

AUDIOWEB: PLATAFORMA DE CONTENIDO INTERACTIVO




Realizado por: Rubén Ruiz Ortiz

Fecha: 12/12/25

Profesor: Teresa Alonso Aguilar

Curso: Base de datos

ÍNDICE

ÍNDICE.....	2
 ENUNCIADO DEL PROBLEMA.....	3
MODELO ENTIDAD/RELACIÓN.....	4
MODELO RELACIONAL EN WORKBENCH.....	5
CONSULTAS SQL DEL PROYECTO.....	6
CONCLUSIÓN DEL PROYECTO.....	13

ENUNCIADO DEL PROBLEMA

La aplicación desarrollada es una plataforma de contenidos audiovisuales orientada a la publicación y visualización de vídeos y a la interacción entre los usuarios. Se ha elegido este tipo de aplicación debido al auge de las plataformas de contenido audiovisual en formato vertical, muy utilizadas actualmente en dispositivos móviles, lo que permite modelar de forma realista las entidades y relaciones propias de este tipo de sistemas.

Es necesario almacenar información de los usuarios registrados en la plataforma. De cada usuario se guarda un identificador único, nombre, correo electrónico, contraseña, fecha de registro y país de residencia.

Los usuarios se clasifican mediante una jerarquía en dos tipos: creadores de contenido y administradores.

De los administradores se almacena el nivel de acceso, el sector y el área responsable.

De los creadores de contenido se almacena el número de seguidores, si el creador está certificado, los ingresos generados y el número total de “me gusta” recibidos.

Los usuarios pueden seguir a otros usuarios dentro de la plataforma. Un usuario puede seguir a varios usuarios y, a su vez, puede ser seguido por varios usuarios.

Los creadores de contenido pueden crear vídeos. Cada vídeo es creado por un único creador, mientras que un creador puede crear varios vídeos. De cada vídeo se almacena un identificador único, título, descripción, fecha de publicación, duración del vídeo y la URL del contenido.

Los usuarios pueden interactuar con los vídeos mediante reacciones. Cada reacción registra el tipo de reacción realizada y la fecha en la que se produce. Un usuario puede reaccionar a varios vídeos y un vídeo puede recibir reacciones de varios usuarios.

Los usuarios también pueden mencionar vídeos dentro de la plataforma. Un usuario puede mencionar varios vídeos y un vídeo puede ser mencionado por distintos usuarios.

Los vídeos pueden recibir comentarios. Cada comentario depende de un vídeo concreto y se identifica mediante un número de comentario dentro del mismo, además de almacenar la fecha en la que se realiza el comentario. Un vídeo puede tener varios comentarios, pero cada comentario pertenece a un único vídeo.

La plataforma permite el uso de hashtags para clasificar los vídeos. De cada hashtag se almacena un identificador único y su nombre. Un vídeo puede tener varios hashtags y un mismo hashtag puede estar asociado a distintos vídeos.

