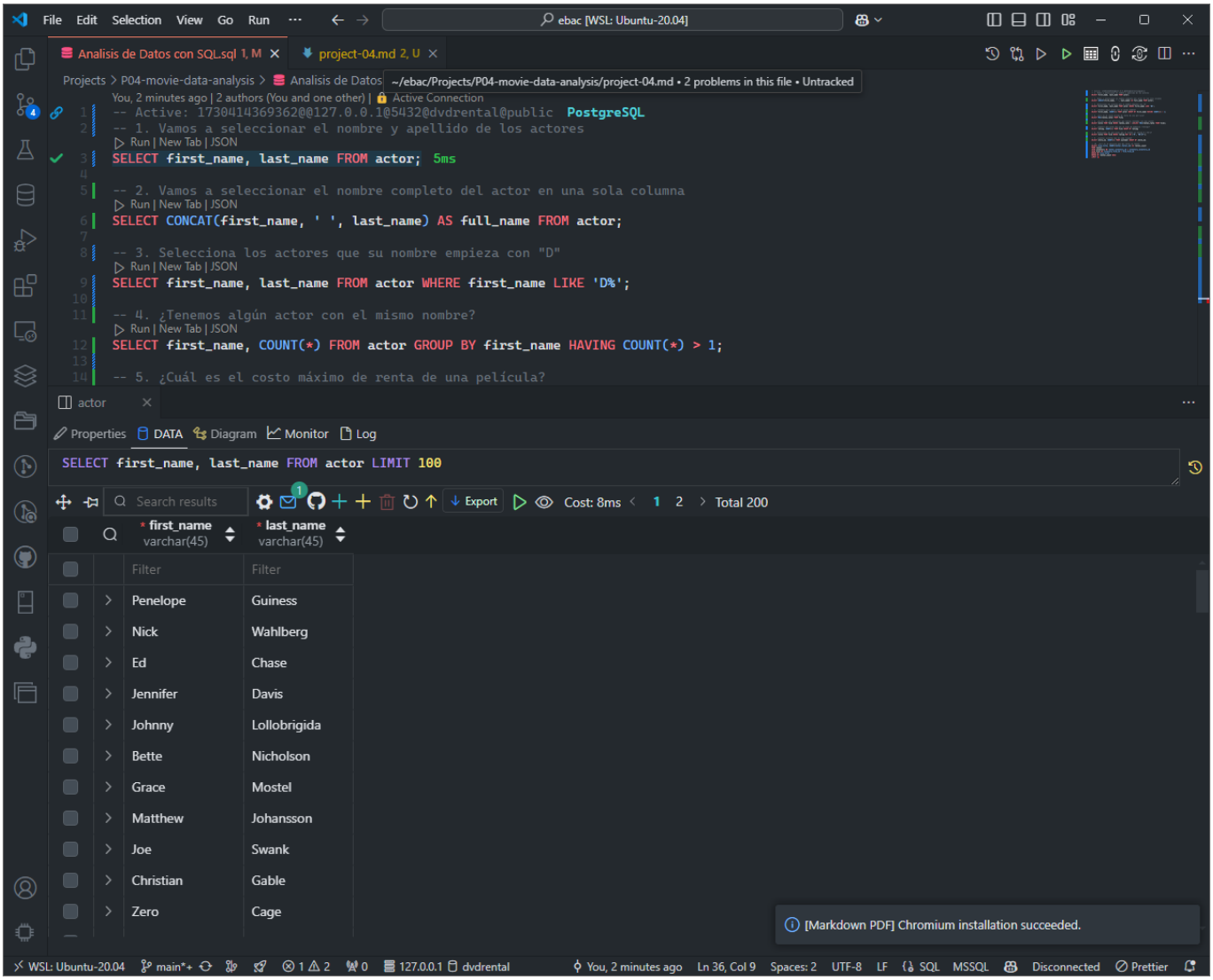
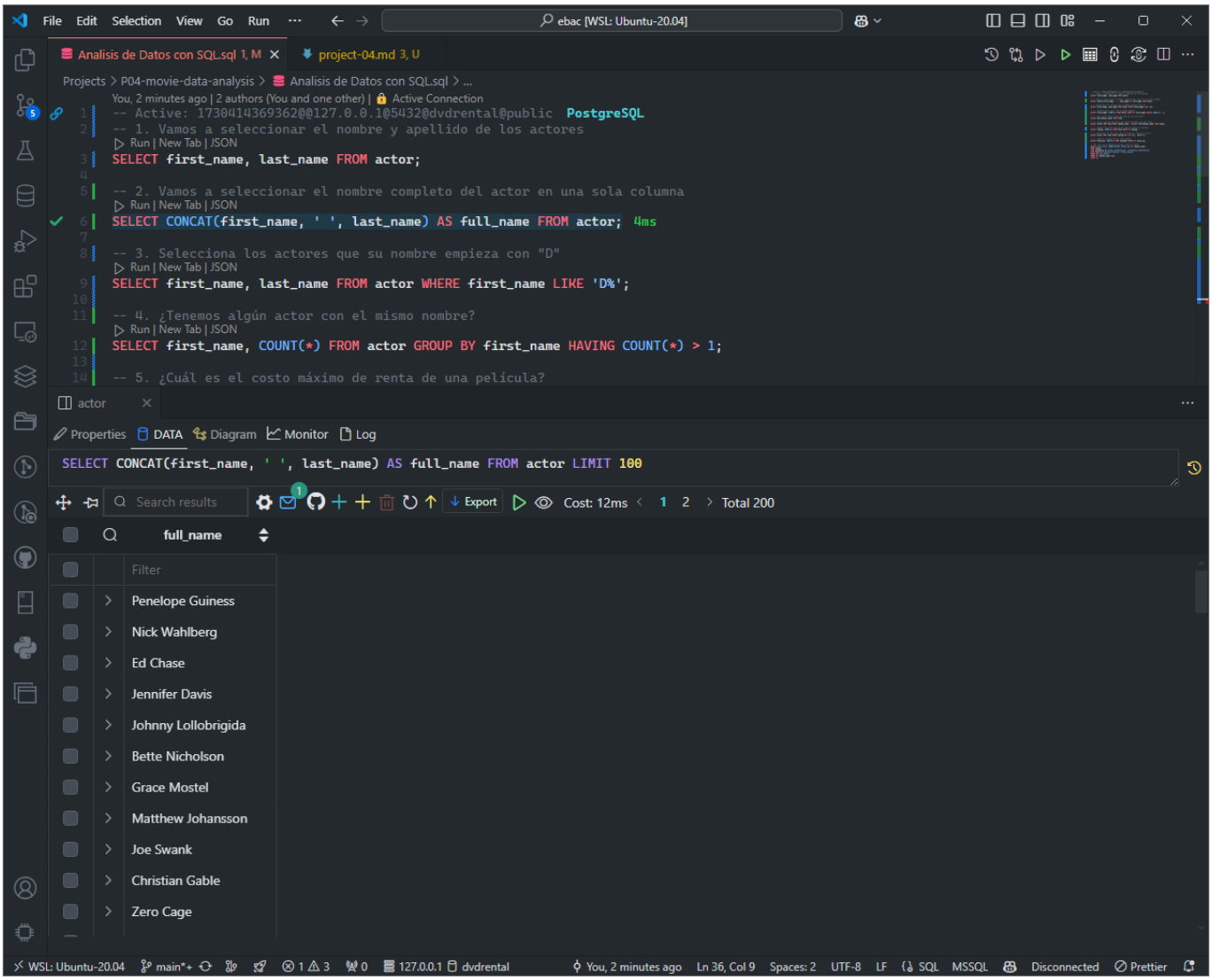
Project 04: Movie Data Analysis

