



INSTITUTO POLITECNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN
INGENIERIA Y TECNOLOGIAS AVANZADAS

Planes de ejecución

PRACTICA 2

INGENIERIA TELEMÁTICA BASE DE DATOS DISTRIBUIDAS

PROFESOR: De la Cruz Sosa Carlos

ALUMNOS:

Legorreta Rodriguez Maria Fernanda

Macías Galván Arturo Daniel

Palacios Reyes Leslie Noemi

GRUPO: 3TM3

SEMESTRE 2025/2

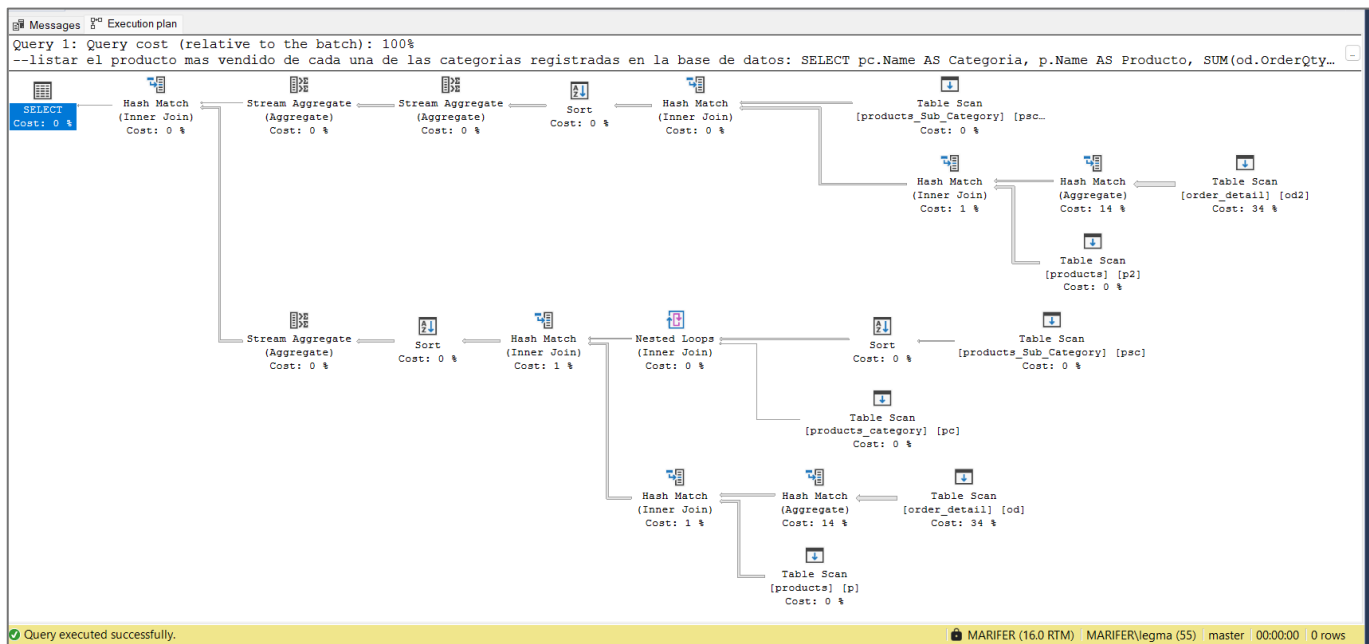
No. de consulta	1
Base de datos	PracticaPE
Descripción	Listar el producto más vendido de cada una de las categorías registradas en la base de datos.
Requisitos	Solo seleccionar las tablas especificadas de la base de datos AdventureWorks2022
Significado de los valores de los catálogos	N/A
Análisis lógico sin índices	<p>Se unen las tablas de detalles de órdenes, productos, subcategorías y categorías. Luego, se agrupa por categoría y producto, y se suma la cantidad vendida de cada producto.</p> <p>Después, usa la cláusula HAVING para quedarse solo con el producto que más se vendió dentro de cada categoría. Para saber cuál es ese producto, compara las ventas de cada uno con el número más alto de ventas en su categoría, que se calcula aparte en una subconsulta. Al final, se muestra una lista con la categoría, el producto más vendido de esa categoría, y cuántas unidades se vendieron.</p>
Análisis lógico con índices	En el optimizador de consultas, se eligió índice no agrupado en lugar de un Table Scan completo, pues se considera que puede ser más eficiente acceder primero por el índice y luego buscar los datos restantes si es que se necesita. Lo que reduce el costo de ejecución si es que el índice cubre la consulta.