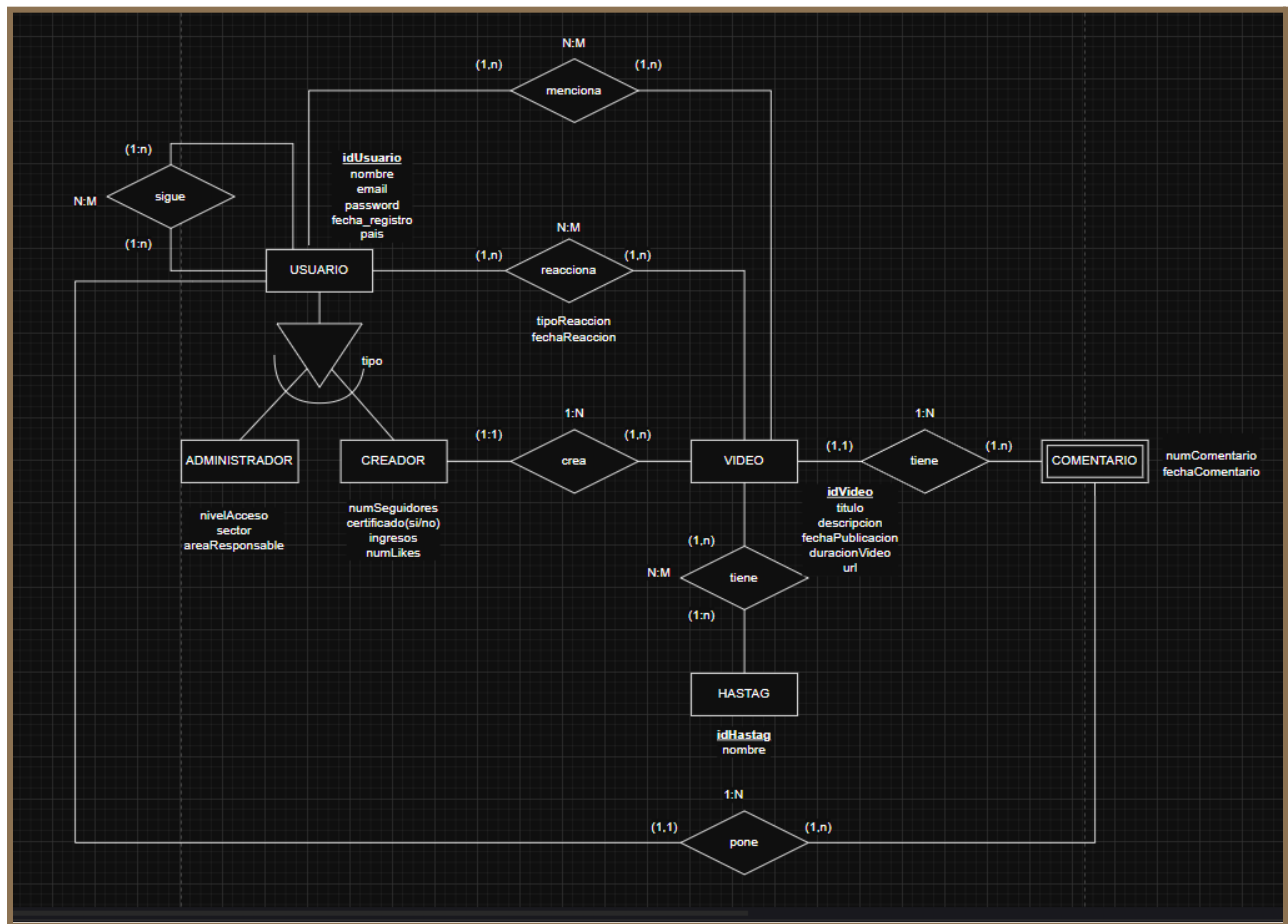
Además, los usuarios pueden escribir varios comentarios, pero un comentario en concreto solo puede ser escrito por un único usuario.

MODELO ENTIDAD/RELACIÓN

El diseño de nuestra plataforma comienza con el Modelo Entidad/Relación, donde definimos la estructura de nuestra base de datos y las reglas de interacción entre las distintas entidades. El núcleo del diseño se basa en una jerarquía de usuarios donde la entidad principal, Usuario, se divide en dos perfiles: el Creador, que genera contenido y gestiona ingresos, y el Administrador, encargado de la gestión interna y sectores responsables.

Entre estas entidades destacan relaciones fundamentales como la de "creación", que vincula cada vídeo con un único creador. También se han implementado relaciones más complejas, como la relación reflexiva de "seguimiento" entre usuarios, o las interacciones de "reacción" y "mención", que permiten que múltiples usuarios interactúen con diversos vídeos registrando datos específicos como el tipo de respuesta o la fecha. Un aspecto técnico relevante es el tratamiento del Comentario como una entidad débil, ya que su existencia depende estrictamente de un vídeo concreto; si el vídeo desaparece, el comentario pierde su sentido y referencia dentro del sistema.

