# Práctica: Poblado y consultas

# use crunchy;

## /\*1-los nombres de listas creadas los usuarios con su nombre y su fecha de creación y los animes que contienen y las listas creadas a partir de 2019\*/

# select lista.nombre as 'nombre de la lista' ,usuario.nickname as 'nickname del usuario',lista.fecha\_de\_creacion as 'fecha de creacion de la lista',anime.nombre as 'nombre del anime'

# from lista

# join usuario on usuario.id=lista.id\_usuario

# join lista\_anime on lista\_anime.id\_lista = lista.id

# join anime on anime.id = lista\_anime.id\_anime

# where lista.fecha\_de\_creacion > date\_sub(curdate(),interval 4 year);

# 

# Al menos 5 consultas deberán trabajar sobre 3 tablas 1/5

# Al menos 5 consultas utilizaran ́ funciones de fecha no repetidas 1/5

## /\*2-usuarios que pagan un plan mensual y con quien lo comparten\*/

# select usuario.nickname as 'usuario que paga' ,nombreRata.nickname as 'usuario que ratea'

# from usuario

# join plan\_menusal\_usuario on plan\_menusal\_usuario.id\_usuario = usuario.id

# join plan\_mensual on plan\_mensual.id = plan\_menusal\_usuario.id\_plan\_mensual

# join metodo\_de\_pago on metodo\_de\_pago.id\_plan\_mensual = plan\_mensual.id

# join usuario\_comparte\_plan\_mensual on usuario\_comparte\_plan\_mensual.id\_plan\_mensual = plan\_mensual.id

# join usuario nombreRata on nombreRata.id = usuario\_comparte\_plan\_mensual.id\_usuario\_ratea;

# 

* Al menos 5 consultas deberán trabajar sobre 4 tablas 1/5

# /\*3-personaje que no tengan adaptación animada y el número de veces que un producto suyo se ha vendido\*/

select personaje.nombre, count(producto.id\_personaje) as 'producto de ese personaje'

from personaje

join producto on producto.id\_personaje = personaje.id

where personaje.id not in (select personaje.id

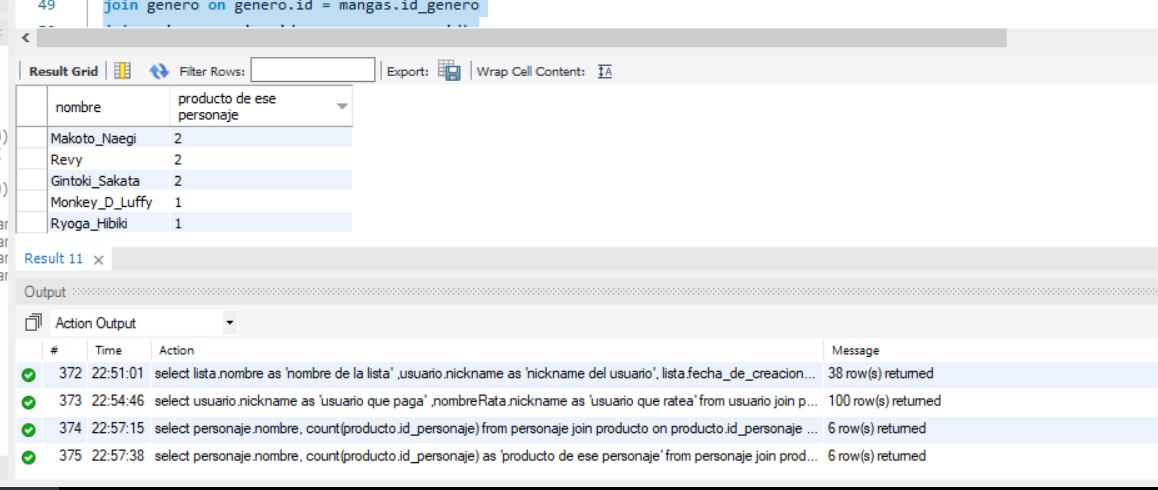
from personaje

join mangas on mangas.id = personaje.id\_manga

join genero on genero.id = mangas.id\_genero

join anime on anime.id\_genero = genero.id)

group by producto.id\_personaje;