Resultados:

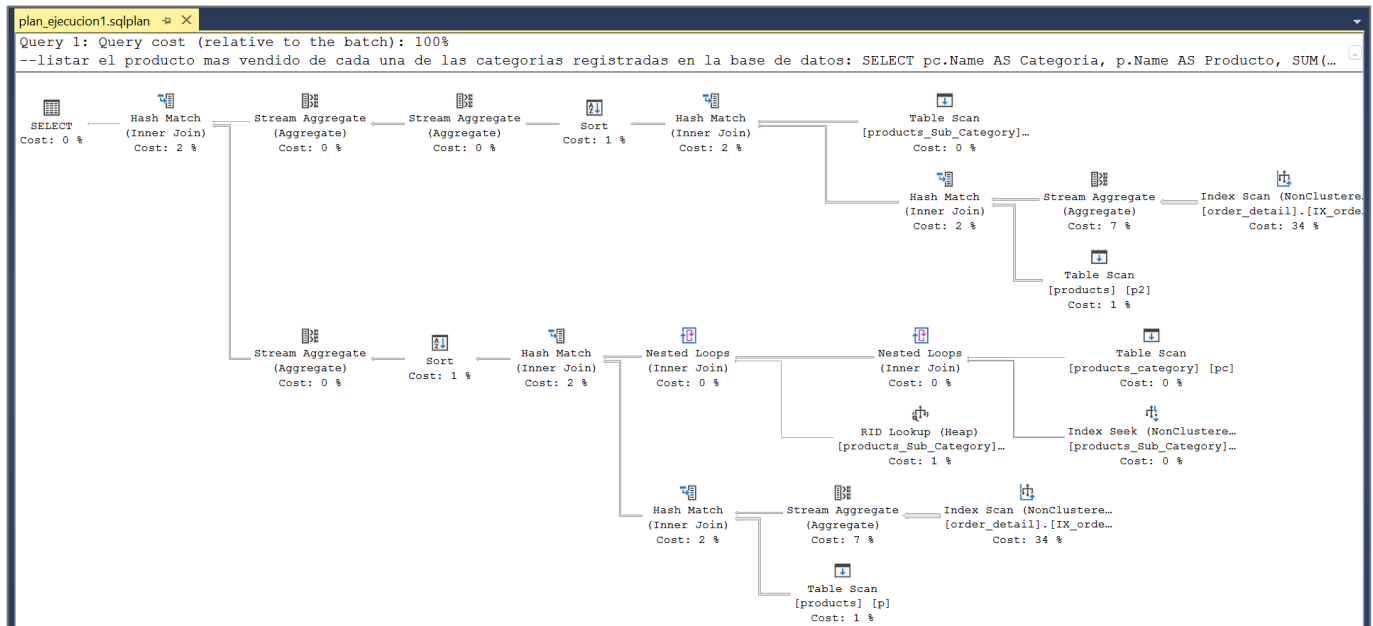
Results		Messages	
	Categoria	Producto	CantidadVendida
1	Bikes	Mountain-200 Black, 38	2977
2	Components	LL Road Frame - Black, 52	1581
3	Clothing	AWC Logo Cap	8311
4	Accessories	Water Bottle - 30 oz.	6815

MARIFER (16.0 RTM) | MARIFER\legma (55) | master | 00:00:00 | 4 rows

Plan de ejecución sin índices:



Plan de ejecución con índices:



Análisis:

Como análisis general de la primera consulta, se puede decir que al trabajar con el índice no agrupado nos ayuda a reducir la carga de lectura y mejora la eficiencia del plan de ejecución al permitir escanear solo una parte de los datos es por eso que se pueden ver RID Lookup si se habla de un heap o Key Lookup si se está hablando de un clustered.

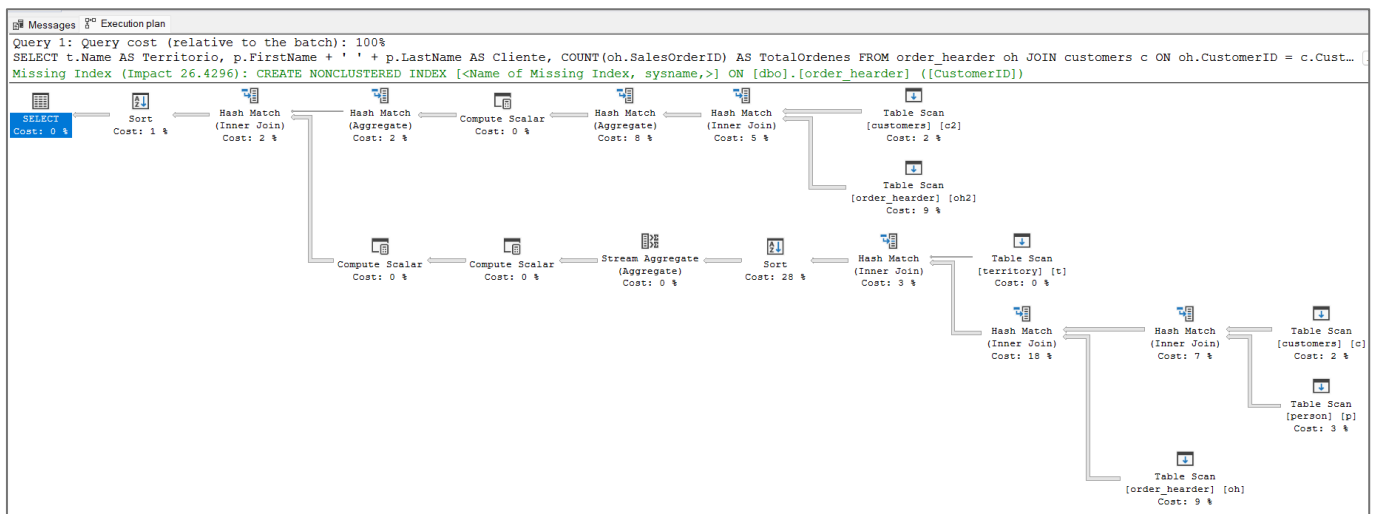
No. de consulta	2
Base de datos	PracticaPE
Descripción	Listar el nombre de los clientes con más ordenes por cada uno de los territorios registrados en la base de datos
Requisitos	Solo seleccionar las tablas especificadas de la base de datos AdventureWorks2022
Significado de los valores de los catálogos	N/A
Análisis lógico sin índices	Esta consulta busca encontrar, por cada territorio, al cliente que ha hecho la mayor cantidad de órdenes. Primero, une varias tablas (order_header, customers, person y territory) para obtener el nombre del territorio y del cliente. Luego agrupa los datos por territorio y cliente, contando cuántas órdenes hizo cada uno. En la cláusula HAVING, se filtra para conservar solo aquellos clientes cuyo número de órdenes sea igual al máximo dentro de su territorio, el cual se calcula con una subconsulta que agrupa y cuenta las órdenes por cliente dentro del mismo territorio. Así, al final se muestran únicamente los clientes top de cada territorio, ordenados alfabéticamente por territorio y nombre.
Análisis lógico con índices	Se crearon índices para mejorar el rendimiento de la consulta, reduciendo Table Scan y permitiendo Index Scan. Esto optimizó los JOIN, agrupaciones y filtrados, disminuyendo el costo de ejecución y acelerando la respuesta. La indexación fue clave para que la consulta funcione eficientemente con grandes volúmenes de datos.