Author: Juliho Castillo Colmenares

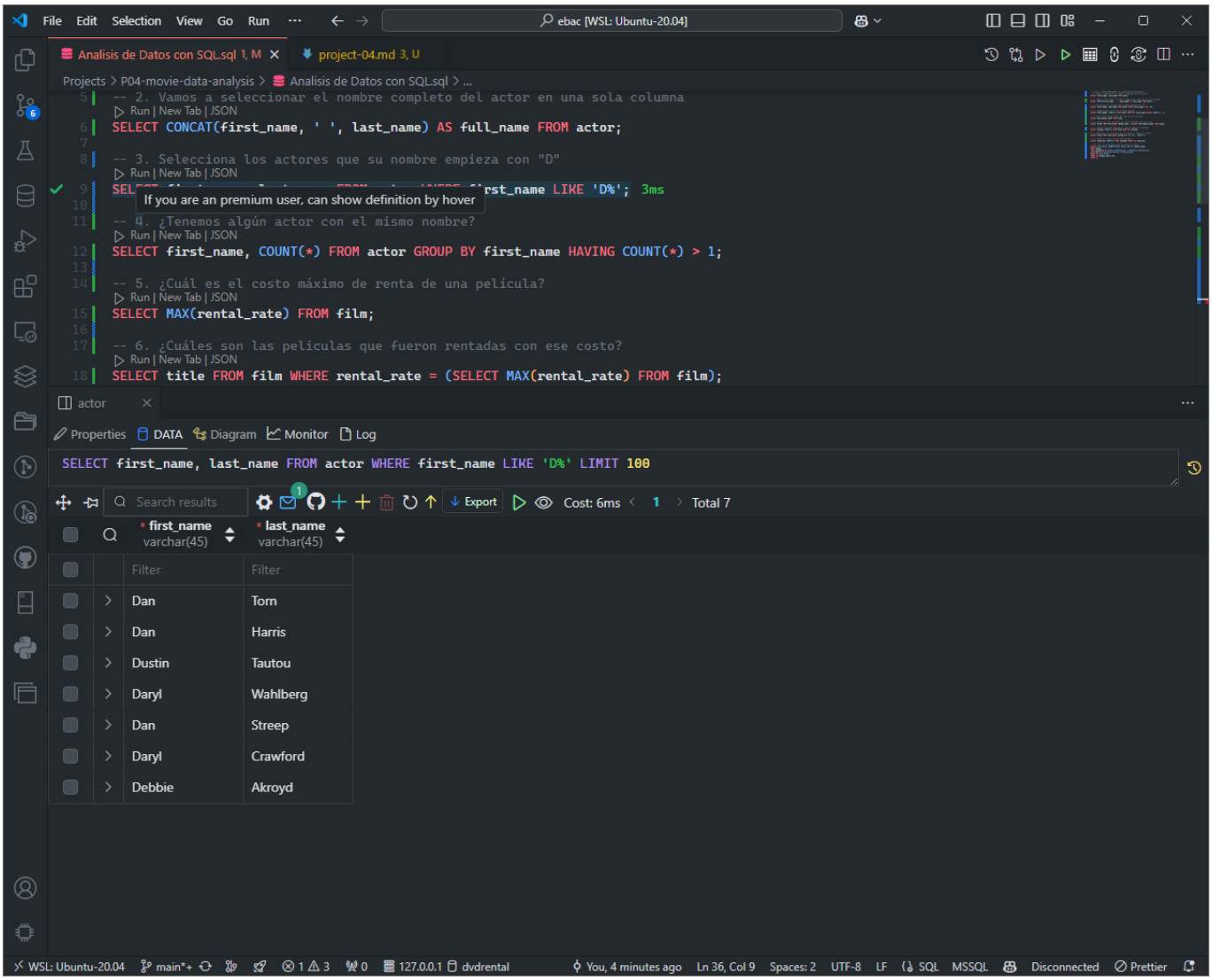
Exercise 1



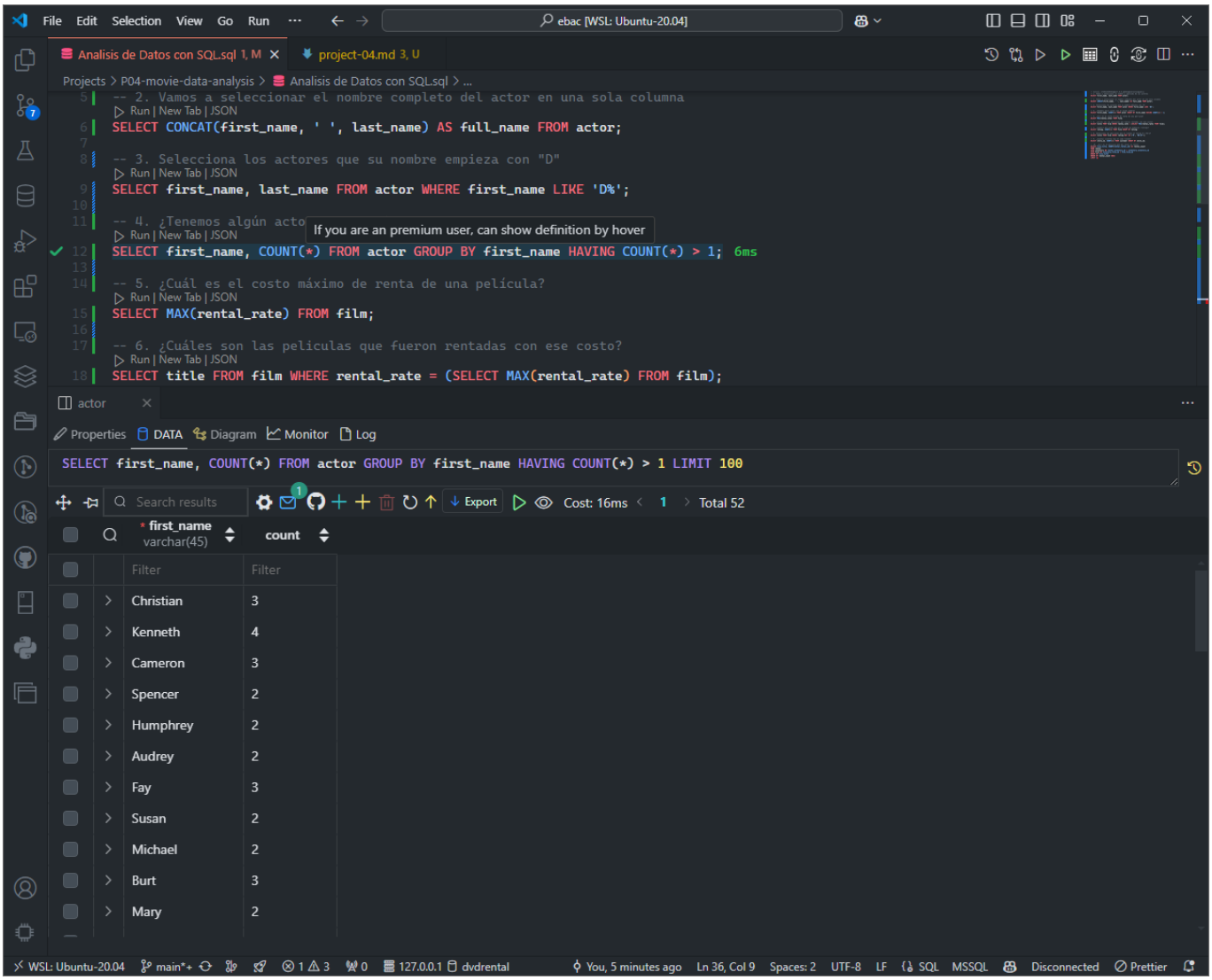
Exercise 2


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
You, 2 minutes ago | 2 authors (You and one other) | Active Connection
-- Active: 1730414369362@@127.0.0.1@5432@dvdrental@public PostgreSQL
-- 1. Vamos a seleccionar el nombre y apellido de los actores
Run | New Tab | JSON
SELECT first_name, last_name FROM actor;
-- 2. Vamos a seleccionar el nombre completo del actor en una sola columna
Run | New Tab | JSON
SELECT CONCAT(first_name, ' ', last_name) AS full_name FROM actor; 4ms
-- 3. Selecciona los actores que su nombre empieza con "D"
Run | New Tab | JSON
SELECT first_name, last_name FROM actor WHERE first_name LIKE 'D%';
-- 4. ¿Tenemos algún actor con el mismo nombre?
Run | New Tab | JSON
SELECT first_name, COUNT(*) FROM actor GROUP BY first_name HAVING COUNT(*) > 1;
-- 5. ¿Cuál es el costo máximo de renta de una película?
actor
Properties DATA Diagram Monitor Log
SELECT CONCAT(first_name, ' ', last_name) AS full_name FROM actor LIMIT 100
Search results
Export
Cost: 12ms
1 2
Total 200
full_name
Filter
Penelope Guiness
Nick Wahlberg
Ed Chase
Jennifer Davis
Johnny Lollobrigida
Bette Nicholson
Grace Mostel
Matthew Johansson
Joe Swank
Christian Gable
Zero Cage
WSL: Ubuntu-20.04
main*+
1 3
0
127.0.0.1 dvdrental
You, 2 minutes ago
Ln 36, Col 9
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