* Al menos 5 consultas deberán trabajar sobre 3 tablas 2/5
* Al menos 2 consultas subconsultas 1/2
* Al menos 4 consultas utilizaran ́ GROUP BY / HAVING 1/4

# /\*4-la lista que hace más tiempo se ha creado, el usuario que la creo, el pan de ese usuario y factura\*/

select lista.fecha\_de\_creacion as 'fecha\_lista',usuario.nickname as 'usuario que la creo' ,metodo\_de\_pago.factura

from usuario

join lista on lista.id\_usuario = usuario.id

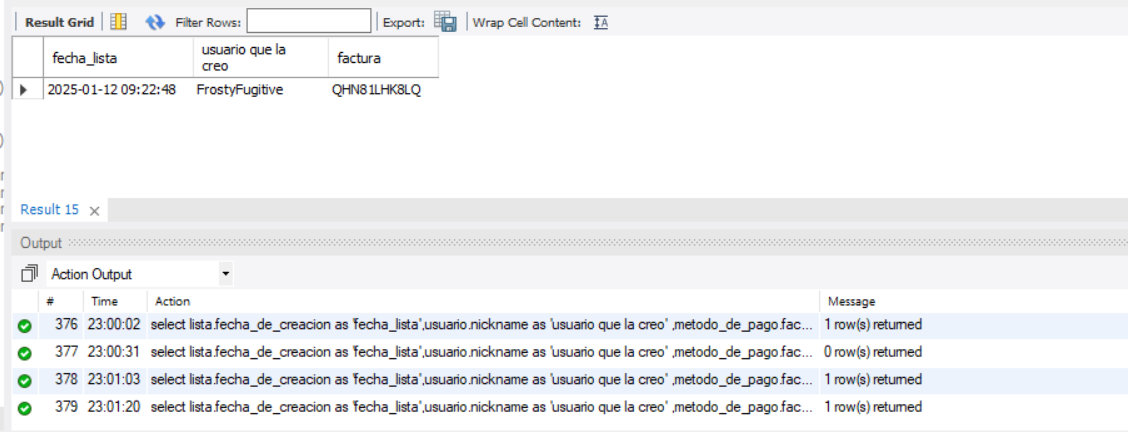
join plan\_menusal\_usuario on plan\_menusal\_usuario.id\_usuario = usuario.id

join plan\_mensual on plan\_menusal\_usuario.id\_plan\_mensual = plan\_mensual.id

join metodo\_de\_pago on metodo\_de\_pago.id\_plan\_mensual = plan\_mensual.id

having fecha\_lista = (select max(lista.fecha\_de\_creacion)

from lista);



* Al menos 5 consultas deberán trabajar sobre 4 tablas 2/5
* Al menos 2 consultas contendrán subconsultas 2/2
* Al menos 4 consultas utilizaran GROUP BY / HAVING 2/4

# /\*5- Compañías de envió las cuales estén en un atasco el producto que es y al usuario que va destinado\*/

select compañia\_de\_envio.nombre as 'nombre\_compañia', compañia\_de\_envio.estado\_del\_trafico ,personaje.nombre as 'nombre del producto' , usuario.nickname

from compañia\_de\_envio

join producto on producto.id\_compañia\_de\_envio = compañia\_de\_envio.id

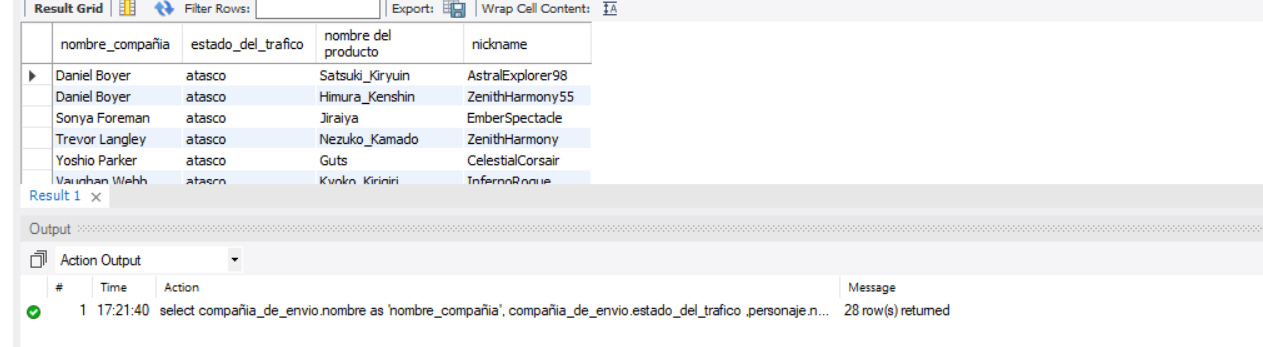
join personaje on personaje.id = producto.id\_personaje