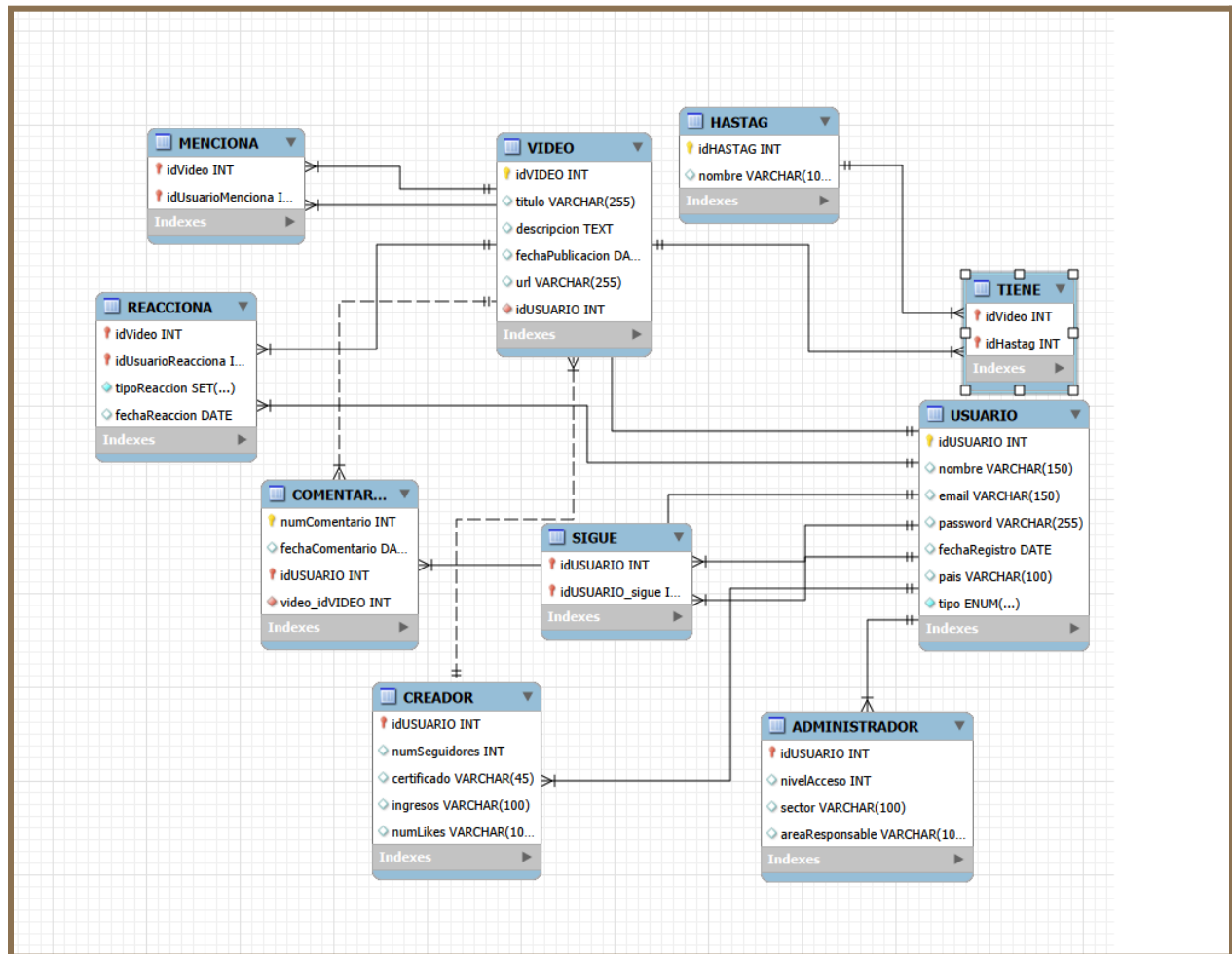
Por último, los vídeos se organizan mediante etiquetas o hashtags. Esto permite agrupar todo el contenido por temas de forma masiva, logrando que los usuarios encuentren lo que buscan de manera mucho más rápida y fácil.



MODELO RELACIONAL EN WORKBENCH

Tras definir el mapa visual del proyecto, utilizamos MySQL Workbench para construir la estructura real de la base de datos, transformando las entidades en tablas físicas donde cada dato se organiza en filas y columnas.

