

# 1. Listar los casos positivos por entidad de residencia

## versión 1

```
select * into Consulta1 from datoscovid
```

```
alter table Consulta1 alter column ENTIDAD_RES nvarchar(15) not null  
create clustered index CIX_ENTIDAD_RES on Consulta1 (ENTIDAD_RES)
```

```
select ENTIDAD_RES, count(*) total_confirmado  
from Consulta1  
where CLASIFICACION_FINAL between 1 and 3  
group by ENTIDAD_RES
```

## version 2

```
select ENTIDAD_RES, count(*) total_confirmado  
from dbo.datoscovid  
where CLASIFICACION_FINAL between 1 and 3  
group by ENTIDAD_RES  
order by ENTIDAD_RES
```

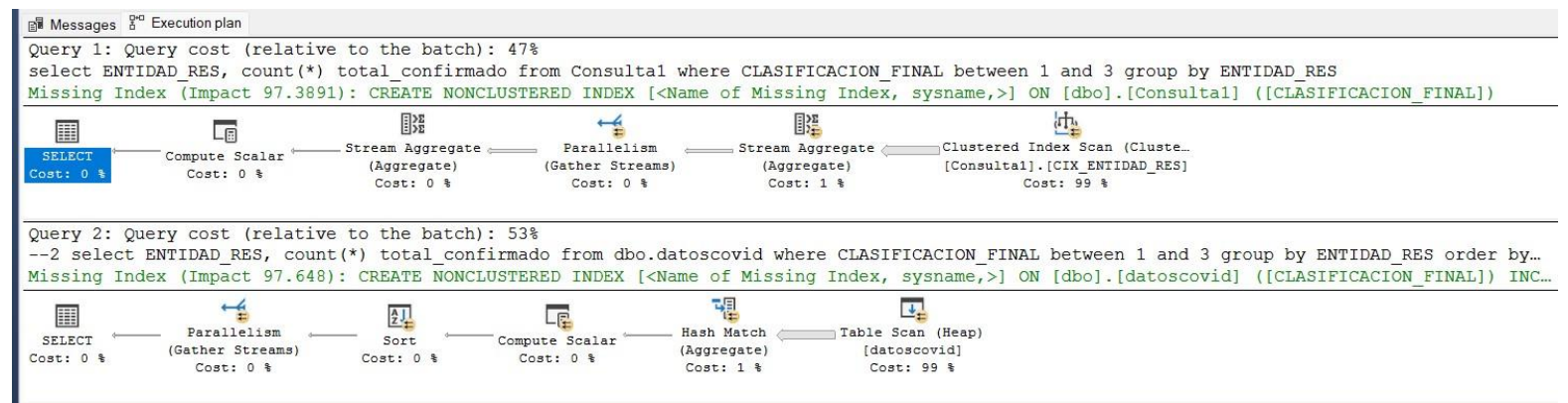
145 %

Results Messages

	ENTIDAD_RES	total_confirmado
1	17	66703
2	01	61705
3	10	67737
4	21	167075
5	22	140248
6	12	97719
7	03	101286
8	18	57733
9	06	54225
10	11	277113
11	14	234286
12	32	67332
13	19	309540
14	26	162567
15	27	189393
16	31	108481
17	02	120104

Query executed successfully.

SYKES (15.0 RTM) | SYKES\chenc (57) | covidHistorico | 00:01:35 | 32 rows



## 2. Listar los casos sospechosos por entidad

### Versión 1

```
select ENTIDAD_UM, ENTIDAD_RES, count(*) total_sospechosos
from dbo.datoscovid
where CLASIFICACION_FINAL = 6
group by ENTIDAD_UM, ENTIDAD_RES
order by ENTIDAD_UM
```

### versión 2

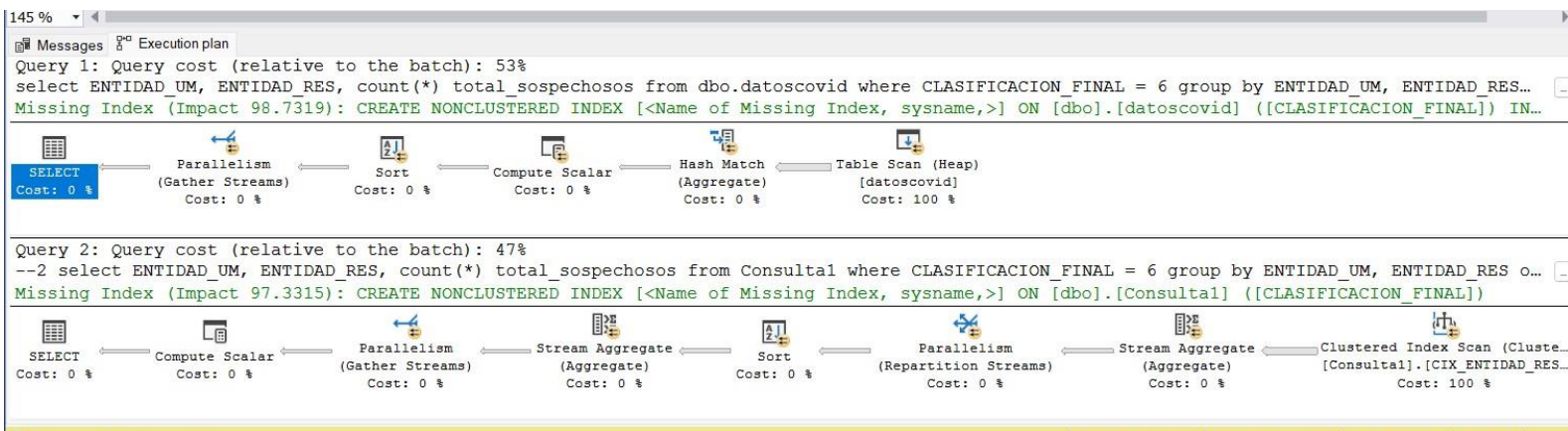
```
alter table Consulta1 alter column ENTIDAD_UM nvarchar(15) not null
drop index CIX_ENTIDAD_RES on Consulta1
create clustered index CIX_ENTIDAD_RES_ENTIDAD_UM on Consulta1
(ENTIDAD_RES, ENTIDAD_UM)
```

```
select ENTIDAD_UM, ENTIDAD_RES, count(*) total_sospechosos
from Consulta1
where CLASIFICACION_FINAL = 6
group by ENTIDAD_UM, ENTIDAD_RES
order by ENTIDAD_UM
```

145 %

Results		Messages	
	ENTIDAD_UM	ENTIDAD_RES	total_sospechosos
1	01	22	1
2	01	09	4
3	01	16	2
4	01	23	1
5	01	11	3
6	01	03	1
7	01	12	1
8	01	32	39
9	01	17	1
10	01	14	34
11	01	01	10753
12	01	30	1
13	01	24	1
14	01	28	1
15	01	19	1
16	01	26	1
17	01	05	1

✓ Query executed successfully.