Resultados:

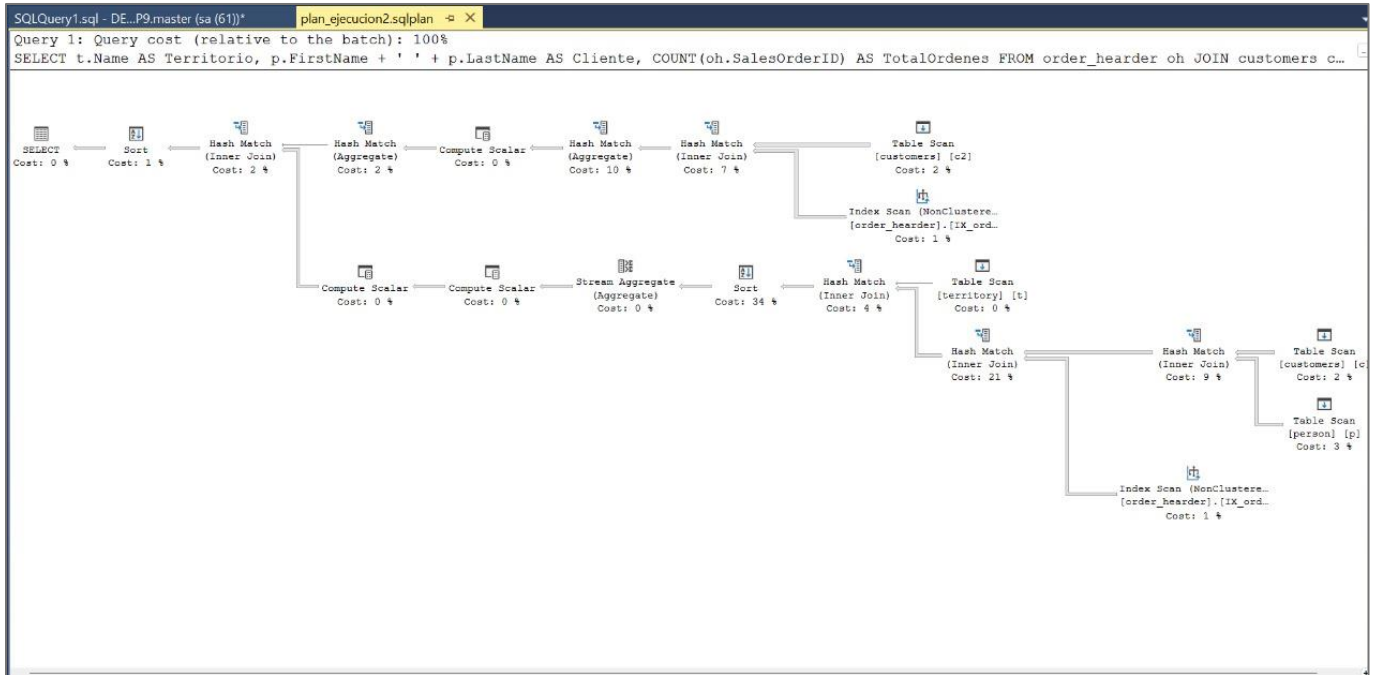
	Territorio	Cliente	TotalOrdenes
1	Australia	Barbara Decker	4
2	Australia	Brenda Barlow	4
3	Australia	Bridget Browqett	4
4	Australia	Bruno Deniut	4
5	Australia	Denis Taylor	4
6	Australia	Eric Coleman	4
7	Australia	Gracia Tuell	4
8	Australia	James Clark	4
9	Australia	Jeff Hay	4
10	Australia	Jeff Henshaw	4
11	Australia	Jenny Lysaker	4
12	Australia	Judy Storjohann	4
13	Australia	Kok-Ho Loh	4
14	Australia	Mae Anderson	4
15	Australia	Nancy Buchan...	4
16	Australia	Neal Hasty	4
17	Australia	Neil Charney	4
18	Australia	Phillip Basalge	4