Finalmente, hemos ajustado la configuración de las columnas mediante reglas como ENUM y SET, lo que obliga al sistema a aceptar solo opciones válidas y predefinidas, garantizando que la base de datos esté siempre limpia, ordenada y libre de fallos al introducir la información manualmente



CAMBIOS CON SET Y ENUM

```

alter table USUARIO modify column tipo enum('creador','administrador') not null;

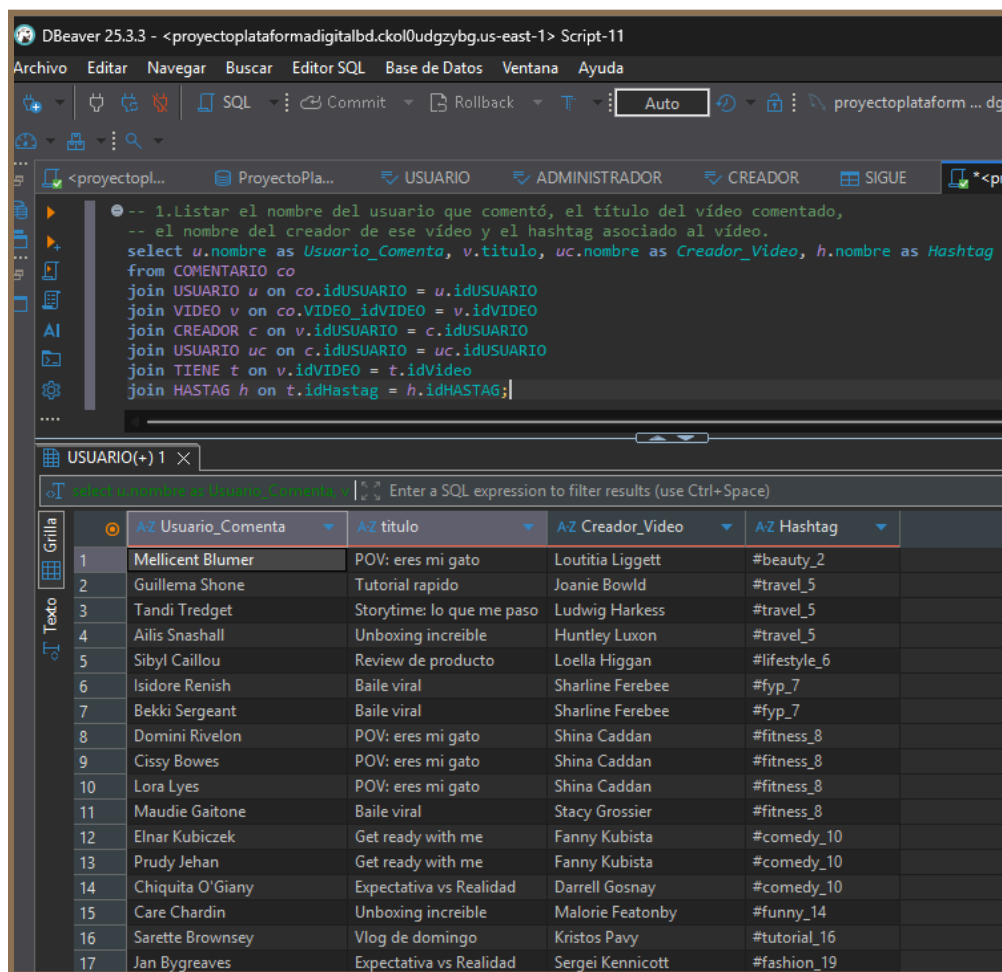
alter table REACCIONA drop column tipoReaccion;

● alter table REACCIONA
  add column tipoReaccion set('Me encanta', 'Like', 'Asombro', 'Divertido', 'No me gusta')
  not null default 'No me gusta' after idUsuarioReacciona;
  
```


CONSULTAS SQL DEL PROYECTO

En este apartado es donde realmente vemos para qué sirve todo el trabajo anterior, ya que usamos el lenguaje SQL para hacerle preguntas a la base de datos y obtener respuestas útiles. En lugar de mirar tablas sueltas, aquí combinamos la información de los usuarios, los vídeos y las interacciones para generar reportes reales, como saber qué hashtags son los más usados o qué creadores están teniendo más éxito. Básicamente, esta sección demuestra cómo transformar un montón de datos guardados en información valiosa para gestionar la plataforma de forma inteligente.

