SQL-HandsOn2

# Alberta “Albi” Kovatcheva

## Part 1

1. Write a query to find the first and last name, customer ID and rental ID for customers who have rented a film.

Query:

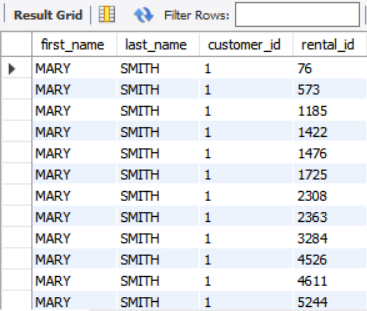
SELECT first\_name, last\_name, customer\_id, rental\_id

FROM sakila.customer

INNER JOIN sakila.rental

USING (customer\_id);

Results (1000 rows returned):



1. Write a query that finds all films with actors that have an actor\_id 5.

Query:

SELECT actor\_id, first\_name, last\_name, film\_id, title

FROM sakila.actor

INNER JOIN sakila.film\_actor

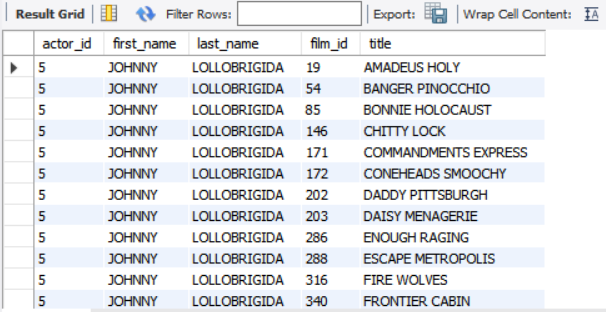
USING(actor\_id)

INNER JOIN sakila.film

USING(film\_id)

WHERE actor\_id = 5;

Results (29 rows returned):



1. Write a query that lists out all information of every film along with the name of the language for each film, even if a language does not exist for that film.

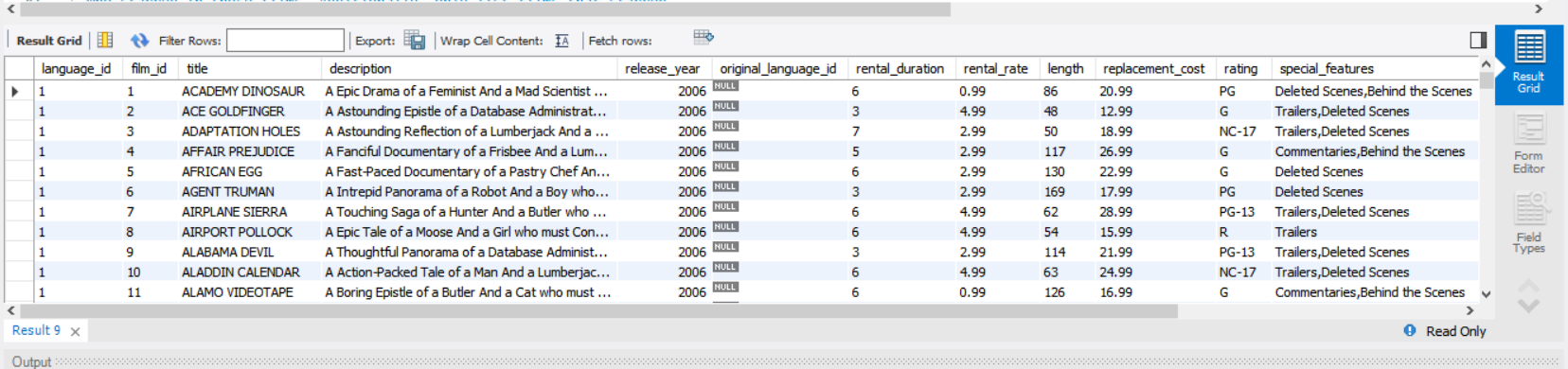
Query:

SELECT \* FROM sakila.film

LEFT OUTER JOIN sakila.language

USING (language\_id);

Results (1000 rows returned):



1. Write a query that lists out the title of films and the name of the actors who starred in those films. Additionally, only list films that starred artists whose first names start with a vowel.

Query:

SELECT title, first\_name, last\_name FROM sakila.film

JOIN sakila.film\_actor

USING (film\_id)

JOIN sakila.actor

USING (actor\_id)

WHERE first\_name LIKE "a%"

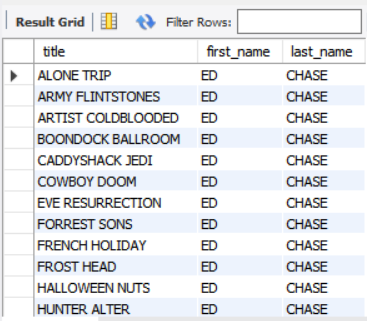
OR first\_name LIKE "e%"

OR first\_name LIKE "i%"

OR first\_name LIKE "o"

OR first\_name LIKE "u";

Results (577 rows returned):



## Part 2

1. You have just been hired as a Data Analyst for a company that rents films to customers. They would like an inventory list of films that were rented for more than $4.99.

Query:

SELECT film\_id, title, rental\_rate

FROM sakila.film

JOIN sakila.inventory

USING(film\_id)

WHERE rental\_rate > 4.99;

Results (0 rows returned):