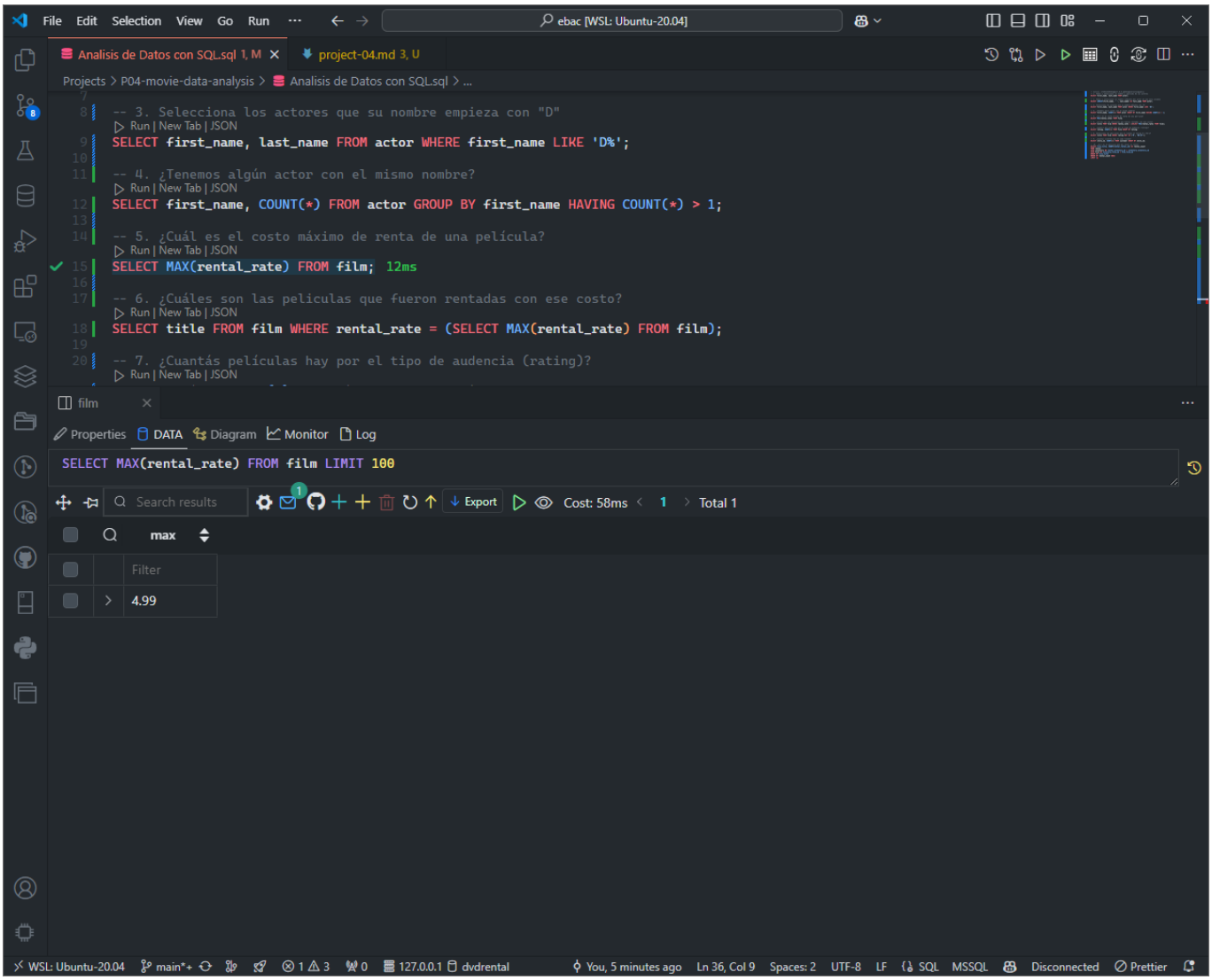
Exercise 3


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 2. Vamos a seleccionar el nombre completo del actor en una sola columna
Run | New Tab | JSON
SELECT CONCAT(first_name, ' ', last_name) AS full_name FROM actor;
-- 3. Selecciona los actores que su nombre empieza con "D"
Run | New Tab | JSON
If you are an premium user, can show definition by hover
rst_name LIKE 'D%'; 3ms
-- 4. ¿Tenemos algún actor con el mismo nombre?
Run | New Tab | JSON
SELECT first_name, COUNT(*) FROM actor GROUP BY first_name HAVING COUNT(*) > 1;
-- 5. ¿Cuál es el costo máximo de renta de una película?
Run | New Tab | JSON
SELECT MAX(rental_rate) FROM film;
-- 6. ¿Cuáles son las peliculas que fueron rentadas con ese costo?
Run | New Tab | JSON
SELECT title FROM film WHERE rental_rate = (SELECT MAX(rental_rate) FROM film);
actor
Properties DATA Diagram Monitor Log
SELECT first_name, last_name FROM actor WHERE first_name LIKE 'D%' LIMIT 100
Search results
Export
Cost: 6ms
1
Total 7
first_name varchar(45)
last_name varchar(45)
Filter
Filter
Dan Torn
Dan Harris
Dustin Tautou
Daryl Wahlberg
Dan Streep
Daryl Crawford
Debbie Akroyd
WSL: Ubuntu-20.04
main*+
127.0.0.1 dvdrental
You, 4 minutes ago
Ln 36, Col 9
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

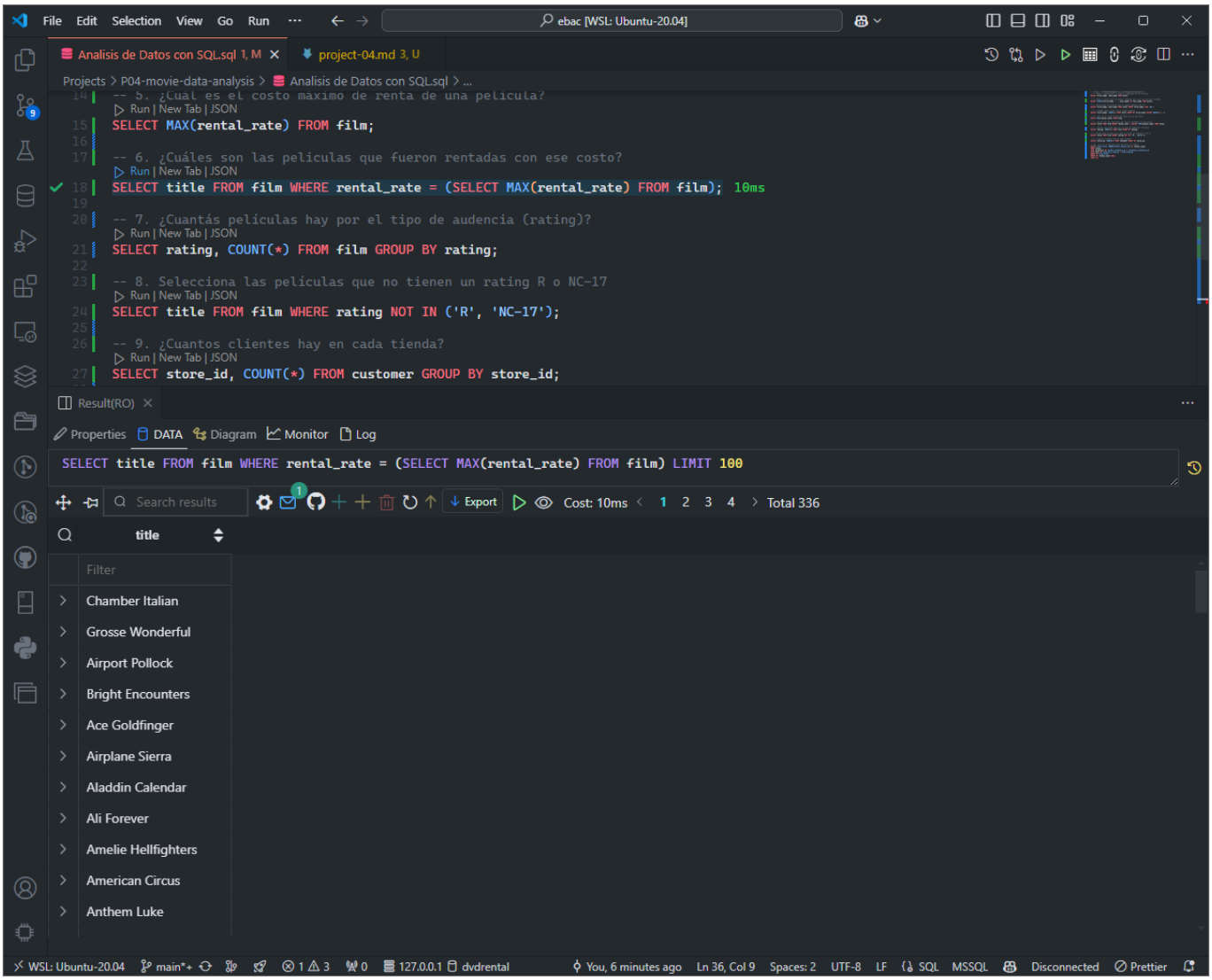
Exercise 4


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 2. Vamos a seleccionar el nombre completo del actor en una sola columna
Run | New Tab | JSON
SELECT CONCAT(first_name, ' ', last_name) AS full_name FROM actor;
-- 3. Selecciona los actores que su nombre empieza con "D"
Run | New Tab | JSON
SELECT first_name, last_name FROM actor WHERE first_name LIKE 'D%';
-- 4. ¿Tenemos algún acto
Run | New Tab | JSON
If you are an premium user, can show definition by hover
SELECT first_name, COUNT(*) FROM actor GROUP BY first_name HAVING COUNT(*) > 1; 6ms
-- 5. ¿Cuál es el costo máximo de renta de una película?
Run | New Tab | JSON
SELECT MAX(rental_rate) FROM film;
-- 6. ¿Cuáles son las peliculas que fueron rentadas con ese costo?
Run | New Tab | JSON
SELECT title FROM film WHERE rental_rate = (SELECT MAX(rental_rate) FROM film);
actor
Properties DATA Diagram Monitor Log
SELECT first_name, COUNT(*) FROM actor GROUP BY first_name HAVING COUNT(*) > 1 LIMIT 100
Search results
Export
Cost: 16ms
1
Total 52
first_name varchar(45)
count
Filter
Filter
Christian 3
Kenneth 4
Cameron 3
Spencer 2
Humphrey 2
Audrey 2
Fay 3
Susan 2
Michael 2
Burt 3
Mary 2
WSL: Ubuntu-20.04
main*+
1 3
0
127.0.0.1 dvdrental
You, 5 minutes ago
Ln 36, Col 9
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