3. Listar el top 5 de municipios por entidad con el mayor número de casos reportados, indicando casos sospechosos y casos confirmados.

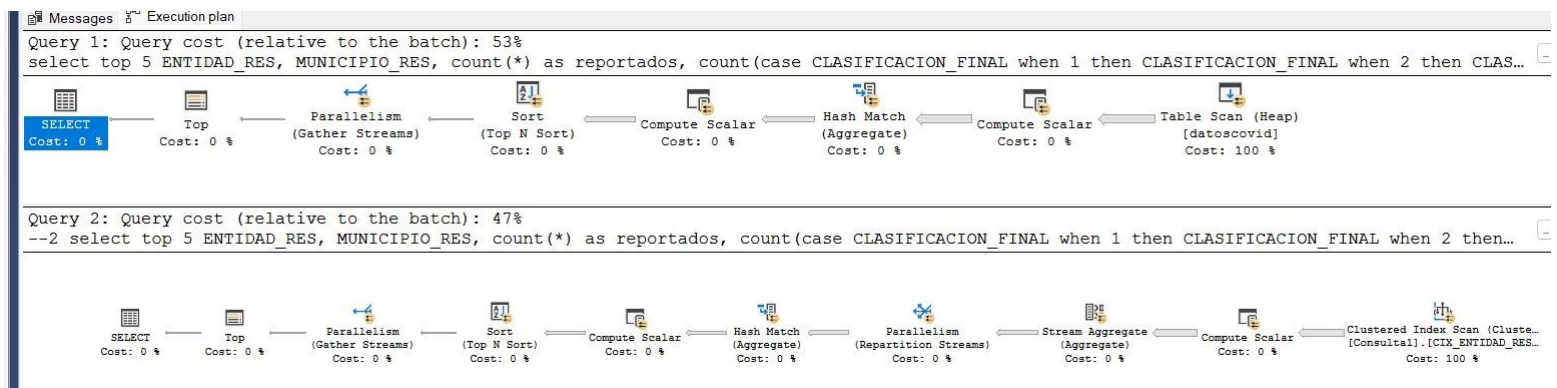
#### Versión 1

```
select top 5 ENTIDAD_RES, MUNICIPIO_RES, count(*) as reportados, count(case
CLASIFICACION_FINAL when 1 then CLASIFICACION_FINAL
                                when 2 then
CLASIFICACION_FINAL
                                when 3 then CLASIFICACION_FINAL
                                end) as confirmado,
count(case CLASIFICACION_FINAL when 6 then CLASIFICACION_FINAL end) as
sospechoso
from dbo.datoscovid
group by ENTIDAD_RES, MUNICIPIO_RES
order by reportados desc
```

#### versión 2

```
drop index CIX_ENTIDAD_RES_ENTIDAD_UM on Consulta1
create clustered index CIX_ENTIDAD_RES_MUNICIPIO_RES on Consulta1
(ENTIDAD_RES,MUNICIPIO_RES)
select top 5 ENTIDAD_RES, MUNICIPIO_RES, count(*) as reportados, count(case
CLASIFICACION_FINAL when 1 then CLASIFICACION_FINAL
                                when 2 then
CLASIFICACION_FINAL
                                when 3 then CLASIFICACION_FINAL
                                end) as confirmado,
count(case CLASIFICACION_FINAL when 6 then CLASIFICACION_FINAL end) as
sospechoso
from Consulta1
group by ENTIDAD_RES,MUNICIPIO_RES
order by reportados desc
```

	ENTIDAD_RES	MUNICIPIO_RES	reportados	confirmado	sospechoso
1	09	007	822536	204101	12124
2	09	005	612632	167693	14836
3	09	010	477655	185653	10158
4	09	012	363363	132823	4798
5	09	015	305972	75790	6434



#### 4. Determinar el municipio con el mayor número de defunciones en casos confirmados.

##### Versión 1

```
create view Defuciones as
select * from datoscovid
where FECHA_DEF != '9999-99-99'

select top 1 MUNICIPIO_RES, count(case CLASIFICACION_FINAL when 1 then
CLASIFICACION_FINAL
                                when 2 then CLASIFICACION_FINAL
                                when 3 then CLASIFICACION_FINAL end)
as DefucionesConfirmados
from Defuciones
group by MUNICIPIO_RES
order by DefucionesConfirmados desc
```

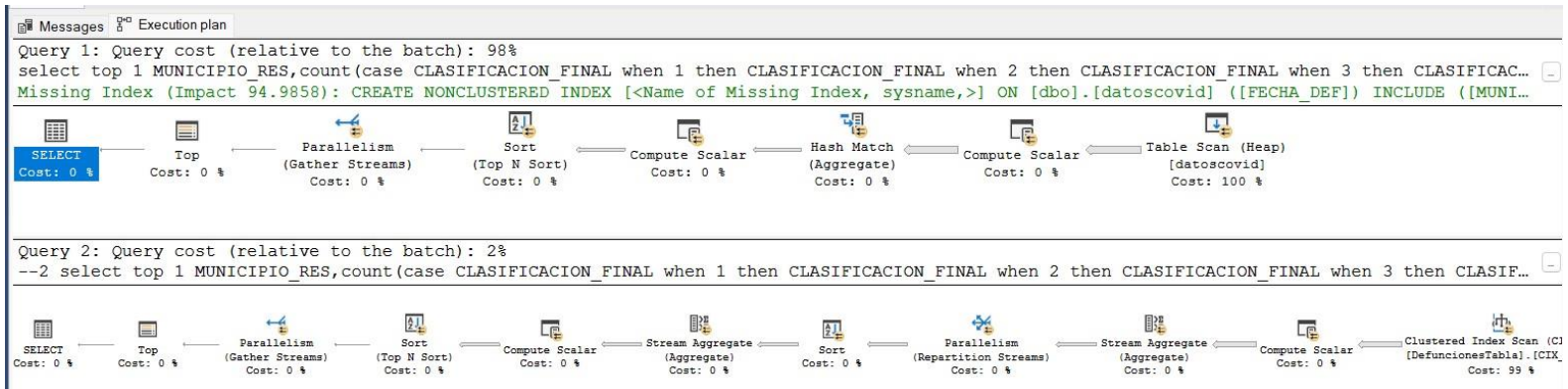
##### versión 2

```
select * into DefucionesTabla from Defuciones
create clustered index CIX_Defuciones on DefucionesTabla (MUNICIPIO_RES)

select top 1 MUNICIPIO_RES, count(case CLASIFICACION_FINAL when 1 then
CLASIFICACION_FINAL
                                when 2 then CLASIFICACION_FINAL
                                when 3 then CLASIFICACION_FINAL end)
as DefucionesConfirmados
from DefucionesTabla
group by MUNICIPIO_RES
```

```
order by DefuncionesConfirmados desc
```