join mangas on mangas.id = personaje.id\_manga

join usuario\_mangas on usuario\_mangas.id\_mangas = mangas.id

join usuario on usuario.id = usuario\_mangas.id\_usuario

having estado\_del\_trafico = 'atasco';



* Al menos 4 consultas utilizaran GROUP BY / HAVING 2/4
* Al menos 5 consultas deberán trabajar sobre 4 tablas 3/5

# /\*6-el personaje del que más pedidos se han hecho con sus animes (si lo tiene) \*/

select count(\*) as 'numero\_de\_envios',personaje.nombre as 'nombre\_personaje' , anime.nombre as 'nombre\_anime'

from personaje

join producto on producto.id\_personaje = personaje.id

join mangas on mangas.id = personaje.id\_manga

join genero on genero.id = mangas.id\_genero

join anime on anime.id\_genero = genero.id

group by producto.id\_personaje

having numero\_de\_envios= (select max(numero\_de\_envios)

from(

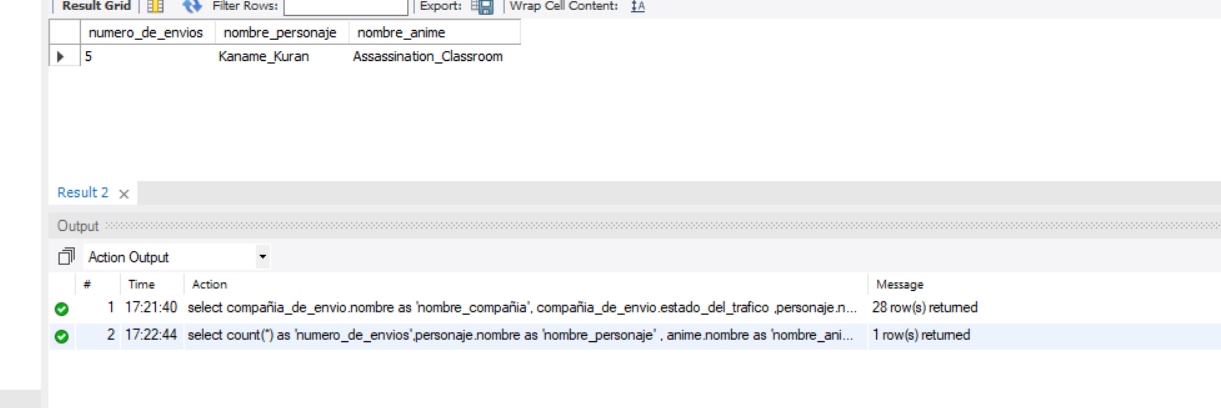
select count(\*) as 'numero\_de\_envios'

from personaje

join producto on producto.id\_personaje = personaje.id

group by producto.id\_personaje)

as numero\_envios);



* Al menos 5 consultas deberán trabajar sobre 4 tablas 4/5
* Al menos 4 consultas utilizaran GROUP BY / HAVING 3/ 4

# /\*7-todos los usuarios que tengan más de 2 año con la plataforma y las veces que han comprado en la tienda (tomaos el curdate () como la fecha máxima de mi base) \*/

select usuario.nickname ,plan\_mensual.fecha\_inicio as 'fecha', count(\*) as 'veces\_comprado\_tienda'