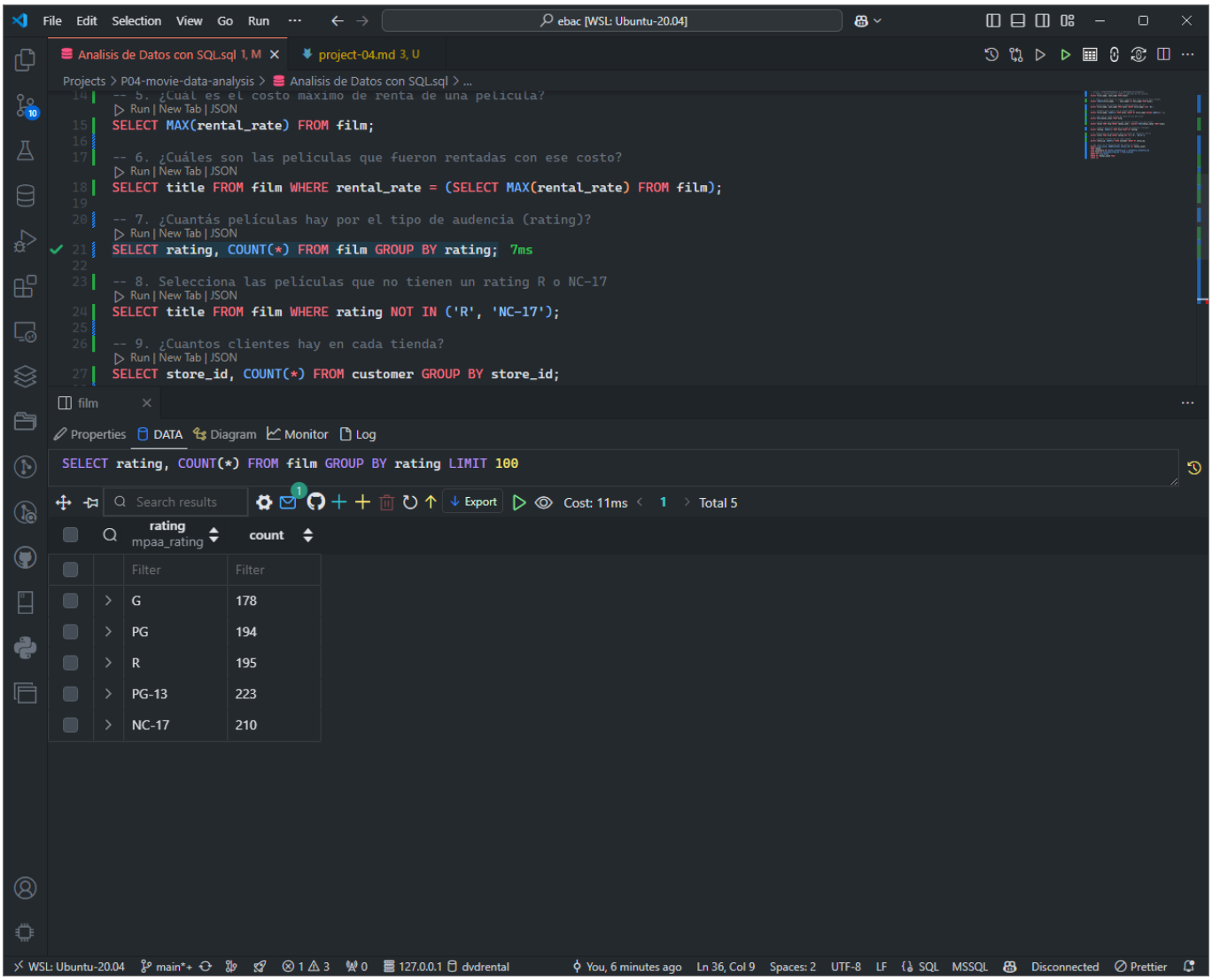
Exercise 5


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 3. Selecciona los actores que su nombre empieza con "D"
Run | New Tab | JSON
SELECT first_name, last_name FROM actor WHERE first_name LIKE 'D%';
-- 4. ¿Tenemos algún actor con el mismo nombre?
Run | New Tab | JSON
SELECT first_name, COUNT(*) FROM actor GROUP BY first_name HAVING COUNT(*) > 1;
-- 5. ¿Cuál es el costo máximo de renta de una película?
Run | New Tab | JSON
SELECT MAX(rental_rate) FROM film; 12ms
-- 6. ¿Cuáles son las películas que fueron rentadas con ese costo?
Run | New Tab | JSON
SELECT title FROM film WHERE rental_rate = (SELECT MAX(rental_rate) FROM film);
-- 7. ¿Cuantás películas hay por el tipo de audencia (rating)?
Run | New Tab | JSON
film
Properties DATA Diagram Monitor Log
SELECT MAX(rental_rate) FROM film LIMIT 100
Search results
Export
Cost: 58ms
1
Total 1
max
Filter
4.99
WSL: Ubuntu-20.04
main*+
1 3
0
127.0.0.1 dvdrental
You, 5 minutes ago
Ln 36, Col 9
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

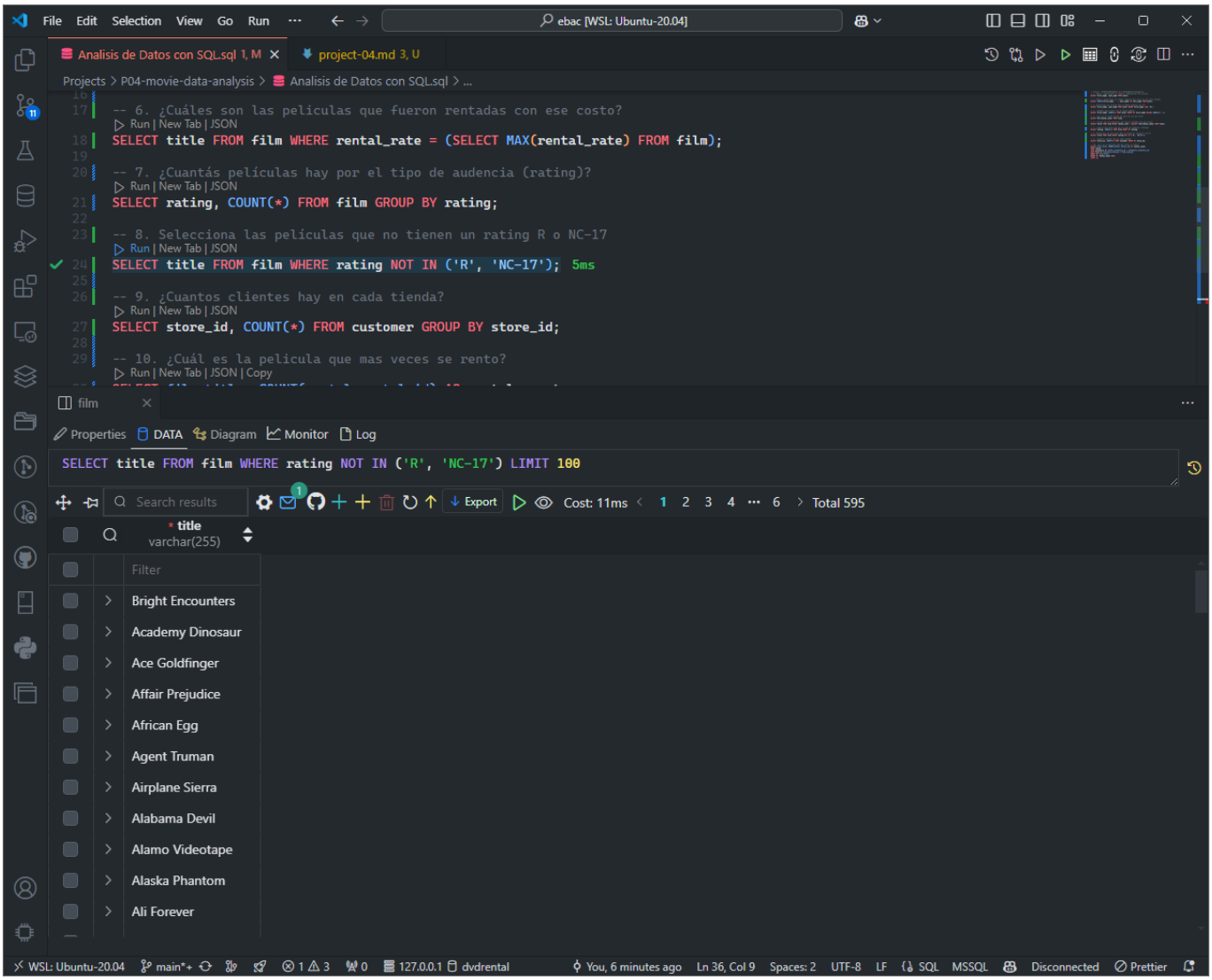
Exercise 6


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 5. ¿Cual es el costo maximo de renta de una pelicula?
Run | New Tab | JSON
SELECT MAX(rental_rate) FROM film;
-- 6. ¿Cuáles son las peliculas que fueron rentadas con ese costo?
Run | New Tab | JSON
SELECT title FROM film WHERE rental_rate = (SELECT MAX(rental_rate) FROM film); 10ms
-- 7. ¿Cuantás películas hay por el tipo de audencia (rating)?
Run | New Tab | JSON
SELECT rating, COUNT(*) FROM film GROUP BY rating;
-- 8. Selecciona las películas que no tienen un rating R o NC-17
Run | New Tab | JSON
SELECT title FROM film WHERE rating NOT IN ('R', 'NC-17');
-- 9. ¿Cuantos clientes hay en cada tienda?
Run | New Tab | JSON
SELECT store_id, COUNT(*) FROM customer GROUP BY store_id;
Result(RO)
Properties DATA Diagram Monitor Log
SELECT title FROM film WHERE rental_rate = (SELECT MAX(rental_rate) FROM film) LIMIT 100
Search results
Export
Cost: 10ms
1 2 3 4
Total 336
title
Filter
Chamber Italian
Grosse Wonderful
Airport Pollock
Bright Encounters
Ace Goldfinger
Airplane Sierra
Aladdin Calendar
Ali Forever
Amelie Hellfighters
American Circus
Anthem Luke
WSL: Ubuntu-20.04
main*+
127.0.0.1 dvdrental
You, 6 minutes ago
Ln 36, Col 9
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