Query executed succes... | MARIFER (16.0 RTM) | MARIFER\legma (55) | master | 00:00:00 | 92 rows

Plan de ejecución sin índices:



Plan de ejecución con índices:



Análisis:

La consulta identifica al cliente con más órdenes por territorio. Usa JOIN, GROUP BY y una subconsulta en HAVING para encontrar al cliente con el máximo de órdenes. Está bien construida, pero requiere índices para evitar sobrecarga en el procesamiento. Es útil para análisis por región y comportamiento del cliente.

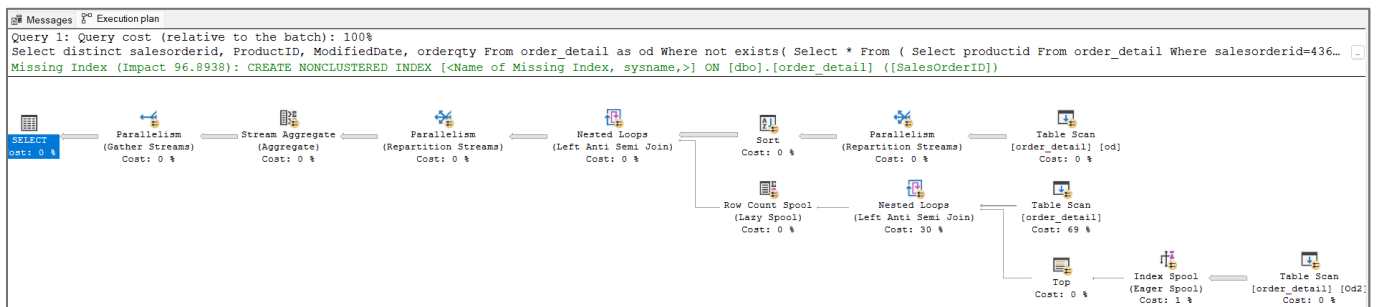
No. de consulta	3
Base de datos	PracticaPE
Descripción	Listar los datos generales de las ordenes que tengan al menos los mismos productos de la orden con salesorderid=43676
Requisitos	Solo seleccionar las tablas especificadas de la base de datos AdventureWorks2022
Significado de los valores de los catálogos	N/A
Análisis lógico sin índices	Primero, se extraen los productid de la orden con SalesOrderID = 43676. Luego, para cada orden en la tabla, se verifica que no falte ninguno de esos productos usando una doble condición NOT EXISTS, lo que implementa la lógica de “subconjunto”: si una orden contiene todos los productos de la orden 43676 (aunque tenga más), será seleccionada. Finalmente, se muestran los datos distintivos (salesorderid, productid, modifieddate, orderqty) de las órdenes que cumplen con esa condición
Análisis lógico con índices	El índice (SalesOrderID, ProductID) mejora el acceso a los datos evitando escaneos completos. Optimiza la evaluación del NOT EXISTS, reduciendo el uso de CPU y tiempo de ejecución.

Resultados

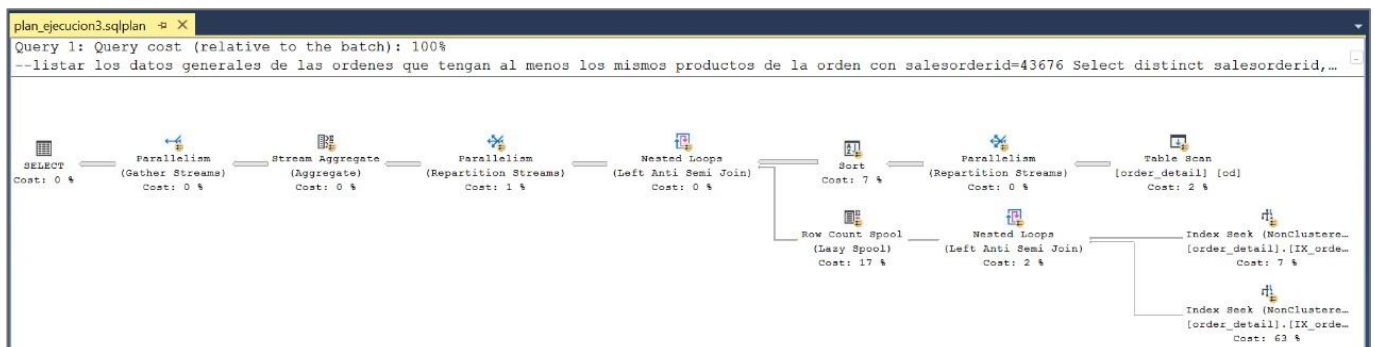
	salesorderid	ProductID	ModifiedDate	orderqty
1	43891	709	2011-07-01 00:00:00.000	4
2	43891	771	2011-07-01 00:00:00.000	3
3	43891	775	2011-07-01 00:00:00.000	2
4	44783	709	2011-10-31 00:00:00.000	6
5	44783	775	2011-10-31 00:00:00.000	4
6	44783	777	2011-10-31 00:00:00.000	2
7	43676	710	2011-05-31 00:00:00.000	1
8	43676	775	2011-05-31 00:00:00.000	3
9	43676	776	2011-05-31 00:00:00.000	2
10	43891	710	2011-07-01 00:00:00.000	2
11	43900	709	2011-07-01 00:00:00.000	8
12	43900	710	2011-07-01 00:00:00.000	4
13	43900	771	2011-07-01 00:00:00.000	4
14	43900	775	2011-07-01 00:00:00.000	3
15	43900	776	2011-07-01 00:00:00.000	2
16	43900	777	2011-07-01 00:00:00.000	1

