**Queries**

1. What query would you run to get all the customers inside city\_id = 312? Your query should return customer first name, last name, email, and address.

**select customer.first\_name, customer.last\_name, customer.email, address.address, address.address2, address.postal\_code**

**from customer**

**join address on customer.address\_id = address.address\_id**

**where city\_id = 312;**

2. What query would you run to get all comedy films? Your query should return film title, description, release year, rating, special features, and genre (category).

**select film.title, film.description, film.release\_year, film.rating, film.special\_features, category.name**

**from film**

**join film\_category on film.film\_id = category.category\_id**

**where category.name = "Comedy";**

3. What query would you run to get all the films joined by actor\_id=5? Your query should return the actor id, actor name, film title, description, and release year.

**select actor.actor\_id, concat(actor.first\_name, ' ', actor.last\_name) as actor, title, film.description, film.release\_year**

**from film**

**join film\_actor on film.film\_id = film\_actor.film\_id**

**join actor on film\_actor.actor\_id = actor.actor\_id**

**where actor.actor\_id = 5;**

4. What query would you run to get all the customers in store\_id = 1 and inside these cities (1, 42, 312 and 459)? Your query should return customer first name, last name, email, and address.

**select customer.first\_name, customer.last\_name, customer.email, address.address, address.address2**

**from customer**

**join address on customer.address\_id = address.address\_id**

**join city on address.address\_id = city.city\_id**

**where customer.store\_id = 1 and city.city\_id in (1,42,312,459)**

5. What query would you run to get all the films with a "rating = G" and "special feature = behind the scenes", joined by actor\_id = 15? Your query should return the film title, description, release year, rating, and special feature. *Hint: You may use LIKE function in getting the 'behind the scenes' part.*

**select film.title, film.description, film.release\_year, film.rating, film.special\_features**

**from film**

**join film\_actor on film.film\_id = film\_actor.film\_id**

**join actor on film\_actor.film\_id = actor.actor\_id**

**where film.rating = "G" and film.special\_features like "%behind the scenes%" and actor.actor\_id = 15;**

6. What query would you run to get all the actors that joined in the film\_id = 369? Your query should return the film\_id, title, actor\_id, and actor\_name.

**select film.title, film.film\_id, actor.actor\_id, concat(actor.first\_name, " ", actor.last\_name) as actor\_name**

**from film**

**join film\_actor on film.film\_id = film\_actor.film\_id**

**join actor on film\_actor.actor\_id = actor.actor\_id**

**where film.film\_id = 369;**

7. What query would you run to get all drama films with a rental rate of 2.99? Your query should return film title, description, release year, rating, special features, and genre (category).

**select film.title, film.description, film.release\_year, film.rating, film.special\_features, film.rental\_rate, category.name**

**from film**

**join film\_category on film.film\_id = film\_category.film\_id**

**join category on film\_category.category\_id = category.category\_id**

**where category.name ="Drama" and film.rental\_rate = 2.99;**

8. What query would you run to get all the action films which are joined by SANDRA KILMER? Your query should return film title, description, release year, rating, special features, genre (category), and actor's first name and last name.

**select film.title, film.description, film.release\_year, film.rating, film.special\_features, category.name, actor.first\_name, actor.last\_name**

**from actor**

**join film\_actor on actor.actor\_id = film\_actor.actor\_id**

**join film on film\_actor.film\_id = film.film\_id**

**join film\_category on film.film\_id = film\_category.film\_id**

**join category on film\_category.category\_id = category.category\_id**

**where actor.first\_name = "Sandra" and actor.last\_name = "Kilmer" and category.name ="Action";**