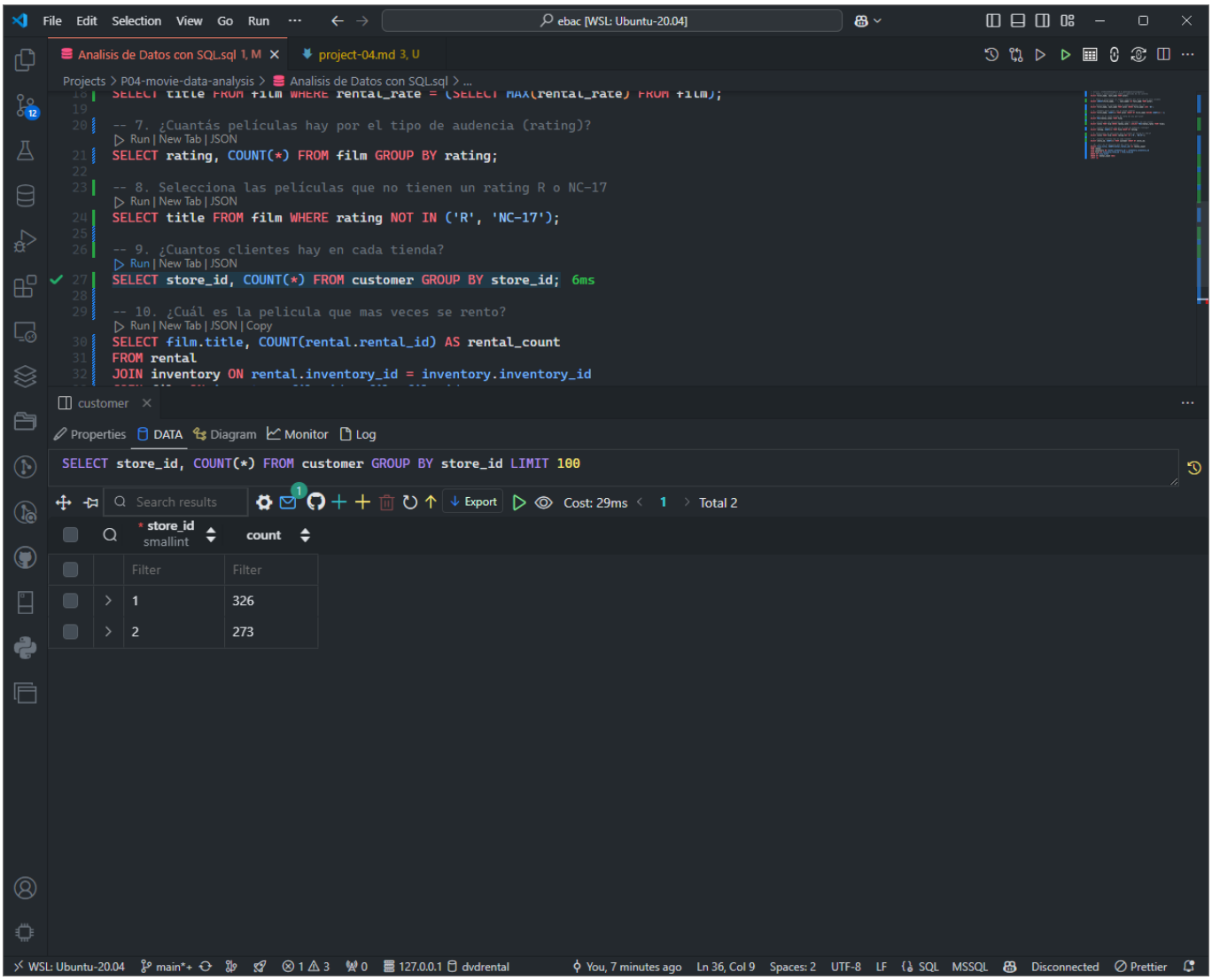
Exercise 7


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 5. ¿Cual es el costo maximo de renta de una pelicula?
Run | New Tab | JSON
SELECT MAX(rental_rate) FROM film;
-- 6. ¿Cuáles son las peliculas que fueron rentadas con ese costo?
Run | New Tab | JSON
SELECT title FROM film WHERE rental_rate = (SELECT MAX(rental_rate) FROM film);
-- 7. ¿Cuantás películas hay por el tipo de audencia (rating)?
Run | New Tab | JSON
SELECT rating, COUNT(*) FROM film GROUP BY rating; 7ms
-- 8. Selecciona las películas que no tienen un rating R o NC-17
Run | New Tab | JSON
SELECT title FROM film WHERE rating NOT IN ('R', 'NC-17');
-- 9. ¿Cuantos clientes hay en cada tienda?
Run | New Tab | JSON
SELECT store_id, COUNT(*) FROM customer GROUP BY store_id;
film
Properties DATA Diagram Monitor Log
SELECT rating, COUNT(*) FROM film GROUP BY rating LIMIT 100
Search results
Export
Cost: 11ms
1
Total 5
rating
mpaa_rating
count
Filter
Filter
G 178
PG 194
R 195
PG-13 223
NC-17 210
WSL: Ubuntu-20.04
main*+
127.0.0.1
dvdrental
You, 6 minutes ago
Ln 36, Col 9
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

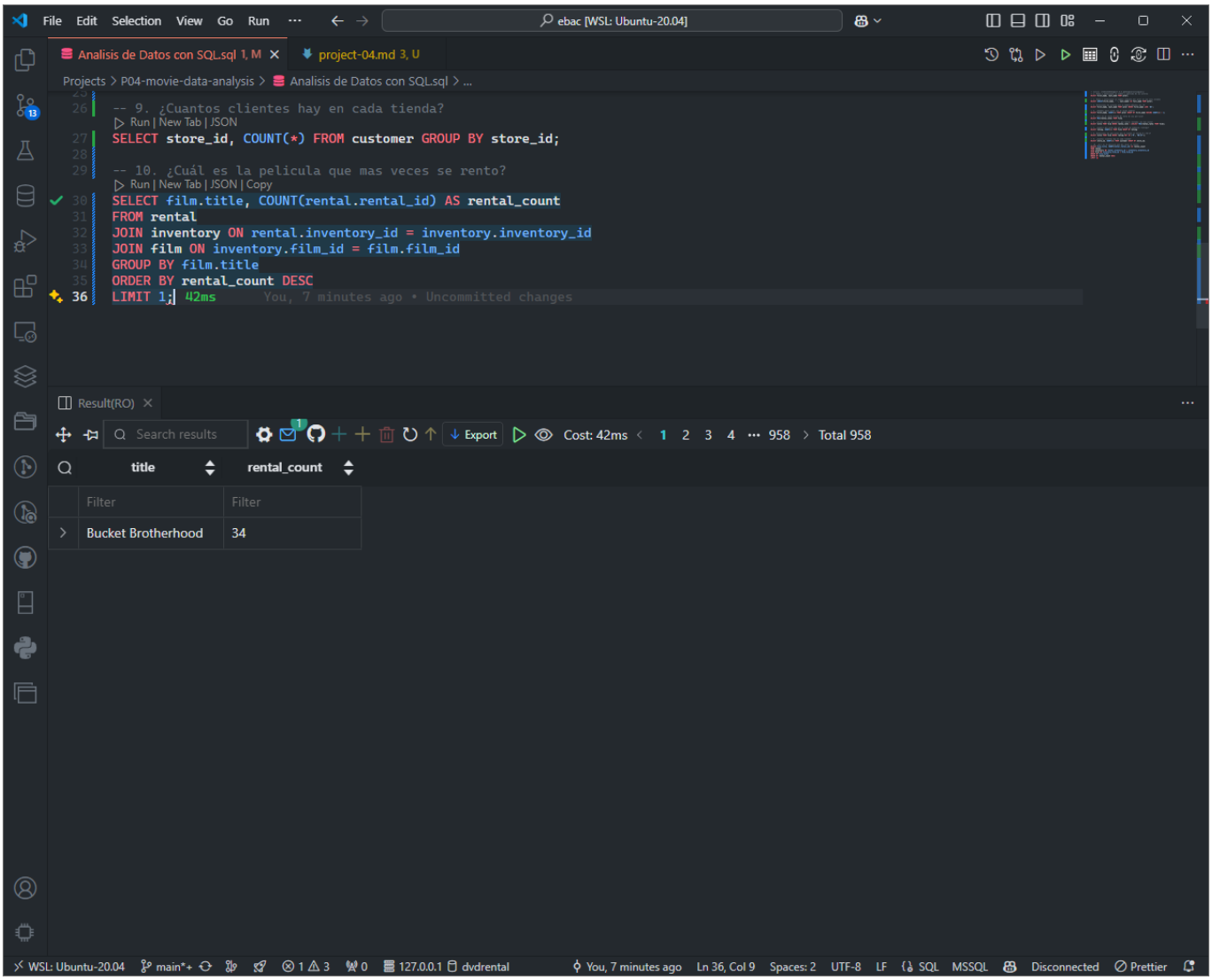
Exercise 8


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 6. ¿Cuáles son las peliculas que fueron rentadas con ese costo?
Run | New Tab | JSON
SELECT title FROM film WHERE rental_rate = (SELECT MAX(rental_rate) FROM film);
-- 7. ¿Cuantás películas hay por el tipo de audencia (rating)?
Run | New Tab | JSON
SELECT rating, COUNT(*) FROM film GROUP BY rating;
-- 8. Selecciona las películas que no tienen un rating R o NC-17
Run | New Tab | JSON
SELECT title FROM film WHERE rating NOT IN ('R', 'NC-17'); 5ms
-- 9. ¿Cuantos clientes hay en cada tienda?
Run | New Tab | JSON
SELECT store_id, COUNT(*) FROM customer GROUP BY store_id;
-- 10. ¿Cuál es la película que mas veces se rento?
Run | New Tab | JSON | Copy
film
Properties DATA Diagram Monitor Log
SELECT title FROM film WHERE rating NOT IN ('R', 'NC-17') LIMIT 100
Search results
Export
Cost: 11ms
1 2 3 4 6
Total 595
title
varchar(255)
Filter
Bright Encounters
Academy Dinosaur
Ace Goldfinger
Affair Prejudice
African Egg
Agent Truman
Airplane Sierra
Alabama Devil
Alamo Videotape
Alaska Phantom
Ali Forever
WSL: Ubuntu-20.04
main*+
127.0.0.1 dvdrental
You, 6 minutes ago
Ln 36, Col 9
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