from producto

join personaje on personaje.id = producto.id\_personaje

join mangas on mangas.id = personaje.id\_manga

join usuario\_mangas on usuario\_mangas.id\_mangas = mangas.id

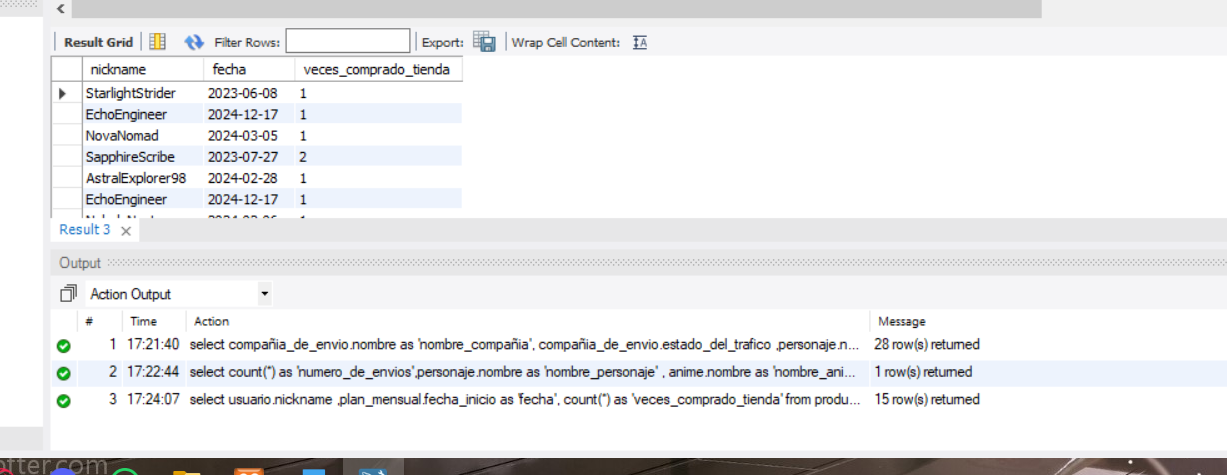
join usuario on usuario.id = usuario\_mangas.id\_usuario

join plan\_menusal\_usuario on plan\_menusal\_usuario.id\_usuario = usuario.id

join plan\_mensual on plan\_mensual.id = plan\_menusal\_usuario.id\_plan\_mensual

where plan\_mensual.fecha\_inicio >date\_sub((select max(fecha) from ( select plan\_mensual.fecha\_inicio as 'fecha' from plan\_mensual )as fecha\_maxima) ,interval 24 month)

group by producto.id\_usuario;



* Al menos 5 consultas deberán trabajar sobre 4 tablas 5/5
* Al menos 4 consultas utilizaran GROUP BY / HAVING 4/4
* Al menos 5 consultas utilizaran ́ funciones de fecha no repetidas 2/5

# /\*8-las listas creadas por cada usuario y las litas creadas solo en 2024 \*/

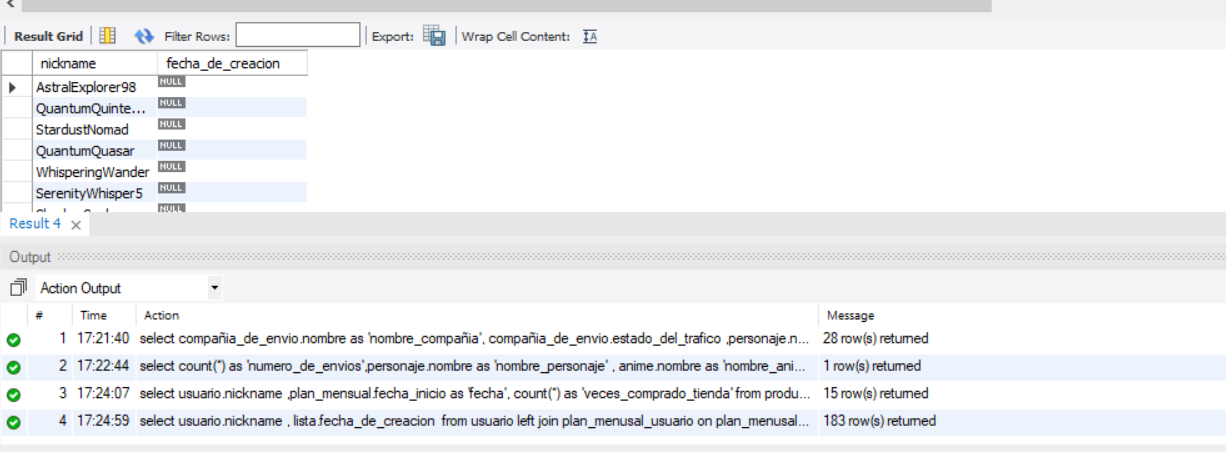
select usuario.nickname , lista.fecha\_de\_creacion

from usuario

left join plan\_menusal\_usuario on plan\_menusal\_usuario.id\_usuario = usuario.id

left join plan\_mensual on plan\_mensual.id = plan\_menusal\_usuario.id\_plan\_mensual

left join lista on lista.id\_usuario = usuario.id;

