

Desafío - Manipulación de datos y transaccionalidad en las operaciones.

desarrollado por Marcelo Esparza Acuña

Descripción

Aplicando los conceptos y herramientas aprendidas hasta ahora, crea las queries SQL necesarias que contesten las siguientes preguntas.

****Preparando el entorno SQL ****

Creación de BD

```
create database desafio2_marcelo_esparza_143;
```

creación de tabla

```
CREATE TABLE IF NOT EXISTS INSCRITOS(  
    cantidad INT,  
    fecha DATE,  
    fuente VARCHAR  
);
```

insertando datos de prueba a la tabla

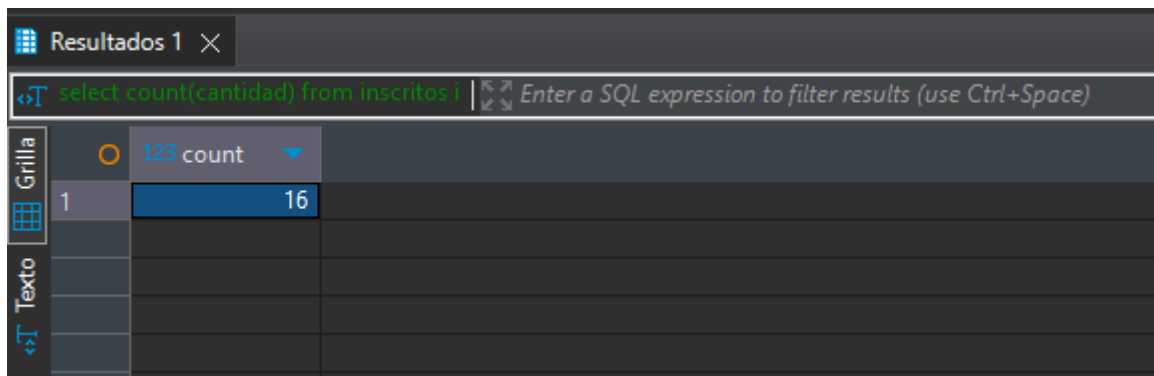
```
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 44, '01/01/2021', 'Blog'  
);  
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 56, '01/01/2021', 'Página'  
);  
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 39, '01/02/2021', 'Blog'  
);  
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 81, '01/02/2021', 'Página'  
);  
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 12, '01/03/2021', 'Blog'  
);  
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 91, '01/03/2021', 'Página'  
);  
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 48, '01/04/2021', 'Blog'  
);  
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 45, '01/04/2021', 'Página'  
);  
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 55, '01/05/2021', 'Blog'  
);
```

```
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 33, '01/05/2021', 'Página' );
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 18, '01/06/2021', 'Blog' );
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 12, '01/06/2021', 'Página' );
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 34, '01/07/2021', 'Blog' );
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 24, '01/07/2021', 'Página' );
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 83, '01/08/2021', 'Blog' );
INSERT INTO INSCRITOS(cantidad, fecha, fuente) VALUES ( 99, '01/08/2021', 'Página' );
```

Resolviendo desafio

1. ¿Cuántos registros hay?

```
select count(cantidad) from inscritos;
```



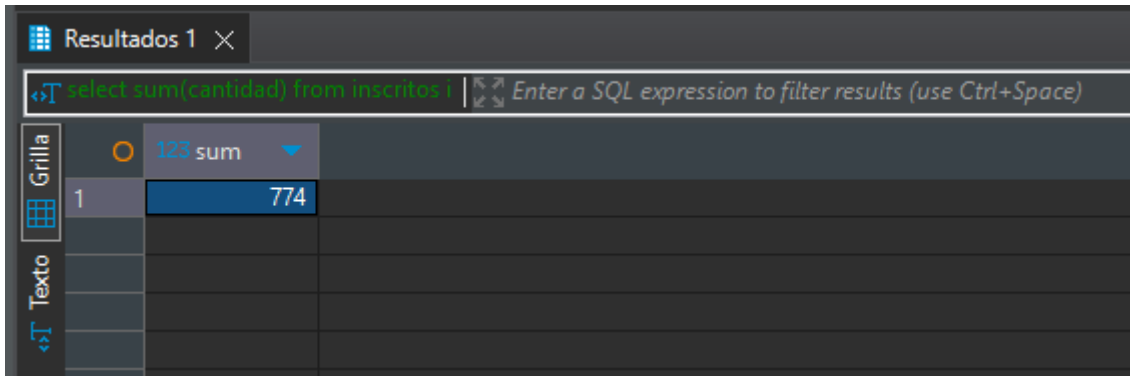
The screenshot shows a database interface with a query editor and a results grid. The query entered is `select count(cantidad) from inscritos;`. The results grid displays a single row with the value 16, representing the total number of records in the `inscritos` table.

