

Bases de données

Note importante : les requêtes proposées ici sont *une* solution possible parmi d'autres, il existe souvent de nombreuses requêtes permettant de répondre à une question donnée.

1.

```
SELECT title, description FROM film WHERE release_year=2000
```

2.

```
SELECT * FROM film WHERE length <= 50
```

3.

```
SELECT title FROM film WHERE title LIKE 'J%'
```

4.

```
SELECT title, description FROM film  
WHERE special_features LIKE '%Deleted Scenes%'
```

5.

```
SELECT AVG(replacement_cost) FROM film  
WHERE description LIKE "%crocodile%"
```

6.

```
SELECT title FROM film ORDER BY release_year LIMIT 10
```

7.

```
SELECT rating, COUNT(*) FROM film GROUP BY rating
```

8.

```
SELECT release_year, COUNT(*) FROM film  
GROUP BY release_year ORDER BY COUNT(*) DESC
```

9.

```
SELECT COUNT(*) FROM film  
JOIN language ON film.language_id = language.language_id  
WHERE name="French"
```

10.

```
SELECT * FROM film  
JOIN film_category ON film.film_id = film_category.film_id  
JOIN category ON film_category.category_id = category.category_id  
WHERE name="Comedy" ORDER BY length DESC LIMIT 1
```

11.

```
SELECT 1.0 * COUNT(*) / COUNT(DISTINCT film_id) FROM film_actor
```

12.

```
SELECT title FROM rental  
JOIN inventory ON rental.inventory_id = inventory.inventory_id  
JOIN film ON inventory.film_id = film.film_id  
GROUP BY customer_id, film.film_id, title  
HAVING COUNT(*)>=3
```

13.

```
SELECT SUM(replacement_cost) FROM film  
JOIN inventory ON film.film_id = inventory.film_id  
WHERE store_id = 1
```

14.

```
SELECT DISTINCT customer.first_name, customer.last_name FROM customer  
JOIN actor ON customer.first_name = actor.first_name  
AND customer.last_name = actor.last_name
```

Il est également possible d'utiliser deux requêtes et INTERSECT!

15.

```
SELECT address, city.city FROM address  
JOIN city ON address.city_id = city.city_id  
JOIN country ON city.country_id = country.country_id  
WHERE country = "Argentina"
```

16.

```
SELECT MIN(release_year - birth_year) FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
JOIN film ON film_actor.film_id = film.film_id
```

17.

```
SELECT first_name, last_name FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
GROUP BY actor.actor_id, actor.first_name, actor.last_name
ORDER BY COUNT(*) DESC LIMIT 5
```

18.

```
SELECT title, AVG(release_year - birth_year) FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
JOIN film ON film_actor.film_id = film.film_id
GROUP BY film.film_id, title
ORDER BY AVG(release_year - birth_year) DESC LIMIT 10
```

19.

```
SELECT first_name, last_name, address, postal_code, city, country
FROM staff JOIN address ON staff.address_id = address.address_id
JOIN city ON address.city_id = city.city_id
JOIN country ON city.country_id = country.country_id
JOIN store ON staff_id = manager_staff_id WHERE store_id = 2
```

20.

```
SELECT DISTINCT first_name, last_name FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
JOIN film_category ON film_actor.film_id = film_category.film_id
JOIN category ON film_category.category_id = category.category_id
WHERE name = "Horror" AND last_name LIKE "A%"
```

21.

```
SELECT first_name, last_name FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
JOIN film_category ON film_actor.film_id = film_category.film_id
GROUP BY actor.actor_id, actor.first_name, actor.last_name
ORDER BY COUNT(DISTINCT category_id) DESC LIMIT 1
```

22.

```
SELECT first_name, last_name, address, postal_code, city, country
FROM customer
JOIN address ON customer.address_id = address.address_id
JOIN city ON address.city_id = city.city_id
JOIN country ON city.country_id = country.country_id
WHERE customer_id = (SELECT customer_id FROM payment
GROUP BY customer_id
ORDER BY SUM(amount) DESC LIMIT 1)
```

23.

```
SELECT DISTINCT first_name, last_name FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
WHERE (first_name != "RIP" OR last_name != "CRAWFORD")
AND film_id IN
(SELECT film.film_id FROM film
JOIN film_actor ON film.film_id = film_actor.film_id
JOIN actor ON film_actor.actor_id = actor.actor_id
JOIN film_category ON film.film_id = film_category.film_id
JOIN category ON film_category.category_id = category.category_id
WHERE first_name = "RIP"
AND last_name = "CRAWFORD" AND name = "Drama")
```

24.

```
SELECT first_name, last_name, address, postal_code, city, country
FROM customer
JOIN address ON customer.address_id = address.address_id
JOIN city ON address.city_id = city.city_id
JOIN country ON city.country_id = country.country_id
WHERE address.city_id IN
(SELECT address.city_id FROM customer
JOIN address ON customer.address_id = address.address_id
GROUP BY address.city_id
HAVING COUNT(*) >= 2)
```