Query execu... | MARIFER (16.0 RTM) | MARIFER\legma (55) | master | 00:01:48 | 183 rows

Plan de ejecución sin índices:



Plan de ejecución con índices:



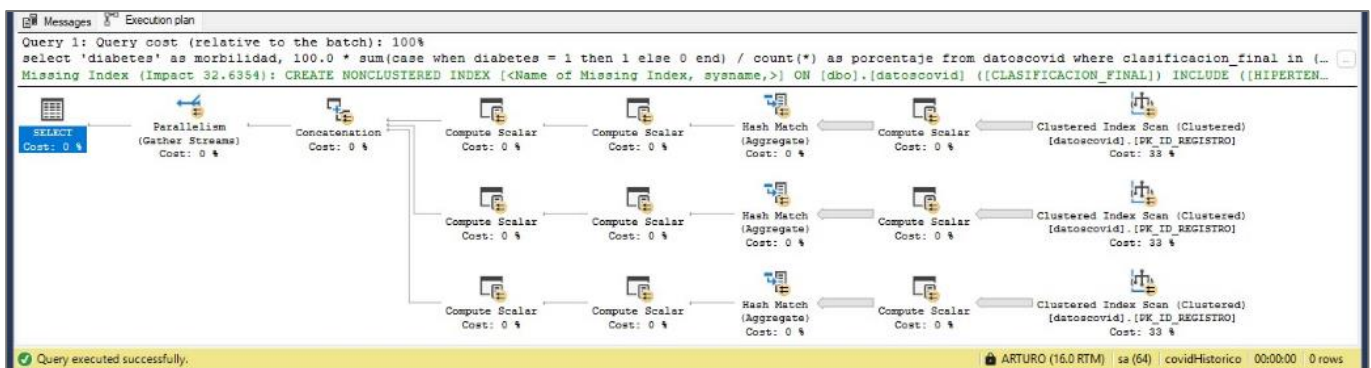
Análisis:

El índice permite localizar rápidamente las combinaciones de SalesOrderID y ProductID, haciendo la consulta más eficiente. Disminuye los Table Scan y mejora el rendimiento general de la consulta.

BASE DE DATOS COVID:

CONSULTA 3

Plan de ejecución sin índices



Clustered Index Scan (Clustered)

Scanning a clustered index, entirely or only a range.

Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (33%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	5613880
Estimated Number of Rows Per Execution	5613880
Estimated Row Size	15 B
Ordered	False
Node ID	6

Predicate

[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(1) OR
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(2) OR
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(3)

Object

[covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO]

Output List

[covidHistorico].[dbo].[datoscovid].DIABETES

Clustered Index Scan (Clustered)

Scanning a clustered index, entirely or only a range.

Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (33%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	5613880
Estimated Number of Rows Per Execution	5613880
Estimated Row Size	15 B
Ordered	False
Node ID	11

Predicate

[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(1) OR
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(2) OR
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(3)

Object

[covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO]

Output List

[covidHistorico].[dbo].[datoscovid].OBESIDAD

Clustered Index Scan (Clustered)

Scanning a clustered index, entirely or only a range.

Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (33%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	5613880
Estimated Number of Rows Per Execution	5613880
Estimated Row Size	15 B
Ordered	False
Node ID	16

Predicate

[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(1) OR
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(2) OR
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(3)

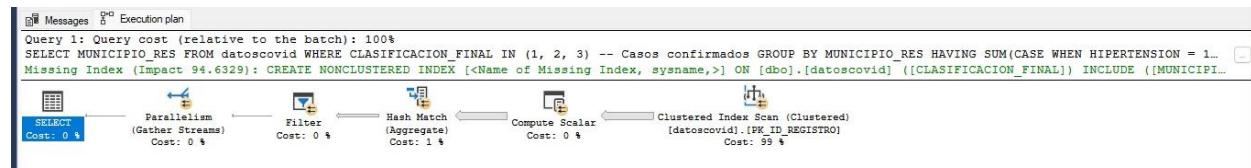
Object

[covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO]