****

* Al menos 2 consultas utilizaran OUTER JOIN 1/2
* Al menos 5 consultas deberán trabajar sobre 3 tablas 3/5

# /\*9-los usuarios que no han creado una lista y los mangas que se ha leído\*/

create view usuario\_crea\_listas as

select usuario.id

from usuario

join lista on lista.id\_usuario = usuario.id;

select usuario.nickname , mangas.nombre

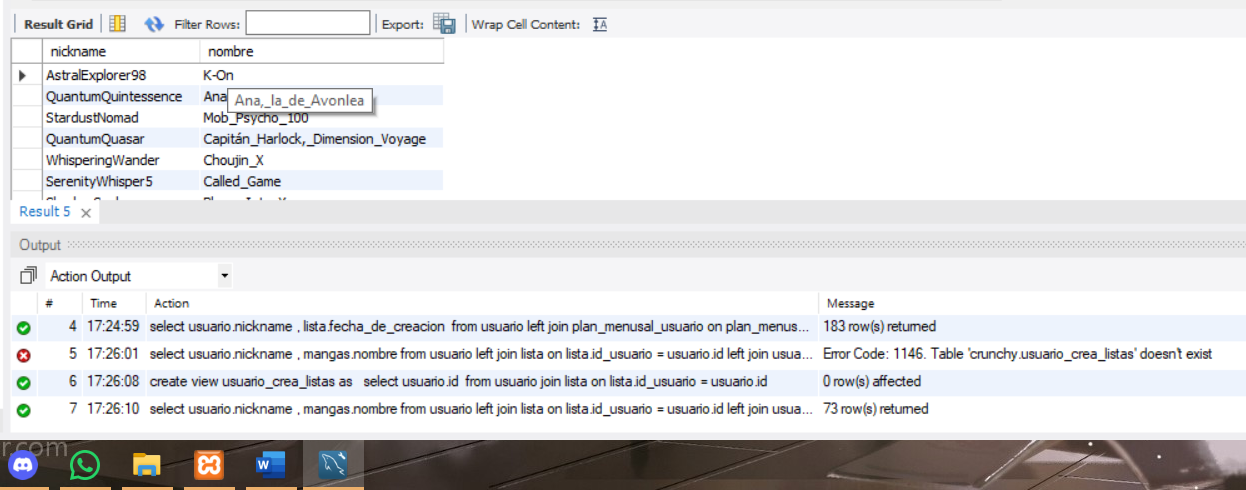
from usuario

left join lista on lista.id\_usuario = usuario.id

left join usuario\_mangas on usuario\_mangas.id\_usuario = usuario.id

left join mangas on mangas.id = usuario\_mangas.id\_mangas

where usuario.id not in (select id from usuario\_crea\_listas) ;



* Al menos 5 consultas deberán trabajar sobre 3 tablas 3/5
* Al menos 2 consultas utilizaran ́ OUTER JOIN 2/2
* Al menos 5 consultas utilizaran ́ vistas 1/5