1. Listar el nombre del usuario que comentó, el título del vídeo comentado, el nombre del creador de ese vídeo y el hashtag asociado al vídeo.



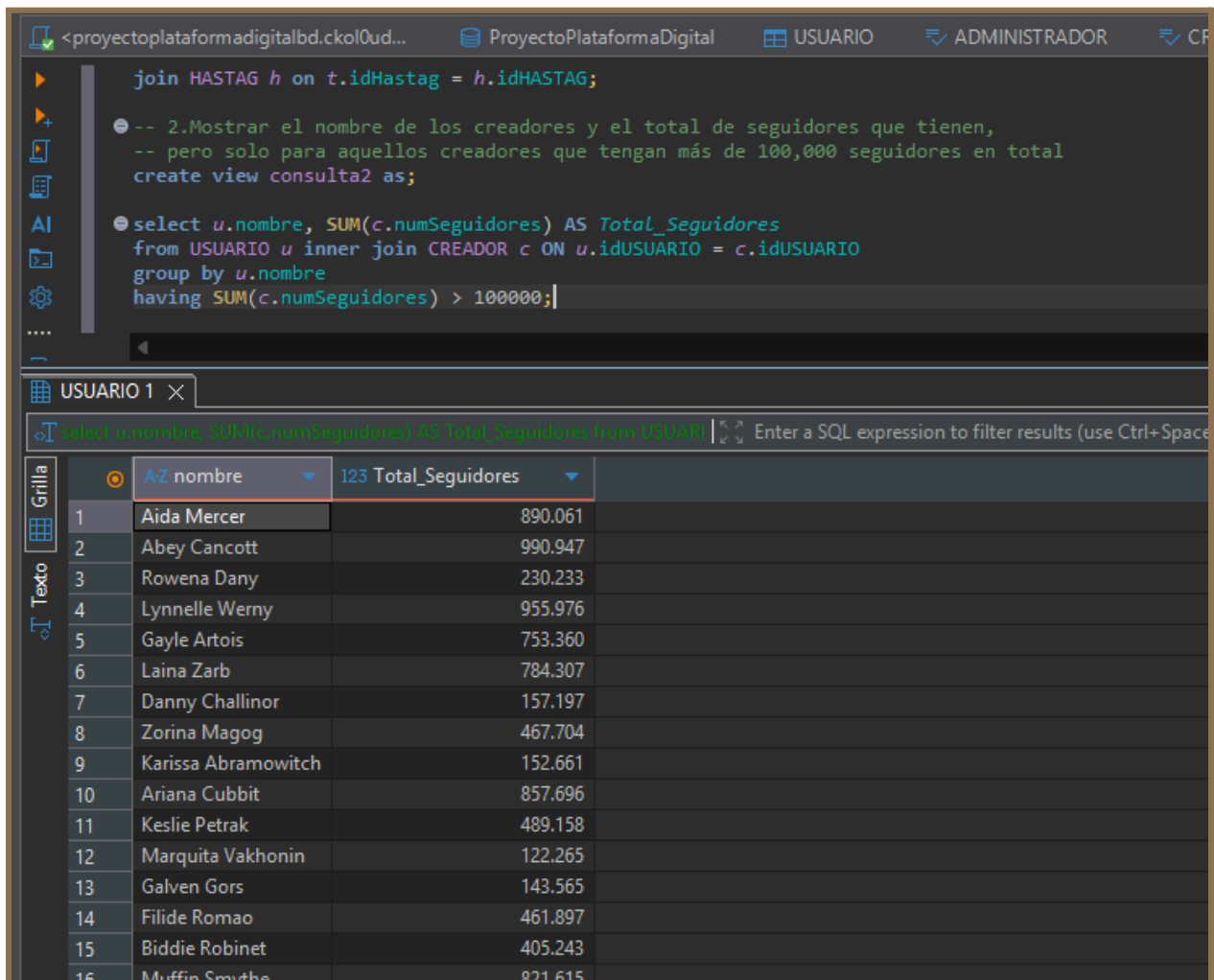
The screenshot shows the DBeaver 25.3.3 interface. The top menu bar includes Archivo, Editar, Navegar, Buscar, Editor SQL, Base de Datos, Ventana, and Ayuda. The main window displays an SQL query in the Editor SQL pane. The query is as follows:

```
-- 1. Listar el nombre del usuario que comentó, el título del vídeo comentado,
-- el nombre del creador de ese vídeo y el hashtag asociado al vídeo.
select u.nombre as Usuario_Comenta, v.titulo, uc.nombre as Creador_Video, h.nombre as Hashtag
from COMENTARIO co
join USUARIO u on co.idUSUARIO = u.idUSUARIO
join VIDEO v on co.VIDEO_idVIDEO = v.idVIDEO
join CREADOR c on v.idUSUARIO = c.idUSUARIO
join USUARIO uc on c.idUSUARIO = uc.idUSUARIO
join TIENE t on v.idVIDEO = t.idVideo
join HASHTAG h on t.idHashtag = h.idHASHTAG;
```