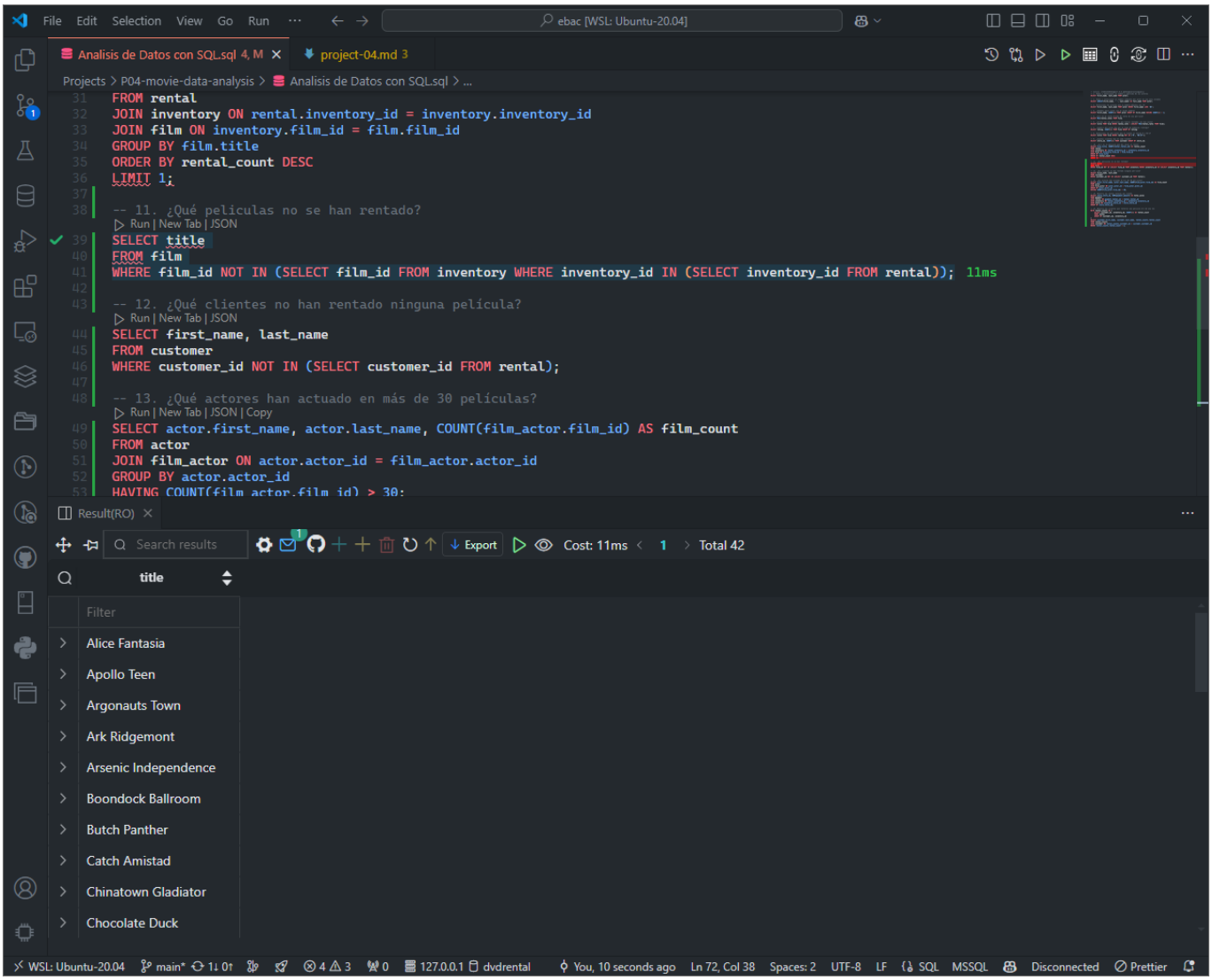
Exercise 9


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 7. ¿Cuantás películas hay por el tipo de audencia (rating)?
Run | New Tab | JSON
SELECT rating, COUNT(*) FROM film GROUP BY rating;
-- 8. Selecciona las películas que no tienen un rating R o NC-17
Run | New Tab | JSON
SELECT title FROM film WHERE rating NOT IN ('R', 'NC-17');
-- 9. ¿Cuantos clientes hay en cada tienda?
Run | New Tab | JSON
SELECT store_id, COUNT(*) FROM customer GROUP BY store_id; 6ms
-- 10. ¿Cuál es la película que mas veces se rento?
Run | New Tab | JSON | Copy
SELECT film.title, COUNT(rental.rental_id) AS rental_count
FROM rental
JOIN inventory ON rental.inventory_id = inventory.inventory_id
customer
Properties DATA Diagram Monitor Log
SELECT store_id, COUNT(*) FROM customer GROUP BY store_id LIMIT 100
Search results
Export
Cost: 29ms
1
Total 2
store_id smallint
count
Filter
Filter
1 326
2 273
WSL: Ubuntu-20.04 main*+ 1 3 0 127.0.0.1 dvdrental
You, 7 minutes ago Ln 36, Col 9 Spaces: 2 UTF-8 LF SQL MSSQL Disconnected Prettier

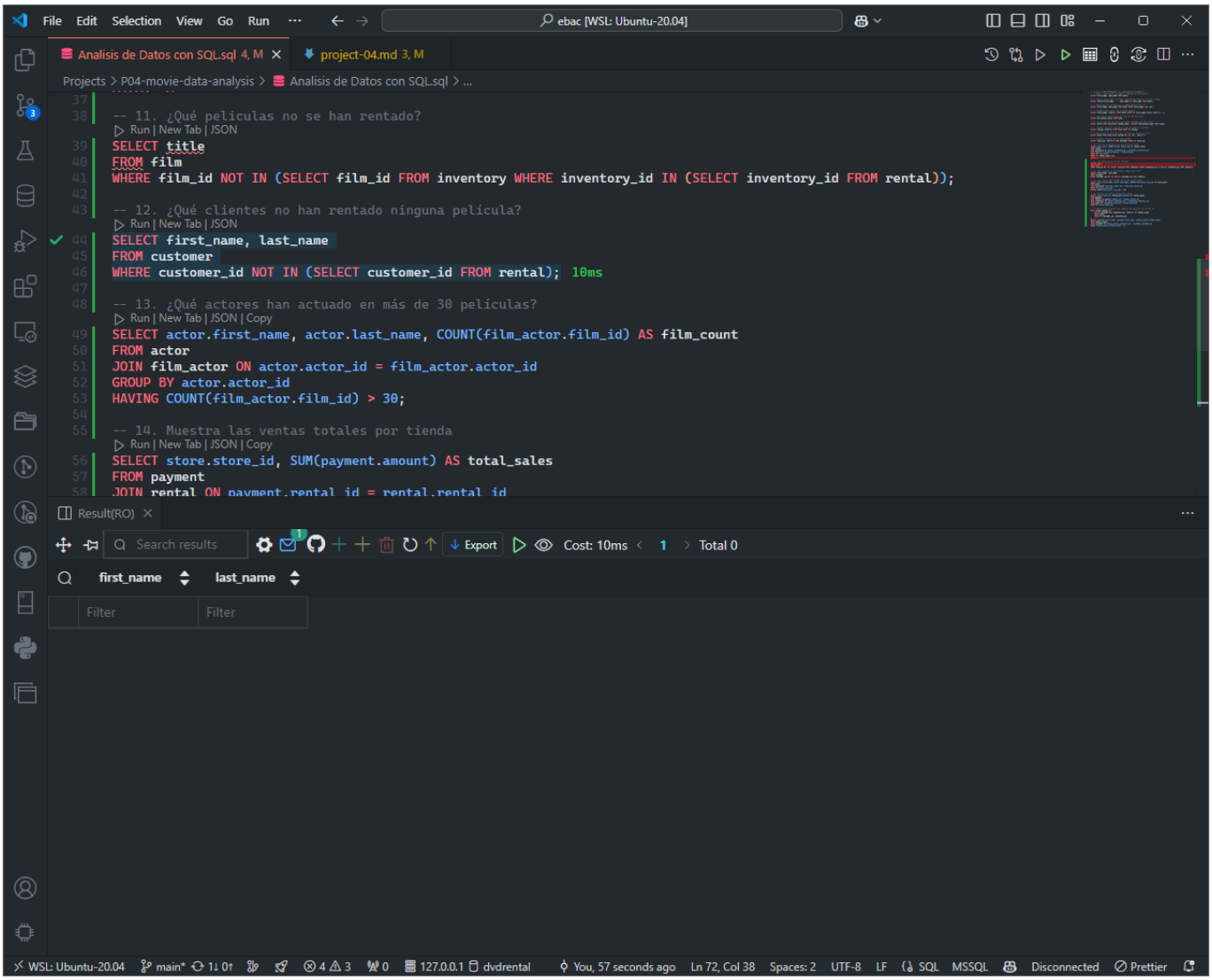
Exercise 10


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 1, M
project-04.md 3, U
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 9. ¿Cuantos clientes hay en cada tienda?
Run | New Tab | JSON
SELECT store_id, COUNT(*) FROM customer GROUP BY store_id;
-- 10. ¿Cuál es la pelicula que mas veces se rento?
Run | New Tab | JSON | Copy
SELECT film.title, COUNT(rental.rental_id) AS rental_count
FROM rental
JOIN inventory ON rental.inventory_id = inventory.inventory_id
JOIN film ON inventory.film_id = film.film_id
GROUP BY film.title
ORDER BY rental_count DESC
LIMIT 1; 42ms
You, 7 minutes ago • Uncommitted changes
Result(RO)
Search results
Export
Cost: 42ms
1 2 3 4 ... 958 Total 958
title
rental_count
Filter
Filter
Bucket Brotherhood
34
WSL: Ubuntu-20.04
main*+
1 3
0
127.0.0.1
dvdrental
You, 7 minutes ago
Ln 36, Col 9
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