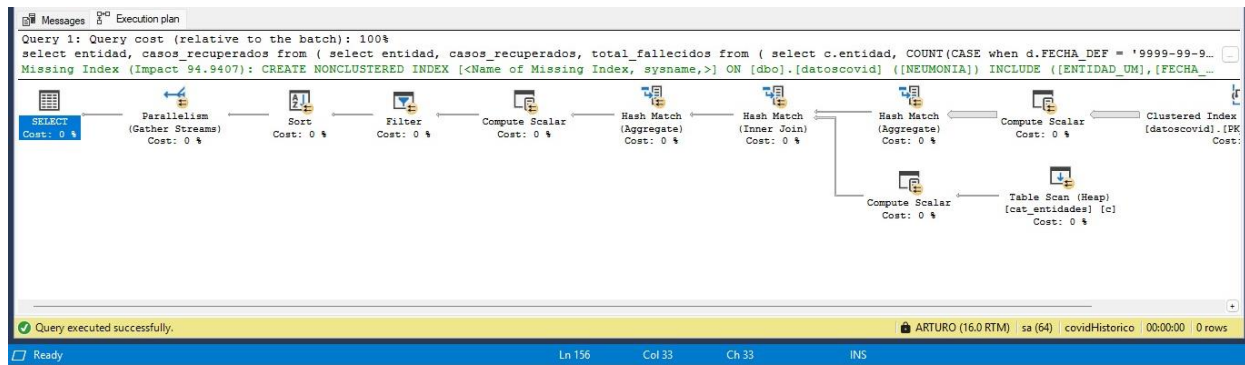
Output List

[covidHistorico].[dbo].[datoscovid].HIPERTENSION

CONSULTA 4



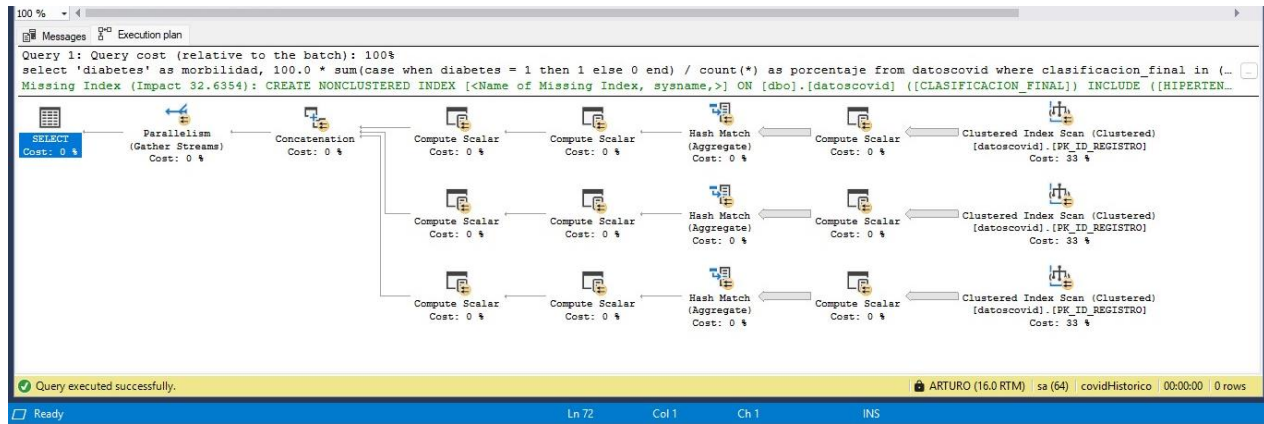
CONSULTA 5



Clustered Index Scan (Clustered)	
Scanning a clustered index, entirely or only a range.	
Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (99%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	735333
Estimated Number of Rows Per Execution	735333
Estimated Row Size	36 B
Ordered	False
Node ID	9
Predicate	
[covidHistorico].[dbo].[datoscovid].[NEUMONIA] as [d].[NEUMONIA]=(1)	
Object	
[covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO] [d]	
Output List	
[covidHistorico].[dbo].[datoscovid].ENTIDAD_UM, [covidHistorico].[dbo].[datoscovid].FECHA_DEF	

CONSULTAS DESPUÉS DE ÍNDICES

CONSULTA 3



Clustered Index Scan (Clustered)

Scanning a clustered index, entirely or only a range.

Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (33%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	5613880
Estimated Number of Rows Per Execution	5613880
Estimated Row Size	15 B
Ordered	False
Node ID	6

Predicate

[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(1) OR
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(2) OR
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(3)

Object

[covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO]

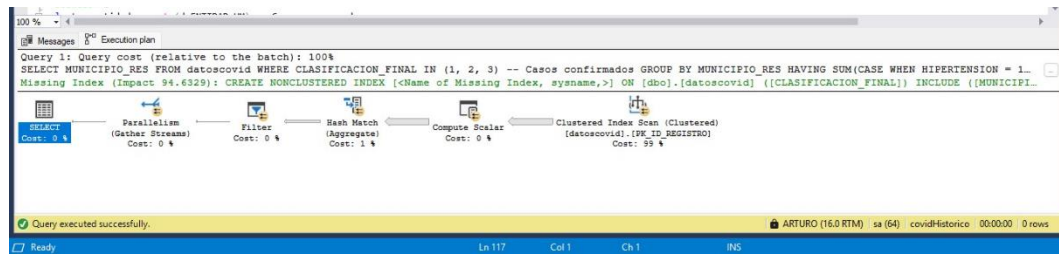
Output List

[covidHistorico].[dbo].[datoscovid].DIABETES

Clustered Index Scan (Clustered)	
Scanning a clustered index, entirely or only a range.	
Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (33%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	5613880
Estimated Number of Rows Per Execution	5613880
Estimated Row Size	15 B
Ordered	False
Node ID	11
Predicate	
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(1) OR [covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(2) OR [covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(3)	
Object	
[covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO]	
Output List	
[covidHistorico].[dbo].[datoscovid].OBESIDAD	