Grilla	123 count
1	16

Comentario:* Para resolver este ejercicio, lo que se hizo fue contar la cantidad de registros existentes en la tabla **inscritos** utilizando la función **count()**

2. ¿Cuántos inscritos hay en total?

```
select sum(cantidad) from inscritos;
```

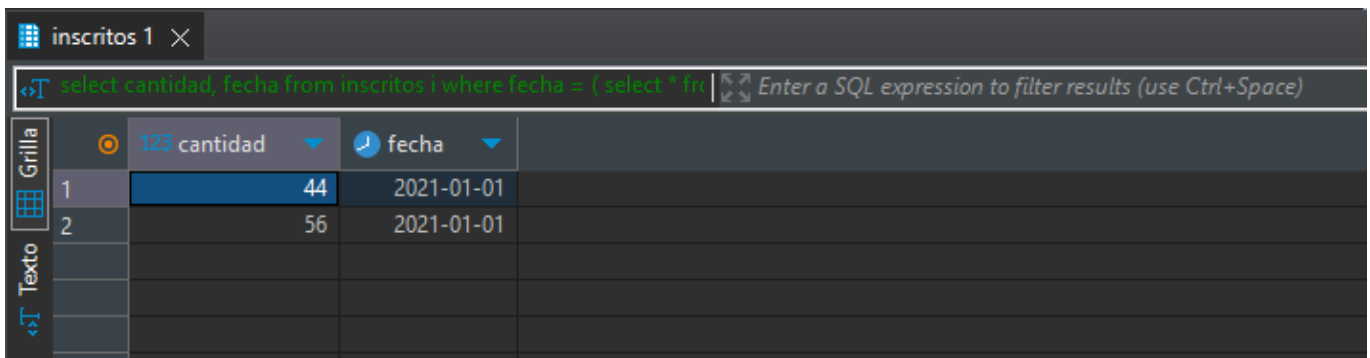


Resultados 1	
Enter a SQL expression to filter results (use Ctrl+Space)	
Grilla	123 sum
1	774
Texto	

Comentario: En este ejercicio lo que se buscaba era encontrar la sumatoria total de inscritos, para esto utilicé la función **sum()** en la consulta SQL, para sumar la cantidad total de inscritos.

3. ¿Cuál o cuáles son los registros de mayor antigüedad?

```
select
    cantidad, fecha
from inscritos
where
    fecha = (
        select * from (select min(fecha) from inscritos)
    );
```

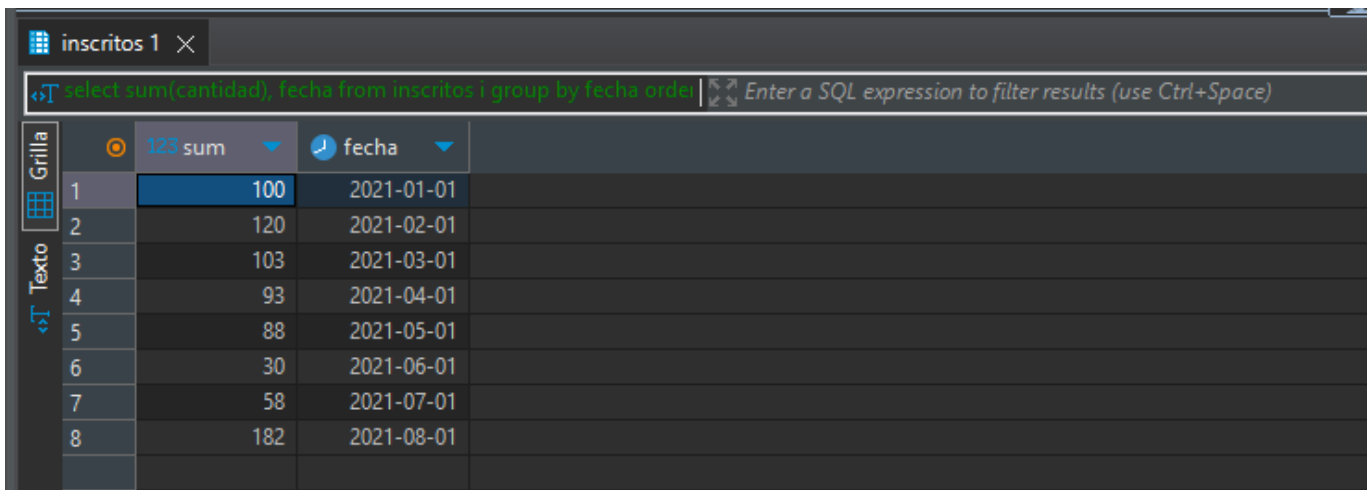


inscritos 1	
Enter a SQL expression to filter results (use Ctrl+Space)	
Grilla	123 cantidad fecha
1	44 2021-01-01
2	56 2021-01-01
Texto	