Below the query editor, the results are displayed in a table. The table has five columns: Usuario_Comenta, titulo, Creador_Video, and Hashtag. The results are as follows:

	Usuario_Comenta	titulo	Creador_Video	Hashtag
1	Mellicent Blumer	POV: eres mi gato	Loutitia Liggett	#beauty_2
2	Guillema Shone	Tutorial rapido	Joanie Bowld	#travel_5
3	Tandi Tredget	Storytime: lo que me paso	Ludwig Harkess	#travel_5
4	Ailis Snashall	Unboxing increible	Huntley Luxon	#travel_5
5	Sibyl Caillou	Review de producto	Loella Higgan	#lifestyle_6
6	Isidore Renish	Baile viral	Sharline Ferebee	#fyp_7
7	Bekki Sergeant	Baile viral	Sharline Ferebee	#fyp_7
8	Domini Rivelon	POV: eres mi gato	Shina Caddan	#fitness_8
9	Cissy Bowes	POV: eres mi gato	Shina Caddan	#fitness_8
10	Lora Lyes	POV: eres mi gato	Shina Caddan	#fitness_8
11	Maudie Gaitone	Baile viral	Stacy Grossier	#fitness_8
12	Einar Kubiczek	Get ready with me	Fanny Kubista	#comedy_10
13	Prudy Jehan	Get ready with me	Fanny Kubista	#comedy_10
14	Chiquita O'Giany	Expectativa vs Realidad	Darrell Gosnay	#comedy_10
15	Care Chardin	Unboxing increible	Malorie Featonby	#funny_14
16	Sarette Brownsey	Vlog de domingo	Kristos Pavy	#tutorial_16
17	Jan Bygreaves	Expectativa vs Realidad	Sergei Kennicott	#fashion_19

2. Mostrar el nombre de los creadores y el total de seguidores que tienen, pero solo para aquellos creadores que tengan más de 100,000 seguidores en total



```

join HASTAG h on t.idHastag = h.idHASTAG;

-- 2. Mostrar el nombre de los creadores y el total de seguidores que tienen,
-- pero solo para aquellos creadores que tengan más de 100,000 seguidores en total
create view consulta2 as;

select u.nombre, SUM(c.numSeguidores) AS Total_Seguidores
from USUARIO u inner join CREADOR c ON u.idUSUARIO = c.idUSUARIO
group by u.nombre
having SUM(c.numSeguidores) > 100000;

```

USUARIO 1 X

select u.nombre, SUM(c.numSeguidores) AS Total_Seguidores from USUARIO

	AZ nombre	123 Total_Seguidores
1	Aida Mercer	890.061
2	Abey Cancott	990.947
3	Rowena Dany	230.233
4	Lynnelle Werny	955.976
5	Gayle Artois	753.360
6	Laina Zarb	784.307
7	Danny Challinor	157.197
8	Zorina Magog	467.704
9	Karissa Abramowitch	152.661
10	Ariana Cubbit	857.696
11	Keslie Petrak	489.158
12	Marquita Vakhonin	122.265
13	Galven Gors	143.565
14	Filide Romao	461.897
15	Biddie Robinet	405.243
16	Muffin Southe	821.615