	MUNICIPIO_RES	DefuncionesConfirmados
1	039	14256



5. Determinar por entidad, si de casos sospechosos hay defunciones reportadas asociadas a neumonía.

# Versión 1

```
select * into DefuncionesNeumonia
from Defuciones
where NEUMONIA = 1
```

```
select * into DefuncionesNeumonia2
from Defunciones
where NEUMONIA = 1
```

```
select ENTIDAD_UM,ENTIDAD_RES,count(case CLASIFICACION_FINAL when 6 then
CLASIFICACION_FINAL end)
```

as

DefuncionesNeumoniaSospechosos

from

```
DefuncionesNeumonia
group by ENTIDAD_UM, ENTIDAD_RES
order by ENTIDAD_UM, ENTIDAD_RES
```

versión 2

```
create clustered index CIX_ENTIDAD_UM_ENTIDAD_RES on DefuncionesNeumonia
(ENTIDAD_UM, ENTIDAD_RES)
```

```
select ENTIDAD_UM,ENTIDAD_RES,count(case CLASIFICACION_FINAL when 6 then
CLASIFICACION_FINAL end)
```

as

DefuncionesNeumoniaSospechosos

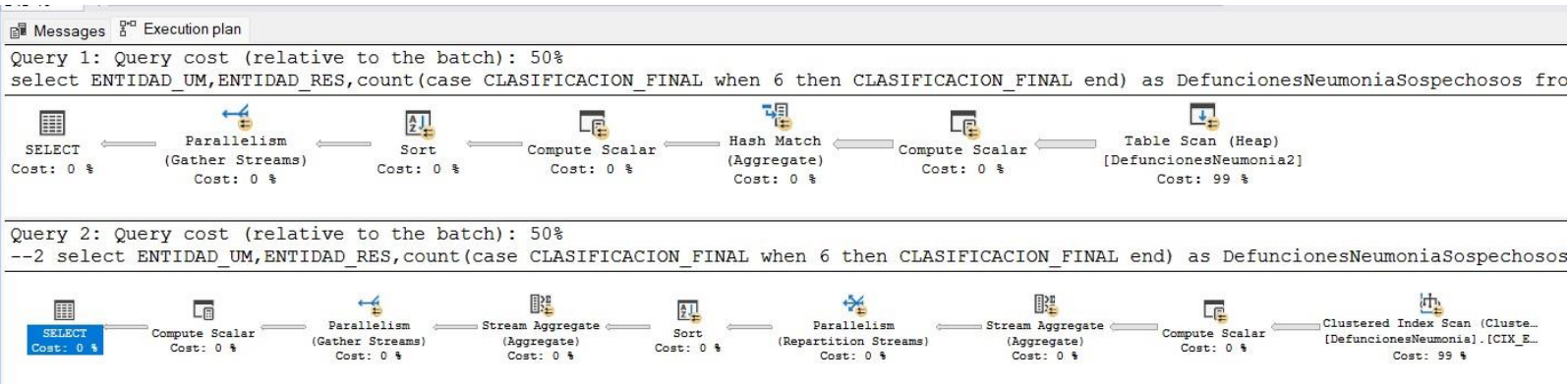
from

```
DefuncionesNeumonia
group by ENTIDAD UM, ENTIDAD RES
```

order by ENTIDAD\_UM, ENTIDAD\_RES

	ENTIDAD_UM	ENTIDAD_RES	DefuncionesNeumoniaSospechosos
1	01	01	34
2	01	09	0
3	01	11	0
4	01	14	2
5	01	15	1
6	01	18	0
7	01	21	0
8	01	24	0
9	01	32	1
10	02	02	224
11	02	03	0
12	02	05	0
13	02	09	0
14	02	11	0
15	02	12	0
16	02	14	0
17	02	15	0
18	02	16	0
19	02	17	0

Query executed successfully.



6. Listar por entidad el total de casos sospechosos, casos confirmados, total de defunciones en los meses de marzo a agosto 2020 y de diciembre 2020 a mayo 2021.

## Versión 1

```
create view Sospechosos as
select ENTIDAD_UM,ENTIDAD_RES,FECHA_INGRESO,
count(case CLASIFICACION_FINAL when 6 then CLASIFICACION_FINAL end) as Sospechoso
from datoscovid
where (FECHA_INGRESO between '2020-03-01' and '2020-08-31') or
(FECHA_INGRESO between '2020-12-01' and '2021-05-31' )
group by ENTIDAD_UM,ENTIDAD_RES,FECHA_INGRESO
```

```
create view Confirmados as
select ENTIDAD_UM,ENTIDAD_RES,FECHA_INGRESO,
count(case CLASIFICACION_FINAL when 1 then CLASIFICACION_FINAL
when 2 then CLASIFICACION_FINAL
```

```

                                when 3 then CLASIFICACION_FINAL end)
as Confirmados
from datoscovid
where (FECHA_INGRESO between '2020-03-01' and '2020-08-31') or
(FECHA_INGRESO between '2020-12-01' and '2021-05-31' )
group by ENTIDAD_UM,ENTIDAD_RES,FECHA_INGRESO

```

```

create view Defunciones2 as
select ENTIDAD_UM,ENTIDAD_RES,FECHA_INGRESO, count(*) as Defuncioness from
Defunciones
where (FECHA_INGRESO between '2020-03-01' and '2020-08-31') or
(FECHA_INGRESO between '2020-12-01' and '2021-05-31' )
group by ENTIDAD_UM,ENTIDAD_RES,FECHA_INGRESO

