1.Which actors have the first name ‘Scarlett’
Answer: select * from actor where first_name='Scarlett';

2.Which actors have the last name ‘Johansson’
Answer: select * from actor where last_name='Johansson';

3.How many distinct actors last names are there?
Answer:select count(distinct last_name) from actor;

4.Which last names are not repeated?
Answer:select last_name from actor group by last_name having count(last_name) = 1;

5.Which last names appear more than once?
Answer: select last_name FROM actor group by last_name having count(last_name) >1;

6.Which actor has appeared in the most films?
Answer: select actor_id, first_name, last_name,count(actor_id) from actor join film_actor using (actor_id)
group by actor_id order by count(actor_id) desc limit 1;

7.Is ‘Academy Dinosaur’ available for rent from Store 1?
Answer: select inventory.inventory_id from inventory join store using (store_id) join film using (film_id)
join rental using (inventory_id) where film.title = 'Academy Dinosaur' and store.store_id = 1
and not exists (select * from rental where rental.inventory_id = inventory.inventory_id and rental.return_date is null);

8.Insert a record to represent Mary Smith renting ‘Academy Dinosaur’ from Mike Hillyer at Store 1 today .
Answer: insert into rental (rental_date, inventory_id, customer_id, staff_id)values (now(), 1, 1, 1);

9.When is ‘Academy Dinosaur’ due?
Answer: select rental_date,rental_date + interval(select rental_duration from film where film_id = 1) day
as due_date from rental where rental_id = (select rental_id from rental order by rental_id desc limit 1);

10.What is that average running time of all the films in the sakila DB?
Answer: select avg(length) as running_time from film;

11.What is the average running time of films by category?
Answer:select category.name, avg(length) from film join film_category using (film_id) join category using (category_id)
group by category.name order by avg(length) desc;

12.Why does this query return the empty set?
Answer: select * from film natural join inventory;