# /\*10-los usuarios que se han acabado un manga en 2022, los nombres de los mangas y su género \*/

select usuario.nickname,mangas.nombre

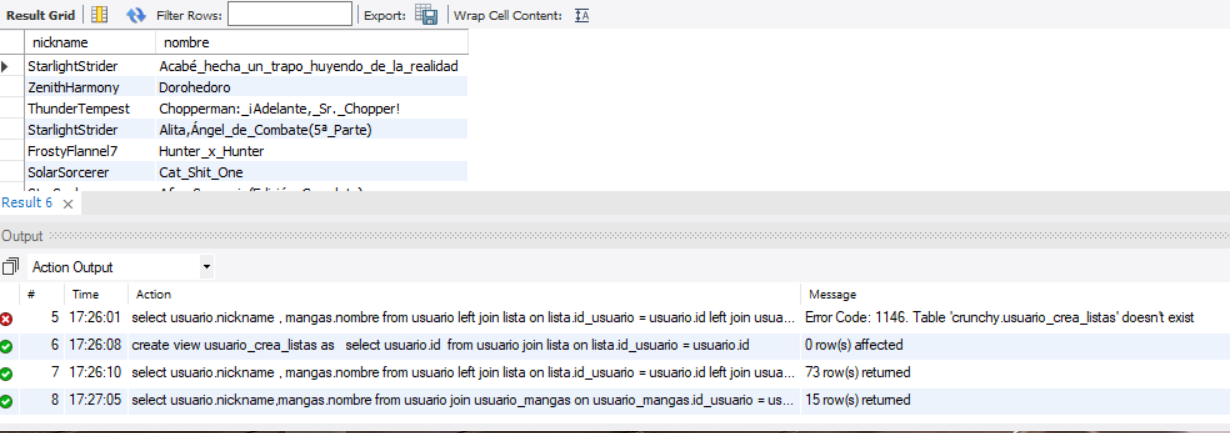
from usuario

join usuario\_mangas on usuario\_mangas.id\_usuario = usuario.id

join mangas on mangas.id = usuario\_mangas.id\_mangas

join genero on genero.id = mangas.id\_genero

where year(fecha\_final\_del\_usuario) = year(date\_sub(curdate(),interval 2 year));

* Al menos 5 consultas utilizaran funciones de fecha no repetidas 3/5
* Al menos 5 consultas deberán trabajar sobre 3 tablas 4/5

# /\*11-los usuarios que tengan el plan mensual de superfan y las listas que han creado desde 2022\*/

create view usuarios\_super\_fan as

select usuario.id as 'usuario\_id'

from usuario

where usuario.id\_plan\_mensual= 2;

drop view usuarios\_super\_fan;

select usuario.nickname , lista.nombre

from usuario

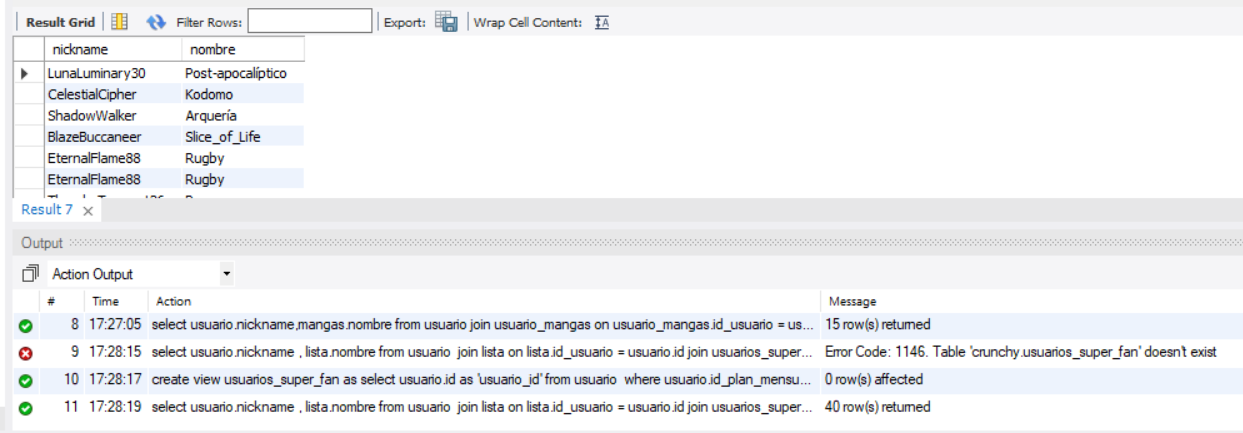
join lista on lista.id\_usuario = usuario.id

join usuarios\_super\_fan on usuarios\_super\_fan.usuario\_id = usuario.id

join lista\_anime on lista\_anime.id\_lista = lista.id

join anime on anime.id = lista\_anime.id\_anime

where lista.fecha\_de\_creacion < date\_add(curdate(),interval 2 year);

****

* Al menos 5 consultas utilizaran funciones de fecha no repetidas 4/5
* Al menos 5 consultas deberán trabajar sobre 3 tablas 5/5
* Al menos 5 consultas utilizaran ́ vistas 2/5