```

```

select
c.ENTIDAD_UM,c.ENTIDAD_RES,c.FECHA_INGRESO,s.Sospechoso,c.Confirmados,d2.Defunciones
s
from Confirmados c
inner join Defunciones2 d2
on c.FECHA_INGRESO = d2.FECHA_INGRESO and c.ENTIDAD_UM = d2.ENTIDAD_UM and
c.ENTIDAD_RES = d2.ENTIDAD_RES
inner join Sospechosos s
on s.FECHA_INGRESO = d2.FECHA_INGRESO and s.ENTIDAD_UM = d2.ENTIDAD_UM and
s.ENTIDAD_RES = d2.ENTIDAD_RES
order by c.ENTIDAD_UM, c.ENTIDAD_RES,c.FECHA_INGRESO

```

## versión 2

```

select * into SospechososTabla from Sospechosos
select * into ConfirmadosTabla from Confirmados
select * into Defunciones2Tabla from Defunciones2

```

```

create clustered index CIX_SospechososTabla on SospechososTabla
(ENTIDAD_UM,ENTIDAD_RES)
create clustered index CIX_ConfirmadosTabla on ConfirmadosTabla
(ENTIDAD_UM,ENTIDAD_RES)
create clustered index CIX_Defunciones2Tabla on Defunciones2Tabla
(ENTIDAD_UM,ENTIDAD_RES)

```

```

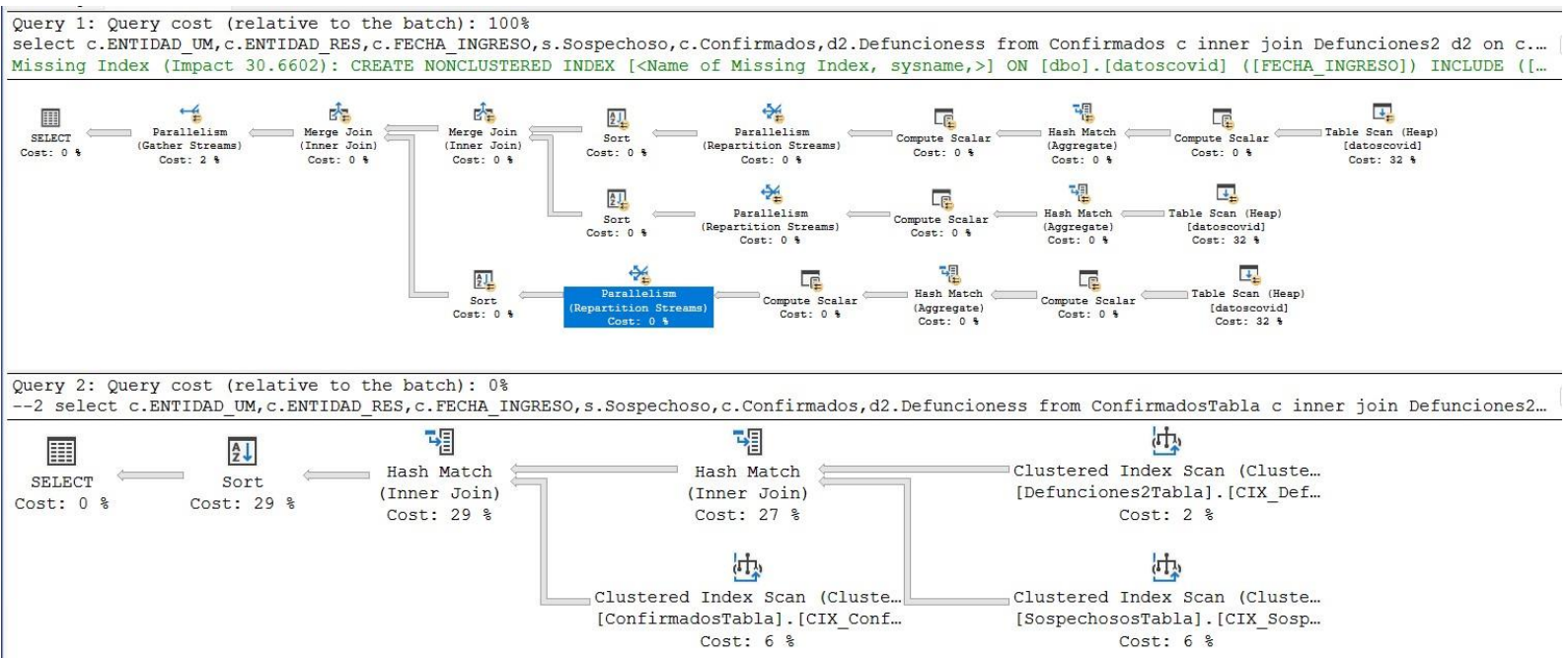
select
c.ENTIDAD_UM,c.ENTIDAD_RES,c.FECHA_INGRESO,s.Sospechoso,c.Confirmados,d2.Defunciones
s
from ConfirmadosTabla c
inner join Defunciones2Tabla d2
on c.FECHA_INGRESO = d2.FECHA_INGRESO and c.ENTIDAD_UM = d2.ENTIDAD_UM and
c.ENTIDAD_RES = d2.ENTIDAD_RES
inner join SospechososTabla s
on s.FECHA_INGRESO = d2.FECHA_INGRESO and s.ENTIDAD_UM = d2.ENTIDAD_UM and
s.ENTIDAD_RES = d2.ENTIDAD_RES
order by c.ENTIDAD_UM, c.ENTIDAD_RES,c.FECHA_INGRESO

```



Results Messages						
ENTIDAD_UM	ENTIDAD_RES	FECHA_INGRESO	Sospechoso	Confirmados	Defuncioness	
01	01	2020-03-02	5	0	1	
01	01	2020-03-12	7	0	1	
01	01	2020-03-18	13	3	1	
01	01	2020-03-22	0	1	1	
01	01	2020-03-25	6	17	2	
01	01	2020-03-29	1	3	1	
01	01	2020-03-31	28	1	1	
01	01	2020-04-01	20	0	1	
01	01	2020-04-03	21	2	1	
01	01	2020-04-11	3	1	1	
01	01	2020-04-15	9	12	2	
01	01	2020-04-16	10	7	1	
01	01	2020-04-18	3	14	4	
01	01	2020-04-20	11	17	2	
01	01	2020-04-21	16	13	1	
01	01	2020-04-23	9	21	2	
01	01	2020-04-24	14	11	2	

Query executed successfully.





7. Listar los 5 municipios con el mayor número de casos confirmados en niños menos de 13 años con alguna comorbilidad reportada y cuantos de esos casos fallecieron.

