 AND 40000;
```

- i. Revise el plan de ejecución para la consulta del punto 3(h). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Por qué cree que SQL Server no está usando el índice en este caso? ¿Le da SQL Server alguna sugerencia de índice para optimizar la consulta (*missing index*)?
- j. Elimine el índice creado en el punto 3(d) y cree otro nuevo según lo sugerido por SQL Server. (Incluya en el reporte de laboratorio el código SQL de creación del nuevo índice) ¿Cuál es la diferencia entre el índice original y el sugerido por SQL Server?
- k. Vuelva a ejecutar la consulta 3(f) y revise el plan de ejecución. (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Cuáles operadores físicos se están usando? ¿Se está usando el nuevo índice del punto 3(j)? Explique.
- l. Cambie la consulta 3(h) para que use la función valor absoluto, y ejecútela:

```
SELECT SalesOrderID, TotalDue
FROM dbo.SalesOrderHeader
WHERE ABS(TotalDue) BETWEEN 500 AND 40000;
```

- m. Revise el plan de ejecución para la consulta del punto 3(l). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Está SQL Server usando el operador *Index Scan* o el *Index Seek* sobre el índice? Compare este operador contra el operador que mostraba el plan de ejecución del punto 3(k).

- n. Ejecute las consultas 3(l) y 3(h) juntas, así:

```
SELECT SalesOrderID, TotalDue
FROM dbo.SalesOrderHeader
WHERE TotalDue BETWEEN 500 AND 40000;

SELECT SalesOrderID, TotalDue
FROM dbo.SalesOrderHeader
WHERE ABS(TotalDue) BETWEEN 500 AND 40000;
```

- o. Muestre el plan de ejecución real para las consultas del punto 3(n). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Cuál es el costo porcentual (relativo al lote o *batch*) de cada una de las consultas? Reflexione sobre el efecto que tiene el uso de funciones en la optimización de consultas.
- p. Elimine el índice creado en el punto 3(j).

4. Ejercicio 4

- a. Ejecute las siguientes consultas (marque ambas y ejecútelas juntas):

```
SELECT h.SalesOrderID, d.SalesOrderDetailID, h.SalesPersonID
FROM Sales.SalesOrderHeader h
JOIN Sales.SalesOrderDetail d
ON d.SalesOrderID = h.SalesOrderID
JOIN Sales.SalesPerson p
ON p.BusinessEntityID = h.SalesPersonID;

SELECT h.SalesOrderID, d.SalesOrderDetailID, h.SalesPersonID
FROM dbo.SalesOrderHeader h
JOIN dbo.SalesOrderDetail d
ON d.SalesOrderID = h.SalesOrderID
JOIN dbo.SalesPerson p
ON p.BusinessEntityID = h.SalesPersonID;
```

(Tome una captura de pantalla del resultado de ambas consultas e inclúyala en el reporte de laboratorio).

- b. Muestre el plan de ejecución real para la consulta del punto 4(a). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.)

- c. ¿Cuál es el costo porcentual (relativo al lote o *batch*) de cada una de las consultas?
- d. ¿A qué atribuye la diferencia entre el costo de estas consultas? Para contestar esta pregunta, analice el plan de ejecución de cada consulta y revise si se usa algún índice y qué tipo de operadores físicos está usando SQL Server en cada caso.
- e. Intente mejorar el rendimiento de la segunda consulta del punto 4(a) -la que se hace sobre el esquema *dbo*- mediante la creación de índices. Se le sugiere revisar la estrategia recomendada en <https://www.toptal.com/sql-server/sql-database-tuning-for-developers> para crear índices cuando hay *joins*. (Incluya en el reporte de laboratorio el código SQL con la creación de los índices.)
- f. Vuelva a ejecutar las consultas del punto 4(a) y revise el plan de ejecución. (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.) ¿Cuál es el costo porcentual (relativo al lote o *batch*) de cada consulta ahora? ¿Hubo una mejora en el tiempo de ejecución de la segunda consulta? ¿Está SQL Server usando los índices que creó en el punto 4(e)? Explique.
- g. Elimine los índices creados en el punto 4(e).

5. Ejercicio 5

- a. Ejecute la siguiente consulta:

```
SELECT SalesOrderID, SalesPersonID, ShipDate
FROM   dbo.SalesOrderHeader
WHERE  SalesPersonID IN
      (SELECT BusinessEntityID
       FROM   dbo.SalesPerson
       WHERE  TerritoryID > 5)
AND ShipDate > '2014-01-01'
```

- b. Muestre el plan de ejecución real para la consulta del punto 5(a). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.)
- c. Re-escriba la consulta 5(a) en otra equivalente que no tenga una consulta anidada, y ejecútela. (Incluya en el reporte de laboratorio el código SQL de la consulta modificada.)
- d. Muestre el plan de ejecución real para la consulta del punto 5(c). (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de laboratorio.)
- e. Cree uno o varios índices que mejoren el rendimiento de la consulta 5(c). (Incluya en el reporte de laboratorio el código SQL de creación de los índices.)
- f. Vuelva a ejecutar la consulta del punto 5(c) y analice su plan de ejecución. (Tome una captura de pantalla del plan de ejecución e inclúyala en el reporte de