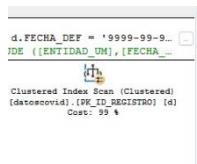
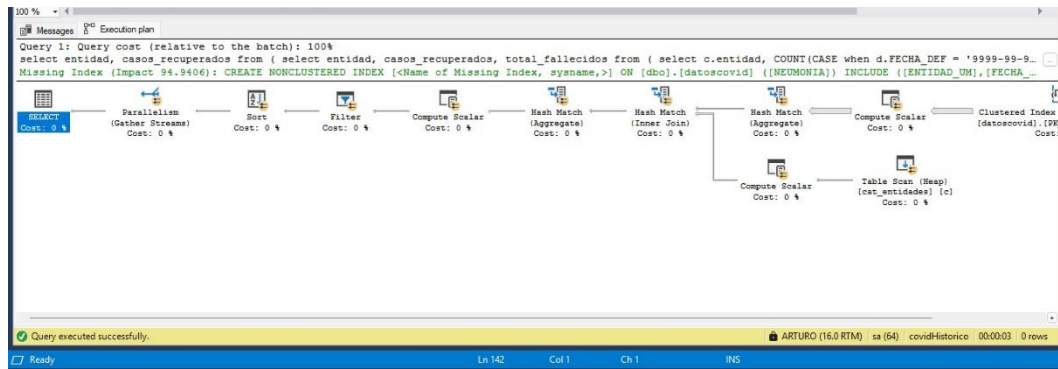
Clustered Index Scan (Clustered)	
Scanning a clustered index, entirely or only a range.	
Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (33%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	5613880
Estimated Number of Rows Per Execution	5613880
Estimated Row Size	15 B
Ordered	False
Node ID	16
Predicate	
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(1) OR [covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(2) OR [covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(3)	
Object	
[covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO]	
Output List	
[covidHistorico].[dbo].[datoscovid].HIPERTENSION	

CONSULTA 4



Clustered Index Scan (Clustered)	
Scanning a clustered index, entirely or only a range.	
Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (99%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	5613880
Estimated Number of Rows Per Execution	5613880
Estimated Row Size	37 B
Ordered	False
Node ID	4
Predicate	
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(1) OR	
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(2) OR	
[covidHistorico].[dbo].[datoscovid].[CLASIFICACION_FINAL]=(3)	
Object	
[covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO]	
Output List	
[covidHistorico].[dbo].[datoscovid].MUNICIPIO_RES, [covidHistorico].	
[dbo].[datoscovid].DIABETES, [covidHistorico].[dbo].	
[datoscovid].HIPERTENSION, [covidHistorico].[dbo].	
[datoscovid].OBESIDAD, [covidHistorico].[dbo].[datoscovid].TABAQUISMO	