Versión 1

```
create view Enfermedades as
select MUNICIPIO_RES, EDAD, CLASIFICACION_FINAL, FECHA_DEF, case when NEUMONIA != 1
THEN 0 else 1
END AS Neumonia, case when DIABETES != 1 THEN 0 else 1
END AS Diabetes, case when EPOC != 1 THEN 0 else 1
END AS Epoc, case when ASMA != 1 THEN 0 else 1
END AS Asma, case when INMUSUPR != 1 THEN 0 else 1
END AS Inmusupr, case when HIPERTENSION != 1 THEN 0 else 1
END AS Hipertension, case when OTRA_COM != 1 THEN 0 else 1
END AS OtrasEnfermedades, case when CARDIOVASCULAR != 1 THEN 0 else 1
END AS Cardiovascular, case when OBESIDAD != 1 THEN 0 else 1
END AS Obesidad, case when RENAL_CRONICA != 1 THEN 0 else 1
END AS RenalCronica, case when TABAQUISMO != 1 THEN 0 else 1
END AS Tabaquismo, case when OTRO_CASO != 1 THEN 0 else 1
END AS OtroCaso from Consula7
where EDAD <13

create view Comorbilidad2 as
select MUNICIPIO_RES, EDAD, count( case CLASIFICACION_FINAL when 1 then
CLASIFICACION_FINAL
                                when 2 then CLASIFICACION_FINAL
                                when 3 then CLASIFICACION_FINAL
end) as Confirmados,
count(FECHA_DEF) as Defunciones,
(Neumonia+Diabetes+Epoc+Asma+Inmusupr+Hipertension+OtrasEnfermedades+Cardiovascular+
Obesidad
+RenalCronica+Tabaquismo+OtroCaso) as Comorbilidad from Enfermedades
where FECHA_DEF != '9999-99-99'
group by MUNICIPIO_RES, EDAD, Neumonia, Diabetes, Epoc, Asma, Inmusupr, Hipertension,
OtrasEnfermedades, Cardiovascular, Obesidad, RenalCronica, Tabaquismo, OtroCaso

create view Comorbilidad3 as
select * from Comorbilidad2
where Comorbilidad >=2

select top 5 MUNICIPIO_RES, EDAD, Confirmados, Defunciones, count(Comorbilidad) as
ComorbilidadTotal
from Comorbilidad3 where EDAD != 0
group by MUNICIPIO_RES, EDAD, Confirmados, Defunciones
order by Confirmados desc
```

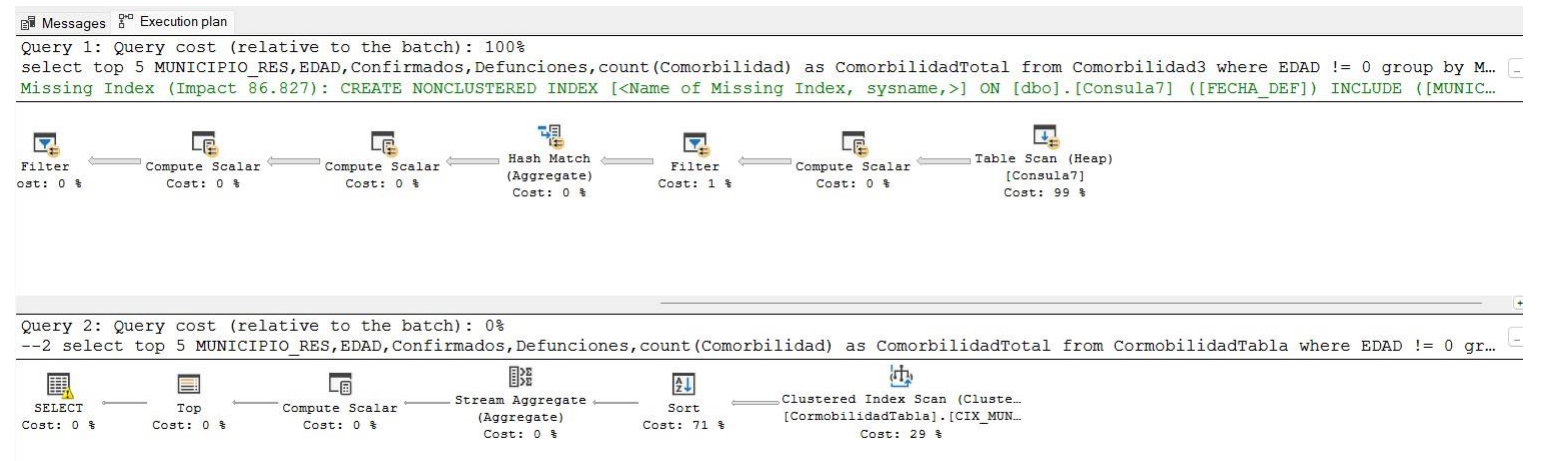
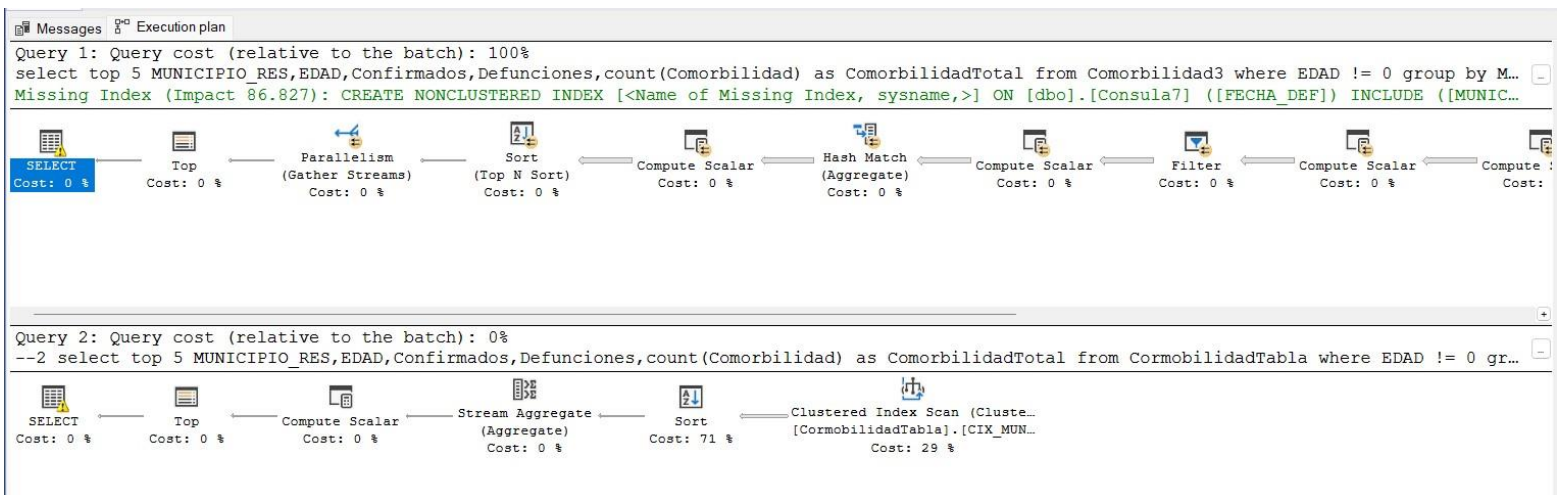
## versión 2

```
select * into CormobilidadTabla from Comorbilidad3
```

```
create clustered index CIX_MUNICIPIO_RES on CormobilidadTabla (MUNICIPIO_RES)
```

```
select top 5 MUNICIPIO_RES,EDAD,Confirmados,Defunciones,count(Comorbilidad) as ComorbilidadTotal  
from CormobilidadTabla where EDAD != 0  
group by MUNICIPIO_RES,EDAD,Confirmados,Defunciones  
order by Confirmados desc
```