3. Listar los títulos de los vídeos cuyos creadores están por encima de la media de ingresos y que además tienen certificado de cuenta.

```
-- 3. Listar los títulos de los vídeos cuyos creadores están por encima
-- de la media de ingresos y que además tienen certificado de cuenta.
select titulo
from VIDEO v
where idusuario in (
    select idusuario
    from CREADOR c
    where ingresos >
        (select avg(ingresos) from CREADOR ci)
    and certificado = 'sí'
);
```

VIDEO 1

select titulo from VIDEO v where idusuario in (select idusuario from CREADOR c where ingresos > (select avg(ingresos) from CREADOR ci) and certificado = 'sí')

	A-Z titulo
1	Baile viral
2	Baile viral
3	Baile viral
4	Review de producto
5	Tutorial rapido
6	POV: eres mi gato
7	Mi viaje a...
8	Get ready with me
9	Vlog de domingo
10	Mi outfit de hoy
11	Expectativa vs Realidad
12	Challenge del dia
13	Mi rutina de mañana
14	Tutorial rapido
15	Mi viaje a...

4. Mostrar cuántos seguidores ha ganado cada creador por cada año. Debes mostrar el nombre del creador y el año de registro del usuario (para simular el crecimiento anual).

```
-- 4. Mostrar cuántos seguidores ha ganado cada creador por cada año. Debes mostrar el nombre
-- del creador y el año de registro del usuario (para simular el crecimiento anual).
select
    u.nombre as creador,
    year(u.fecharegistro),
    sum(c.numseguidores) as total_seguidores
from USUARIO u
join CREADOR c on u.idusuario = c.idusuario
group by u.nombre, u.fechaRegistro ;
```

creador	year(u.fecharegistro)	total_seguidores
Aida Mercer	2.021	890.061
Abey Cancott	2.020	990.947
Rowena Dany	2.021	230.233
Putnem Izachik	2.022	21.553
Lynnelle Werny	2.021	955.976
Gayle Artois	2.020	753.360
Laina Zarb	2.022	784.307
Danny Challinor	2.020	157.197
Zorina Magog	2.021	467.704
Karissa Abramowitch	2.022	152.661
Ariana Cubbit	2.022	857.696
Keslie Petrak	2.022	489.158
Clemmy Falkingham	2.021	20.847
Marquita Vakhonin	2.022	122.265
Galven Gors	2.021	143.565

5. Obtener el nombre, email y el número total de comentarios de los usuarios de 'españa' que han participado comentando vídeos, pero que nunca han subido un vídeo propio (no están en la tabla video).

```
-- 5. Mostrar el nombre del usuario y el total de reacciones de tipo 'Me encanta' que ha realizado,
-- pero solo para aquellos usuarios que han reaccionado a más de 3 vídeos distintos.
create view consulta5 as;

select
    u.nombre,
    u.email,
    count(r.idVIDEO) as total_reacciones_top
from USUARIO u
join REACCIONA r on u.idUSUARIO = r.idUSUARIOreacciona
where find_in_set('Me encanta', r.tipoReaccion) > 0
group by u.nombre, u.email
having count(r.idVIDEO) > 3;
```

USUARIO 1 X

Enter a SQL expression to filter results (use Ctrl+Space)

AZ nombre	AZ email	123 total_reacciones_top
Brad Crowthe	bcrowthe9i@shop-pro.jp	4

6. Mostrar el nombre de los administradores en minúsculas y su sector. Si el área responsable es nula, mostrar 'pendiente de asignar'. Filtrar solo los de sectores que contengan la letra 'a'.

```
-- 6. Mostrar el nombre de los administradores en minúsculas y su sector.
-- Si el área responsable es nula, mostrar 'pendiente de asignar'.
-- Filtrar solo los de sectores que contengan la letra 'a'.
select lower(u.nombre) as administrador, a.sector, ifnull(a.arearesponsable, 'pendiente de asignar') as area
from USUARIO u
join ADMINISTRADOR a on u.idusuario = a.idusuario
where a.sector like '%a%';
```