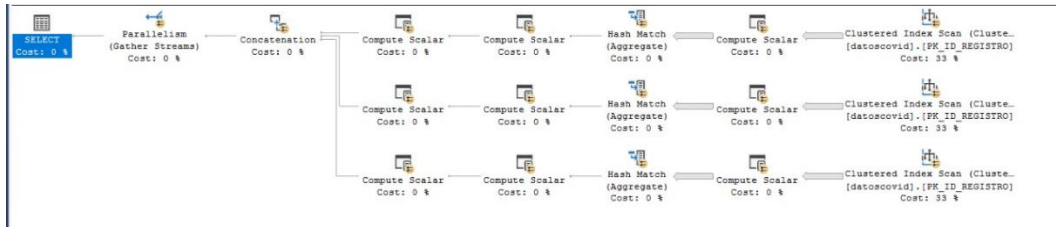
CONSULTA 5



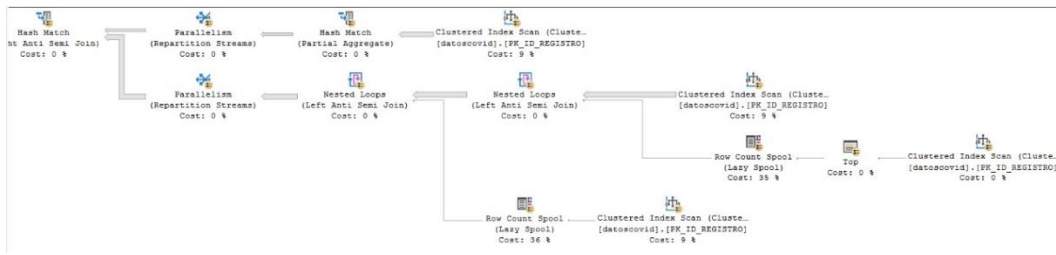
Clustered Index Scan (Clustered)	
Scanning a clustered index, entirely or only a range.	
Physical Operation	Clustered Index Scan
Logical Operation	Clustered Index Scan
Estimated Execution Mode	Batch
Storage	RowStore
Estimated Operator Cost	394.294 (99%)
Estimated I/O Cost	390.096
Estimated Subtree Cost	394.294
Estimated CPU Cost	4.19808
Estimated Number of Executions	1
Estimated Number of Rows to be Read	15265600
Estimated Number of Rows for All Executions	748959
Estimated Number of Rows Per Execution	748959
Estimated Row Size	36 B
Ordered	False
Node ID	9
Predicate [covidHistorico].[dbo].[datoscovid].[NEUMONIA] as [d].[NEUMONIA]=(1)	
Object [covidHistorico].[dbo].[datoscovid].[PK_ID_REGISTRO] [d]	
Output List [covidHistorico].[dbo].[datoscovid].ENTIDAD_UM, [covidHistorico].[dbo].[datoscovid].FECHA_DEF	

CONSULTAS DE EQUIPO DISTINTO (Equipo 3)

CONSULTA 3



CONSULTA 4



CONSULTA 5



COMPARACIÓN

CONSULTA 3

En esta consulta, se puede apreciar en ambos planes de ejecución, que la similitud es perfecta, es decir, que ambos planes de ejecución, tanto del equipo externo, como el nuestro es el mismo. Esto se puede observar claramente en que, en los 3 nodos finales, cada uno cuenta con un 33%, representando una tercera parte aproximada del procesamiento usado en la consulta.

CONSULTA 4

En la siguiente consulta, se muestra una gran discrepancia entre ambas consultas de los dos equipos, pues como primera instancia, se observa que nuestro equipo tiene un plan de ejecución lineal, siendo una consulta que el nodo final lleva una capacidad de 99%, mientras que, del equipo externo, se distribuye paralelamente. Esto funciona perfectamente para las consultas con registros amplios, pues la consulta solicitada

requiere buscar en 4 diferentes columnas; al ser distribuido paralelamente esta consulta, mejora el tiempo de espera. Por lo tanto, al ver el plan de ejecución del equipo externo, se puede concluir que es mejor que el nuestro.

CONSULTA 5

En este caso particular en ambos equipos, el nodo final cuenta con un 100% y 99%, siendo el externo y nuestro, respectivamente, sin embargo, al ser planes de ejecución, son estimaciones, así que, al ser valores aproximados, se puede considerar el mismo porcentaje. Dentro del plan de ejecución externo, la consulta es plana y directa con 6 nodos, mientras la nuestra, cuenta con 11 nodos, con una pequeña ramificación. Aunque se cuenten mayores nodos en nuestra consulta, al ser la mayor cantidad en el nodo final en ambas consultas, se podría resumir, que ambas consultas cuentan con el mismo tiempo de procesamiento, pero no la misma cantidad de pasos.

CONCLUSIONES

La creación de índices es un proceso el cual se requiere poder tener una diferente manera de poder ordenar los datos de una tabla, esto con el fin de mejorar la realización de las consultas necesarias para poder encontrar los datos exigidos de una manera más eficiente conforme al sistema.

Dentro de esta práctica, al momento de usar las diferentes consultas con la base de datos de AdventureWorks, se analizó que los índices más correctos para las consultas fueron: ProductId, OrderQty como un primer índice, luego un índice para ProductSubcategoryId y un tercer índice para ProductCategoryId; esto debido que en la primera parte de la consulta se debe mostrar la información del producto y la cantidad, mientras que el resto de índices para operaciones. Así mismo en la segunda consulta se requería un índice para la información de los clientes y 3 índices para operaciones siendo TerritoryId, PersonId, BusinessEntityId. Finalmente, en la tercera consulta los índices usados fueron SalesOrderId y ProductId, porque se requiere mostrar la información del producto y se debe buscar también el Id de la orden específica.

En la segunda parte, los índices para las consultas necesarias en la base de datos de CovidHistorico, se usaron non-clustered index, por el hecho de que las consultas se hicieron en la misma tabla, así que, si se usará clustered index, cambiaría la organización completa de la tabla, desaprovechando las columnas del resto de consultas. En la consulta 3, los índices ocupados fueron para las columnas DIABETES, OBESIDAD e HIPERTENSIÓN, ya que son las columnas que se requieren hacer operaciones. Para la siguiente consulta solo se agregó a los índices creados, un índice de la columna TABAQUISMO. Para la última consulta se usó FECHA_DEF y NEUMONIA, puesto como anteriormente son las columnas a las que se requieren hacer operaciones.

Gracias a todo esto vemos la comparativa de nuestros índices propuestos logrando la creación y uso adecuado de estos para permitir comparar y evaluar diferentes planes de ejecución para una misma consulta, haciendo un mejor uso del rendimiento por lo que podemos reducir recursos de nuestra base de datos que están escaneando las tablas completas (table scan) en alguna otra operación que sea más rápida y que consuma diferente carga por el plan de ejecución necesario, estas mejoras reducen el uso de recursos del sistema, como CPU y memoria, al evitar cargas innecesarias en el motor de base de datos.