Comentario: Para obtener los registros de mayor antigüedad, lo que hice fue obtener la totalidad de la data desde la tabla **inscritos** y mediante una subconsulta filtré la columna fecha utilizando la función **min()**, para obtener la fecha mas antigua.

¿Cuántos inscritos hay por día? (entendiendo un día como una fecha distinta de ahora en adelante)

```
select
    sum(cantidad), fecha
from
    inscritos
group by
    fecha
order by
    fecha asc ;
```



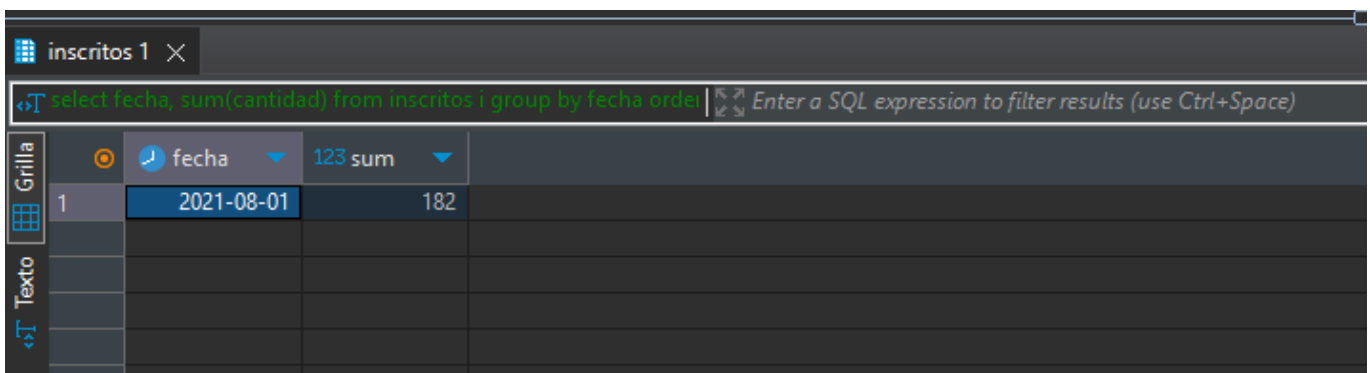
The screenshot shows a database interface with a tab labeled 'inscritos 1'. The SQL query entered is: `select sum(cantidad), fecha from inscritos i group by fecha order by`. The results are displayed in a table with two columns: 'sum' and 'fecha'.

	sum	fecha
1	100	2021-01-01
2	120	2021-02-01
3	103	2021-03-01
4	93	2021-04-01
5	88	2021-05-01
6	30	2021-06-01
7	58	2021-07-01
8	182	2021-08-01

Comentario: Para resolver este ejercicio, la consulta sql suma la cantidad de usuarios a la vez que agrupa el resultado por fechas, con esta consulta obtengo un listado de cantidades agrupadas por fecha sin repetición, finalmente este resultado lo ordeno por fecha de forma ascendente para tener un mejor orden en la visualización.

5. ¿Qué día se inscribieron la mayor cantidad de personas y cuántas personas se inscribieron en ese día?

```
select
    fecha, sum(cantidad)
from
    inscritos i
group by
    fecha
order by
    sum(cantidad) desc
limit 1;
```



The screenshot shows the same database interface with the SQL query: `select fecha, sum(cantidad) from inscritos i group by fecha order by`. The results are displayed in a table with two columns: 'fecha' and 'sum'.

	fecha	sum
1	2021-08-01	182

comentario: La consulta SQL que utilicé es muy similar a la anterior, la diferencia para llegar al resultado ha sido cambiar el orden de visualización de las columnas (primero fecha y en segundo lugar la sumatoria de cantidades de inscritos, solo para mostrar un reporte mas ordenado acorde a la pregunta) y como en este caso se pide la mayor cantidad de personas, ordene el resultado por sumatoria de cantidad de inscritos de mayor a menor (orden descendente) dejando de esta forma la mayor cantidad de inscritos

según fecha en la primera fila, luego solo me resto limitar el resultado a 1 fila para mostrar la información que se pide.