ADMINISTRADOR 1 ×

select lower(u.nombre) as administrador, a.sector, ifnull(a.arearesponsable, 'pendiente de asignar') as area | Enter a SQL expression to filter results (use Ctrl+Space)

	AZ administrador	AZ sector	AZ area
1	abey cancott	Finanzas	Gestion de Pagos
2	rowena dany	Legal	Asuntos Gubernamentales
3	putnem izachik	Recursos Humanos	Capacitacion
4	lynnelle werny	Recursos Humanos	Capacitacion
5	danny challinor	Recursos Humanos	Reclutamiento
6	zorina magog	Recursos Humanos	Gestion de Personal
7	karissa abramowitch	Finanzas	Auditoria Interna
8	ariana cubbit	Legal	Terminos de Servicio
9	keslie petrak	Marketing	Analisis de Metricas
10	clemmy falkingham	Finanzas	Contabilidad
11	marquita vakhonin	Seguridad	Proteccion de Datos
12	galven gors	Legal	Terminos de Servicio

7. Mostrar el nombre del seguidor y el nombre del creador al que sigue, filtrando para que el usuario seguido sea estrictamente del tipo 'creador'.

```
-- 7. Mostrar el nombre del seguidor y el nombre del creador al que sigue,
-- filtrando para que el usuario seguido sea estrictamente del tipo 'creador'.
select u.nombre as usuario_seguidor, u2.nombre as creador_seguido, u2.tipo
from SIGUE s
inner join USUARIO u on s.idusuario = u.idusuario
inner join USUARIO u2 on s.idusuario_sigue = u2.idusuario
where u2.tipo = 'creador';
```

USUARIO 1 X

select u.nombre as usuario_seguidor, u2.nombre as creador_seguido, u2.tipo | Enter a SQL expression to filter results

	AZ usuario_seguidor	AZ creador_seguido	AZ tipo	
1	Ramona Passman	Rowena Dany	creador	
2	Clareta Elliott	Danny Challinor	creador	
3	Prudy Jehan	Karissa Abramowitch	creador	
4	Zed Palfreman	Karissa Abramowitch	creador	
5	Burt Proppers	Karissa Abramowitch	creador	
6	Adriena Wakerley	Ariana Cubbit	creador	
7	Noell Southway	Filide Romao	creador	
8	Barnabe Castillo	Sunny Panks	creador	
9	Roxanne Bukac	Martynne Symms	creador	
10	Reena Isacq	Martynne Symms	creador	
11	Dante Ogles	Martynne Symms	creador	
12	Harli Thebeau	Felita Ledingham	creador	
13	Sada Bristowe	Iolande Brunini	creador	
14	Claudianus Ferrao	Tadd Khomishin	creador	
15	Andrey Bentley	Tadd Khomishin	creador	

CONCLUSIÓN DEL PROYECTO

Con la realización de este proyecto he aprendido a organizar la información de manera más clara y útil, como se haría en una aplicación web real. Me he dado cuenta de lo importante que es pensar bien el diseño de la base de datos antes de empezar a programar. Además, me ha servido para poner en práctica lo que hemos visto en clase, usando un ejemplo cercano como una plataforma de vídeos, y también para mejorar la forma en que estructura un proyecto de principio a fin.

En la siguiente actividad, tuve que realizar la carga de datos para mi proyecto, lo que me ayudó a profundizar a la hora de manejar el DBeaver, el control de las unidades en el sql, ya que muchas veces me daba error a la hora de añadir datos porque se me quedaba corto e incluso el empleo de DDL y DML para borrar datos de una columna por si me equivocaba a la hora de importar. Por cosas a mejorar, podría ser la complejidad de las consultas ya que no tengo un nivel tan alto como para complicarme a la hora de hacerlas.

Enlace a GitHub

https://github.com/RubenRuiz007/ProyectoBD_PlataformaDigital