# /\*12-usuarios que se han empezado un manga hace 36 meses y que tenga estado de ánimo triste\*/

create view usuarios\_tristones as

select usuario.id as 'usuario\_id'

from usuario

where estado\_animo = 'triste';

drop view usuarios\_tristones;

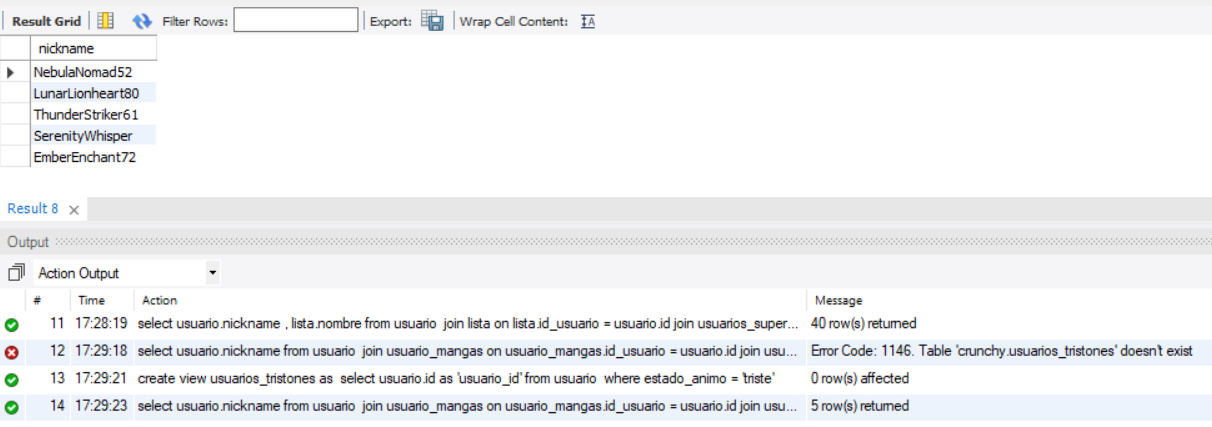
select usuario.nickname

from usuario

join usuario\_mangas on usuario\_mangas.id\_usuario = usuario.id

join usuarios\_tristones on usuarios\_tristones.usuario\_id = usuario.id

where month(fecha\_final\_del\_usuario) = month(date\_sub(curdate(),interval 36 month));



* Al menos 5 consultas utilizaran funciones de fecha no repetidas 5/5
* Al menos 5 consultas utilizaran vistas 3/5

# /\*13-Usuarios con más de 30 episodios vistos y que se han visto el anime de ‘Death\_Note' \*/

create view usuarios\_episoidos\_vistos

as

select usuario.id as 'id\_usuarioo', usuario.nickname as 'nombre\_user'

from usuario

where usuario.episodios\_vistos >30;

drop view usuarios\_episoidos\_vistos;

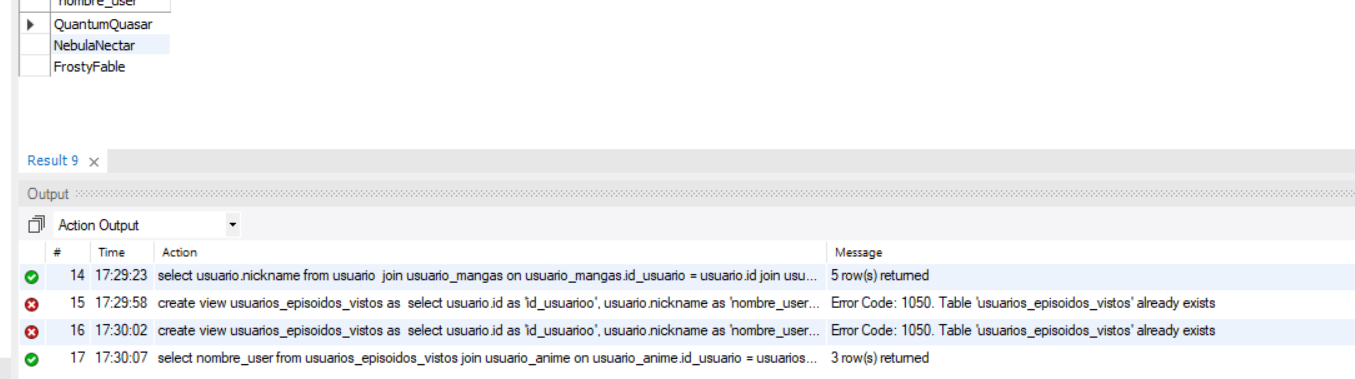
select nombre\_user

from usuarios\_episoidos\_vistos

join usuario\_anime on usuario\_anime.id\_usuario = usuarios\_episoidos\_vistos.id\_usuarioo