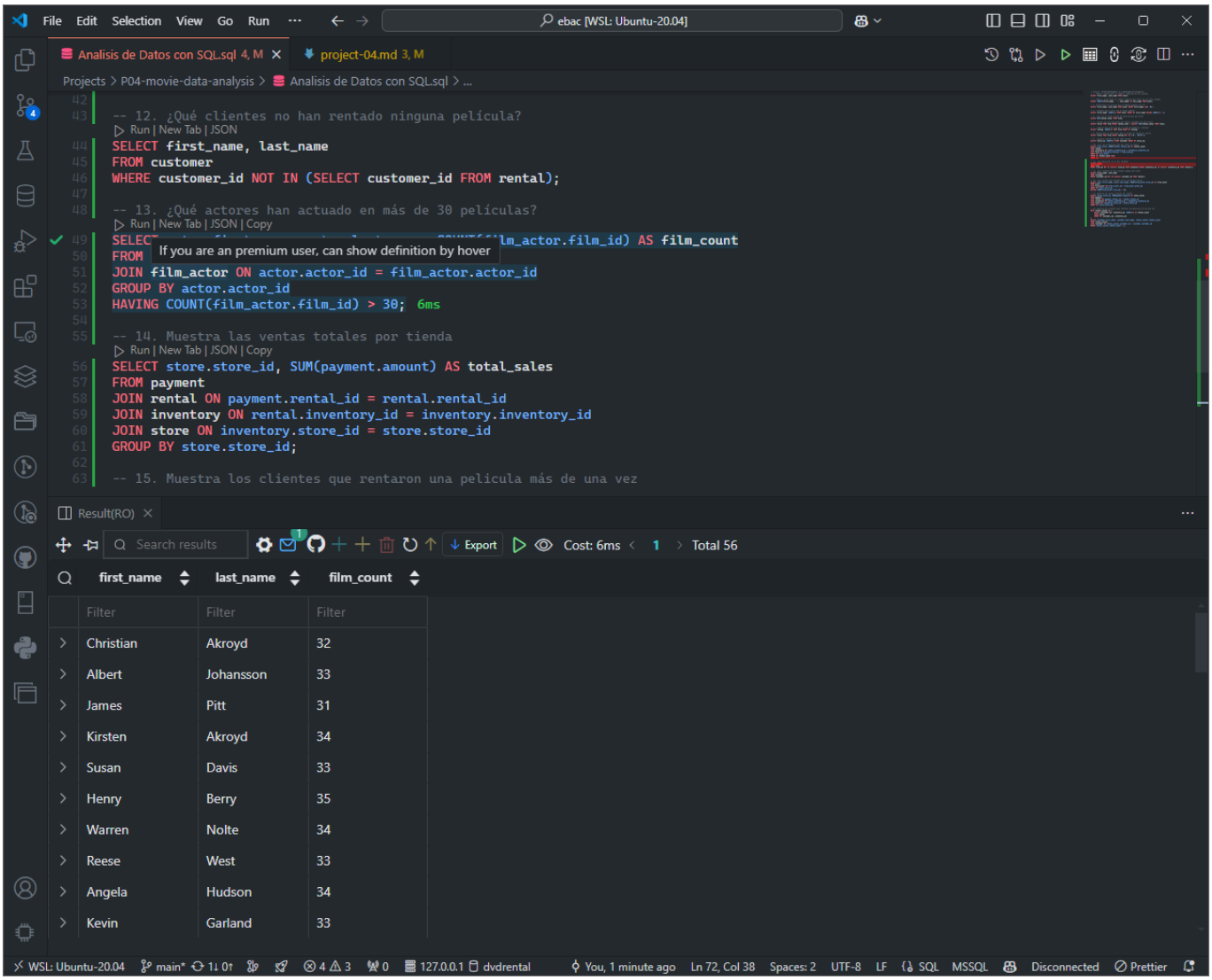
Exercise 11


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 4, M
project-04.md 3
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
FROM rental
JOIN inventory ON rental.inventory_id = inventory.inventory_id
JOIN film ON inventory.film_id = film.film_id
GROUP BY film.title
ORDER BY rental_count DESC
LIMIT 1;
-- 11. ¿Qué peliculas no se han rentado?
Run | New Tab | JSON
SELECT title
FROM film
WHERE film_id NOT IN (SELECT film_id FROM inventory WHERE inventory_id IN (SELECT inventory_id FROM rental)); 11ms
-- 12. ¿Qué clientes no han rentado ninguna película?
Run | New Tab | JSON
SELECT first_name, last_name
FROM customer
WHERE customer_id NOT IN (SELECT customer_id FROM rental);
-- 13. ¿Qué actores han actuado en más de 30 películas?
Run | New Tab | JSON | Copy
SELECT actor.first_name, actor.last_name, COUNT(film_actor.film_id) AS film_count
FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
GROUP BY actor.actor_id
HAVING COUNT(film_actor.film_id) > 30;
Result(RO)
Search results
Export
Cost: 11ms
1
Total 42
title
Filter
Alice Fantasia
Apollo Teen
Argonauts Town
Ark Ridgemont
Arsenic Independence
Boondock Ballroom
Butch Panther
Catch Amistad
Chinatown Gladiator
Chocolate Duck
WSL: Ubuntu-20.04
main*
4 3
0
127.0.0.1
dvdrental
You, 10 seconds ago
Ln 72, Col 38
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