	MUNICIPIO_RES	EDAD	Confirmados	Defunciones	ComorbilidadTotal
1	007	1	3	4	1
2	001	1	2	2	1
3	003	1	2	2	1
4	001	1	1	1	1
5	001	10	1	1	1



8. Determinar si en el año 2020 hay una mayor cantidad de defunciones menores de edad que en el año 2021 y 2022.

Version 1

```
select sq1.*,
      (case when (defME2020>defME2021 and defME2020>defME2022)
        then 'En 2020 hubo MÁS defunciones menores de edad respecto a
2021 y 2022'
        else 'En 2020 hubo MENOS defunciones menores de edad respecto a
2021 y 2022' end) as BOOL2020GT20212022
      from (
        select
FECHA_DEF end) as defME2020,
        count(case when FECHA_DEF LIKE '2020%' then
as defME2021,
        count(case when FECHA_DEF LIKE '2021%' then 1 end)
        count(case when FECHA_DEF LIKE '2022%' then 1 end
) as defME2022
        from dbo.datoscovid where EDAD <18
)as sq1
```

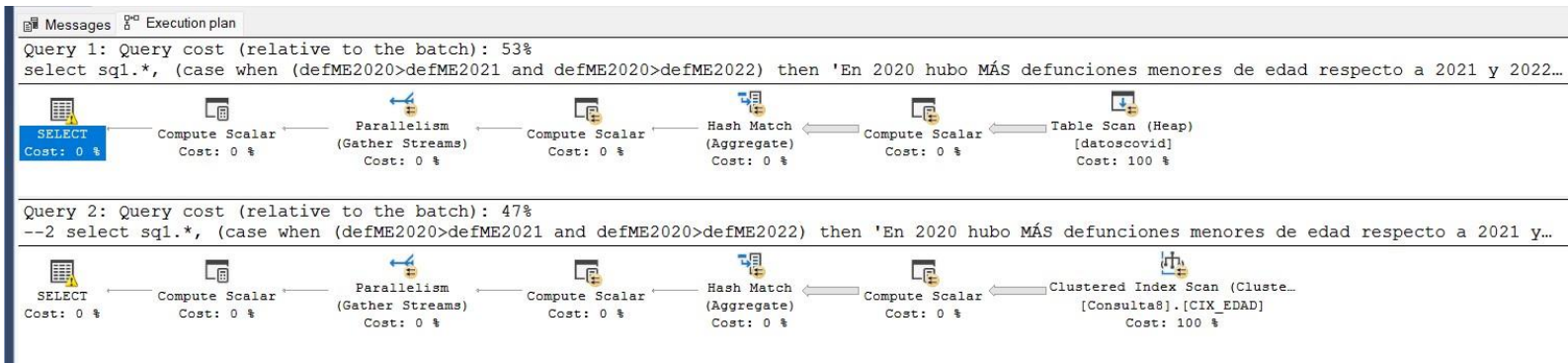
Versión 2

```
select * into Consulta8 from datoscovid

create clustered index CIX_EDAD on Consulta8 (EDAD)

select sq1.*,
      (case when (defME2020>defME2021 and defME2020>defME2022)
        then 'En 2020 hubo MÁS defunciones menores de edad respecto a
2021 y 2022'
        else 'En 2020 hubo MENOS defunciones menores de edad respecto a
2021 y 2022' end) as BOOL2020GT20212022
      from (
        select
FECHA_DEF end) as defME2020,
        count(case when FECHA_DEF LIKE '2020%' then
as defME2021,
        count(case when FECHA_DEF LIKE '2021%' then 1 end)
        count(case when FECHA_DEF LIKE '2022%' then 1 end
) as defME2022
        from Consulta8 where EDAD <18
)as sq1
```

Results Messages				
	defME2020	defME2021	defME2022	BOOL2020GT20212022
1	1721	1663	342	En 2020 hubo MÁS defunciones menores de edad respecto a 2021 y 2022



9. Determinar si en el año 2021 hay un porcentaje mayor al 60 de casos reportados que son confirmados por estudios de laboratorio en comparación al año 2020.

Versión 1

```
select (case when (select count(*)*1.0/(select count(*)
from dbo.datoscovid where FECHA_INGRESO LIKE '2020%' and CLASIFICACION_FINAL = 3)
from dbo.datoscovid where FECHA_INGRESO LIKE '2021%' and CLASIFICACION_FINAL =
3)>1.6
then 'Creció mas de 60% respecto 2020' else 'Creció menos de 60% respecto 2020' end)
as Resultado
```

```
select
(
case when (select count(*)* 1.0 /(select count(*)
from dbo.datoscovid
where FECHA_INGRESO LIKE '2020%'
and CLASIFICACION_FINAL = 3)
from dbo.datoscovid
where FECHA_INGRESO LIKE '2021%'
and CLASIFICACION_FINAL = 3)> 1.6
then 'Creció mas de 60% respecto 2020' else 'Creció menos de 60% respecto
2020' end) as R
```

versión 2

```
drop index CIX_EDAD on Consulta8
```

```
create clustered index CIX_CLASIFICACION_FINAL on Consulta8 (CLASIFICACION_FINAL)
```

