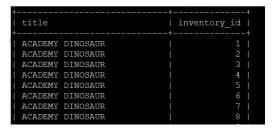
Joins Exercise 6-1

Using sakila database - Checkpoint Lab

- 1. Get a list of all film titles and their inventory number even those we don't have.
 - > You might like to limit the output to 200 to check you have everything looking like this



select film.title, inventory.inventory id from film join inventory on film.film id=inventory.film id LIMIT 200;

- 2. Which films do we not have in stock?
 - ➤ There are 42 records beginning with:



MariaDB [sakila]> select film.title from film left join inventory on film.film_id=inventory.film_id where inventory.inventory_id IS NULL;

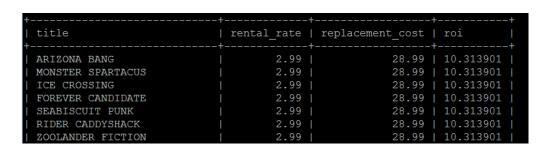
3. List the number of films in which each actor has featured (sort the output in descending order of the number of films)

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 ORDER BY film_count DESC;

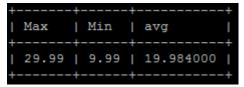


Techniques from this week

4. The store uses a formula to calculate the return-on-investment (or ROI) which is (rental_rate / replacement_cost * 100). List the films, rental replacement cost and ROI which have an ROI more than 10. Order by ROI. *Only have the formula once in the query*



5. List the maximum, minimum and average film replacement cost using subselects in the select clause only (do not use a FROM clause in the main query) – yes this is silly.



Student Database on SQLite

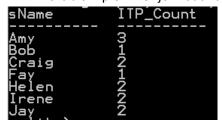
Techniques from earlier this week

Not trivial, you'll have to work on these.

- 6. List the students as pairs who come from the same sized high school. Order by school size.
 - > Only list one pair of each student e.g. if you have Alice and Bob in a record don't also list Bob and Alice (unless they are different students we have two different AMY's).
 - ➤ Work through this in stages removing redundant pairs is the last step. You might like to display more information while developing the query (e.g. sid)



- 7. List each student that has made an application and the number of ITP's they have applied to.
 - This is a simple inner join but I did use something introduced in passing today to get this.

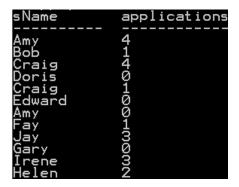


Subqueries and Outer Joins

8. Which students have not applied anywhere?



9. List a count of the number of applications made by each student



10. List the number of institutions that each student has applied to:



11. How many students have applied to each institution?