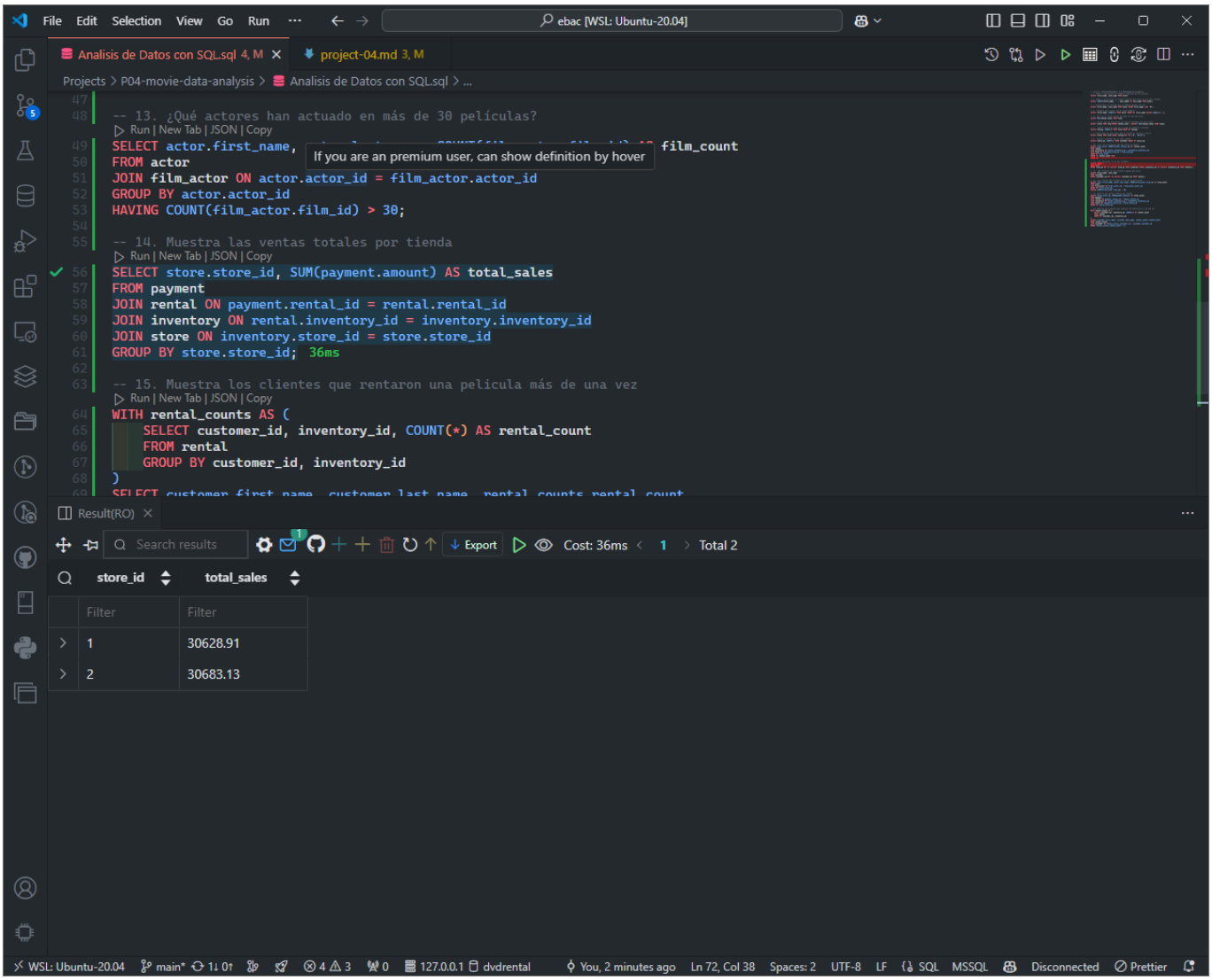
Exercise 12


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 4, M
project-04.md 3, M
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 11. ¿Qué peliculas no se han rentado?
Run | New Tab | JSON
SELECT title
FROM film
WHERE film_id NOT IN (SELECT film_id FROM inventory WHERE inventory_id IN (SELECT inventory_id FROM rental));
-- 12. ¿Qué clientes no han rentado ninguna película?
Run | New Tab | JSON
SELECT first_name, last_name
FROM customer
WHERE customer_id NOT IN (SELECT customer_id FROM rental); 10ms
-- 13. ¿Qué actores han actuado en más de 30 películas?
Run | New Tab | JSON | Copy
SELECT actor.first_name, actor.last_name, COUNT(film_actor.film_id) AS film_count
FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
GROUP BY actor.actor_id
HAVING COUNT(film_actor.film_id) > 30;
-- 14. Muestra las ventas totales por tienda
Run | New Tab | JSON | Copy
SELECT store.store_id, SUM(payment.amount) AS total_sales
FROM payment
Result(RO)
Search results
Export
Cost: 10ms
1
Total 0
first_name
last_name
Filter
Filter
WSL: Ubuntu-20.04
main*
Ln 72, Col 38
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

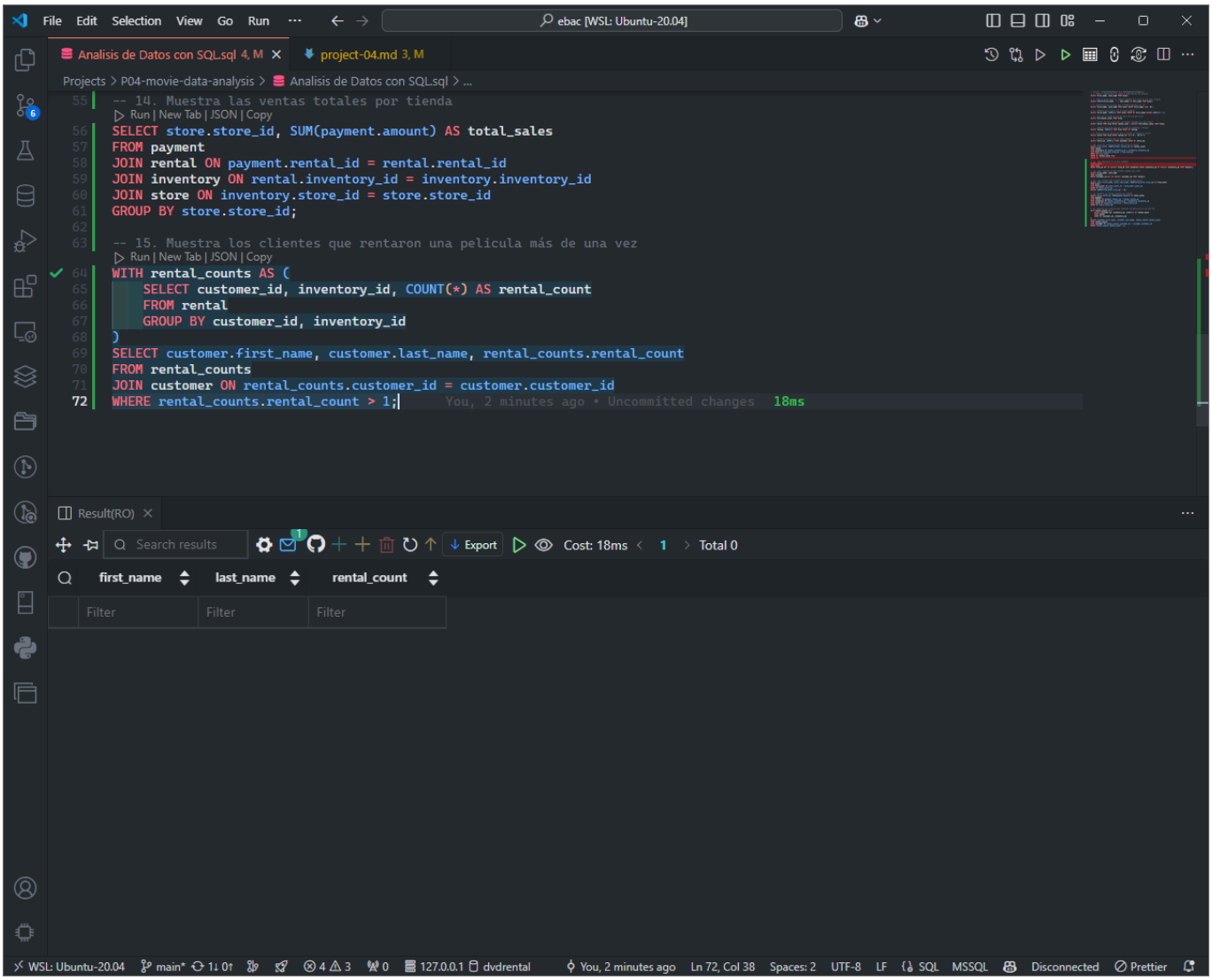
Exercise 13


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 4, M
project-04.md 3, M
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 12. ¿Qué clientes no han rentado ninguna película?
Run | New Tab | JSON
SELECT first_name, last_name
FROM customer
WHERE customer_id NOT IN (SELECT customer_id FROM rental);
-- 13. ¿Qué actores han actuado en más de 30 películas?
Run | New Tab | JSON | Copy
If you are an premium user, can show definition by hover
JOIN film_actor ON actor.actor_id = film_actor.actor_id
GROUP BY actor.actor_id
HAVING COUNT(film_actor.film_id) > 30; 6ms
-- 14. Muestra las ventas totales por tienda
Run | New Tab | JSON | Copy
SELECT store.store_id, SUM(payment.amount) AS total_sales
FROM payment
JOIN rental ON payment.rental_id = rental.rental_id
JOIN inventory ON rental.inventory_id = inventory.inventory_id
JOIN store ON inventory.store_id = store.store_id
GROUP BY store.store_id;
-- 15. Muestra los clientes que rentaron una pelicula más de una vez
Result(RO)
Search results
Export
Cost: 6ms
1
Total 56
first_name
last_name
film_count
Filter
Christian Akroyd 32
Albert Johansson 33
James Pitt 31
Kirsten Akroyd 34
Susan Davis 33
Henry Berry 35
Warren Nolte 34
Reese West 33
Angela Hudson 34
Kevin Garland 33
WSL: Ubuntu-20.04
main*
You, 1 minute ago
Ln 72, Col 38
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

Exercise 14


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 4, M
project-04.md 3, M
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 13. ¿Qué actores han actuado en más de 30 películas?
Run | New Tab | JSON | Copy
SELECT actor.first_name, film_count
FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
GROUP BY actor.actor_id
HAVING COUNT(film_actor.film_id) > 30;
If you are an premium user, can show definition by hover
-- 14. Muestra las ventas totales por tienda
Run | New Tab | JSON | Copy
SELECT store.store_id, SUM(payment.amount) AS total_sales
FROM payment
JOIN rental ON payment.rental_id = rental.rental_id
JOIN inventory ON rental.inventory_id = inventory.inventory_id
JOIN store ON inventory.store_id = store.store_id
GROUP BY store.store_id; 36ms
-- 15. Muestra los clientes que rentaron una pelicula más de una vez
Run | New Tab | JSON | Copy
WITH rental_counts AS (
SELECT customer_id, inventory_id, COUNT(*) AS rental_count
FROM rental
GROUP BY customer_id, inventory_id
)
Result(RO)
Search results
Export
Cost: 36ms
1
Total 2
store_id
total_sales
Filter
Filter
1
30628.91
2
30683.13
WSL: Ubuntu-20.04
main*
You, 2 minutes ago
Ln 72, Col 38
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier

Exercise 15


File Edit Selection View Go Run
ebac [WSL: Ubuntu-20.04]
Analisis de Datos con SQL.sql 4, M
project-04.md 3, M
Projects > P04-movie-data-analysis > Analisis de Datos con SQL.sql > ...
-- 14. Muestra las ventas totales por tienda
Run | New Tab | JSON | Copy
SELECT store.store_id, SUM(payment.amount) AS total_sales
FROM payment
JOIN rental ON payment.rental_id = rental.rental_id
JOIN inventory ON rental.inventory_id = inventory.inventory_id
JOIN store ON inventory.store_id = store.store_id
GROUP BY store.store_id;
-- 15. Muestra los clientes que rentaron una pelicula más de una vez
Run | New Tab | JSON | Copy
WITH rental_counts AS (
SELECT customer_id, inventory_id, COUNT(*) AS rental_count
FROM rental
GROUP BY customer_id, inventory_id
)
SELECT customer.first_name, customer.last_name, rental_counts.rental_count
FROM rental_counts
JOIN customer ON rental_counts.customer_id = customer.customer_id
WHERE rental_counts.rental_count > 1;
You, 2 minutes ago • Uncommitted changes
18ms
Result(RO)
Search results
Export
Cost: 18ms
1
Total 0
first_name
last_name
rental_count
Filter
Filter
Filter
WSL: Ubuntu-20.04
main*
You, 2 minutes ago
Ln 72, Col 38
Spaces: 2
UTF-8
LF
SQL
MSSQL
Disconnected
Prettier