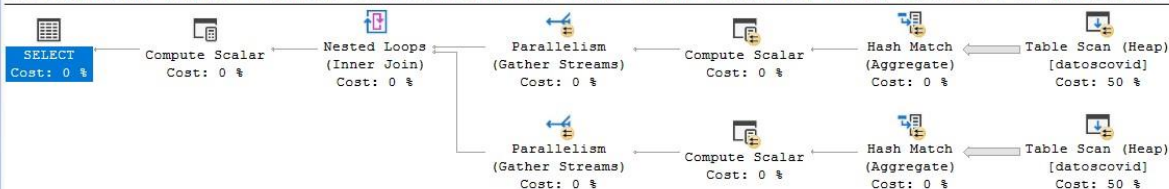
```
select
(
case when (select count(*)* 1.0 /(select count(*)
from dbo.datoscovid
where FECHA_INGRESO LIKE '2020%'
and CLASIFICACION_FINAL = 3)
from dbo.datoscovid
where FECHA_INGRESO LIKE '2021%'
and CLASIFICACION_FINAL = 3)> 1.6
then 'Creció mas de 60% respecto 2020' else 'Creció menos de 60% respecto
2020' end) as R
```

	R
1	Creció mas de 60% respecto 2020

Query 1: Query cost (relative to the batch): 77%

```
select ( case when (select count(*)* 1.0 /(select count(*) from dbo.datoscovid where FECHA_INGRESO LIKE '2020%' and CLASIFICACION_FINAL = 3) from...
```

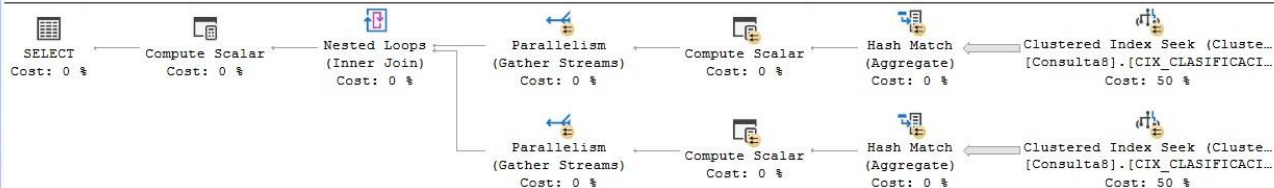
Missing Index (Impact 49.2495): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [dbo].[datoscovid] ([CLASIFICACION\_FINAL],[FECH...



Query 2: Query cost (relative to the batch): 23%

```
--2 select ( case when (select count(*)* 1.0 /(select count(*) from Consulta8 where FECHA_INGRESO LIKE '2020%' and CLASIFICACION_FINAL = 3) from...
```

Missing Index (Impact 47.5465): CREATE NONCLUSTERED INDEX [<Name of Missing Index, sysname,>] ON [dbo].[Consulta8] ([CLASIFICACION\_FINAL],[FECH...



10. Determinar en qué rango de edad: menor de edad, 19 a 40, 40 a 60 o mayor de 60 hay más casos reportados que se hayan recuperado.

Versión 1

```
select (case
  when ((RecME>Rec1940) and (RecME>Rec4060) and (RecME>Rec60M)) then 'Los que
mas se recuperaron fueron los Menores de edad'
  when ((Rec1940>RecME) and (Rec1940>Rec4060) and (Rec1940>Rec60M)) then 'Los
que mas se recuperaron fueron personas entre 19 y 40 años'
  when ((Rec4060>RecME) and (Rec4060>Rec1940) and (Rec4060>Rec60M)) then 'Los
que mas se recuperaron fueron personas entre 19 y 40 años'
  ELSE 'Los que mas se recuperaron fueron personas Mayores de 60 años' end) as
Mejorcategoría from (
  select count(case when EDAD <18 then FECHA_DEF end) as RecME,
  count(case when EDAD BETWEEN 19 and 40 then 1 end) as Rec1940,
  count(case when EDAD BETWEEN 40 and 60 then 1 end ) as Rec4060,
  count(case when EDAD >60 then 1 end ) as Rec60M
  from datoscovid where FECHA_DEF = '9999-99-99') as sq1
```

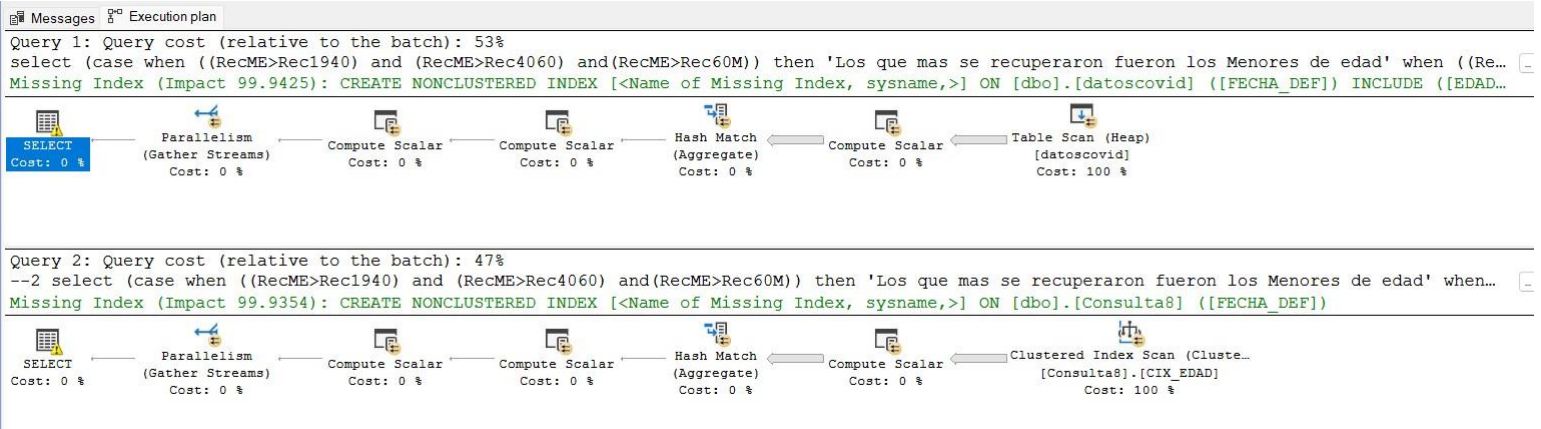
## versión 2

```
drop index CIX_CLASIFICACION_FINAL on Consulta8
```

```
create clustered index CIX_EDAD on Consulta8 (EDAD)
```

```
select (case
    when ((RecME>Rec1940) and (RecME>Rec4060) and(RecME>Rec60M)) then 'Los que
mas se recuperaron fueron los Menores de edad'
    when ((Rec1940>RecME) and (Rec1940>Rec4060) and(Rec1940>Rec60M)) then 'Los
que mas se recuperaron fueron personas entre 19 y 40 años'
    when ((Rec4060>RecME) and (Rec4060>Rec1940) and(Rec4060>Rec60M)) then 'Los
que mas se recuperaron fueron personas entre 19 y 40 años'
    ELSE 'Los que mas se recuperaron fueron personas Mayores de 60 años' end) as
Mejorcategoria from (
    select count(case when EDAD <18 then FECHA_DEF end) as RecME,
    count(case when EDAD BETWEEN 19 and 40 then 1 end) as Rec1940,
    count(case when EDAD BETWEEN 40 and 60 then 1 end ) as Rec4060,
    count(case when EDAD >60 then 1 end ) as Rec60M
    from Consulta8 where FECHA_DEF = '9999-99-99') as sq1
```