join anime on anime.id = usuario\_anime.id\_anime

where anime.nombre = 'Death\_Note';



* Al menos 5 consultas utilizaran vistas 4/5

# /\*14-Usuarios que tenga en plan mensual de fan de los cuales que se ha leído un manga con traducción y que tengan subtítulos\*/

create view usuarios\_fan

as

select usuario.id as 'id\_fan\_user', usuario.nickname as 'nombre\_user'

from usuario

where usuario.id\_plan\_mensual = 0;

drop view usuarios\_fan;

select nombre\_user as 'nombre de usuario'

from usuarios\_fan

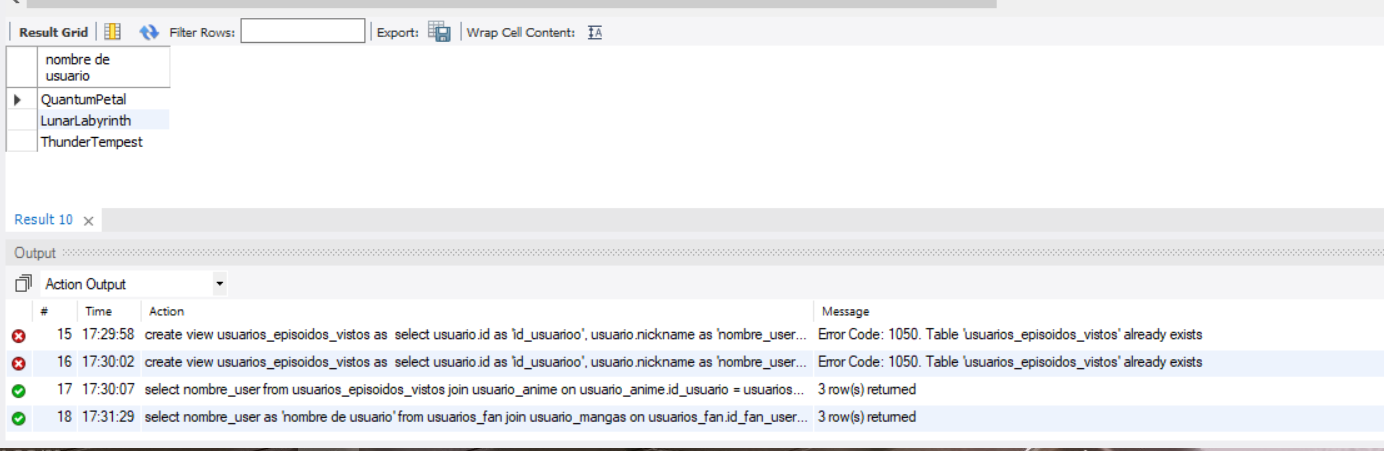
join usuario\_mangas on usuarios\_fan.id\_fan\_user = usuario\_mangas.id\_usuario

join mangas on mangas.id = usuario\_mangas.id\_mangas

join traduccion\_manga on traduccion\_manga.id\_manga= mangas.id

join traducion on traducion.id = traduccion\_manga.id\_traduccion

where traducion.subtitulos = 1;



* Al menos 5 consultas utilizaran vistas 5/5

# /\*15-personajes que midan más de un 1.90 y que tengan adaptación animada \*/

create view personajes\_altos

as

select nombre as 'nombre\_personajes' , personaje.id as 'id\_personaje',id\_manga

from personaje

where estatura >= 190;

drop view personajes\_altos;

select nombre\_personajes

from personajes\_altos

join mangas on mangas.id = personajes\_altos.id\_manga

join genero on mangas.id\_genero = genero.id

join anime on anime.id\_genero = genero.id