Mejorcategoria	
1	Los que mas se recuperaron fueron personas entre 19 y 40 años





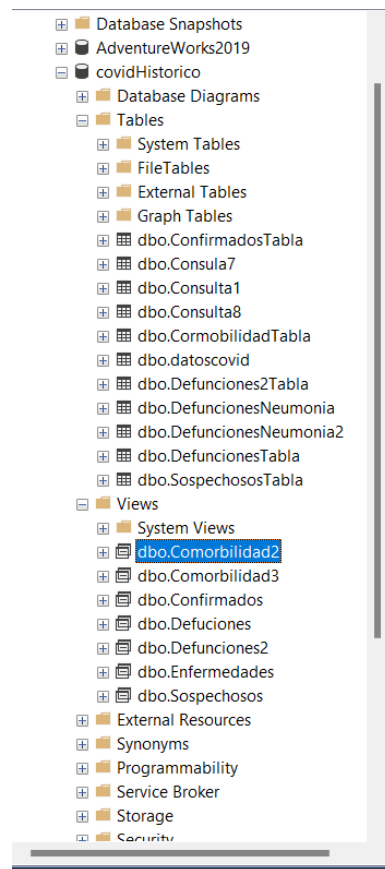
## Configuración inicial de la Base de Datos Covid

Al total realizamos 10 tablas copias, estas las utilizamos para generar los índices agrupados para las consultas.

En la tabla Confirmados solamente están las personas que han sido confirmados de covid, las tablas consultan 7,1,8 son copias de la tabla datos covid, con la única diferencia es que en estas agregamos los índices agrupados. Para la tabla comorbilidad, solamente cuenta con los pacientes que cuentan con comorbilidad.

En las tablas defunciones son los todos los pacientes que fallecieron. Y en las tablas defunciones neumonía, son los pacientes que fallecieron y que presentaron neumonía. Todas las tablas se utilizaron para analizar la versión de las consultas con índices agrupados.

En las vistas, nada más fueron utilizadas para poder hacer las consultas de los ejercicios sin índices agrupados, solamente contienen los datos que en su nombre representa, por ejemplo, la vista comorbilidad tiene a los pacientes que tengan comorbilidad.



## Análisis

### Consulta 1

Nosotros hicimos el índice agrupado sobre la columna ENTIDAD\_RES, con esto obtuvimos un 3% de mejoría en el plan de ejecución, pero analizamos que era mejor que el índice agrupado fuera sobre las columnas CLASIFICACION\_FINAL ya que la consulta hace la búsqueda sobre este atributo.

### Consulta 2

Hicimos un índice agrupado compuesto sobre las columnas ENTIDAD\_RES y ENTIDAD\_NUM, y con esto obtuvimos un 3% de mejoría en el plan de ejecución. Pero analizamos que era mejor opciones hacer el índice agrupado sobre la columna CLASIFICACION\_FINAL, porque es donde se hace la búsqueda.

### Consulta 3

Hicimos un índice agrupado compuesto sobre las columnas ENTIDAD\_RES y ENTIDAD\_NUM, y obtuvimos un 3% de mejoría en el plan de ejecución. Pero de la misma manera, hubiera estado mejor que estudiar en la columna CLASIFICACION\_FINAL.

### Consulta 4

El índice agrupado fue sobre la columna MUNICIPIO\_RES, con este índice obtuvimos un 98% de mejoría en el plan de ejecución, así que fue una muy buena opción el haber agregado el índice agrupado.

### Consulta 5

Hicimos un índice agrupado compuesto sobre las columnas ENTIDAD\_RES y ENTIDAD\_NUM, y obtuvimos un 0% de mejoría en el plan de ejecución. Creo que hubiera sido mucho mas optimizable el haber agregado el índice en CLASIFICACION\_FINAL.

### Consulta 6

Hicimos un índice agrupado compuesto sobre las columnas ENTIDAD\_RES y ENTIDAD\_NUM, sobre las 3 tablas que utilizamos para la consulta. Obtuvimos un 100% de optimización del plan de ejecución.

## Consulta 7

Hicimos un índice agrupado sobre la columna MUNICIPIO\_RES, con este índice obtuvimos un 100% de optimización del plan de ejecución.

## Consulta 8

Hicimos el índice agrupado sobre la columna EDAD, y con este índice obtuvimos una ligera mejoría del 3%.

## Consulta 9

Agregamos un índice agrupado en la CLASIFICACION\_FINAL, con esto obtuvimos una mejoría del 77%.

## Consulta 10

Agregamos el índice agrupado en la columna EDAD, pero obtuvimos nada mas un 3% de mejoría. El índice correcto era CLASIFICACION\_FINAL.

## Conclusiones

Es una base de datos algo complicado porque solamente se estructura con una sola tabla, así que SQL server se tardaba mucho en ejecutar las query, con respecto a los planes de ejecución , fue un poco complicado por la misma razón anterior, ya que cada vez que queríamos comprobar una plan nuevo se tardaba mucho en ejecutarlo, y eso nos llevaba mucho tiempo, de la misma manera las consultas eran algo laboriosas, pero creemos que con este ejercicio nos quedó más claro la importancia tomar

como herramientas los planes de ejecución para optimizar las consultas sabiendo agregar los índices y analizando la base datos.

En conclusión analizamos que la mejor opción de agregar un índice agrupado seria en la columna CLASIFICACION\_FINAL ya que la mayoría de las consultas se basan en la búsqueda sobre esta columna como también todas las columnas se relacionan con CLASIFICACION\_FINAL.