

-- EXERCÍCIOS PÁGINAS 40, 41, 42 e 43

-- Pedro Mattar, 3C2

-- PÁGINA 40

use sakila;

-- QUESTÃO 2

select * from country inner join city on (country.country_id = city.country_id);

-- A)

select * from city inner join address on (city.city_id = address.city_id);

-- B)

select * from address inner join customer on (address.address_id = customer.address_id);

-- C)

select * from customer inner join payment on (customer.customer_id =
payment.customer_id);

-- D)

select * from payment inner join customer on (payment.customer_id =
customer.customer_id);

-- E)

select * from film inner join language on (film.language_id = language.language_id);

-- PÁGINA 41

-- QUESTÃO 3

-- A)

select count(film_id) as NumeroDeFilmes, name as idioma from film inner join language on
(language.language_id= film.language_id) group by name;

-- B)

select count(payment_id) as TotalPagamentos, store.store_id as loja from store inner join
staff on (store.store_id = staff.store_id) inner join payment on (staff.staff_id =
payment.staff_id) group by store.store_id;

-- C)

```
select count(customer_id) as QuantidadeClientes, store.store_id as loja from store inner join
customer on (store.store_id = customer.store_id) group by store.store_id;
```

-- D)

```
select store.store_id, avg(payment.amount) as Media, sum(payment.amount) as Total,
count(payment.payment_id) as QuantidadePagamento, max(payment.amount) as Maximo,
min(payment.amount) as Minimo from store
inner join staff on (store.store_id = staff.store_id) inner join payment on (staff.staff_id =
payment.staff_id) group by store.store_id;
```

-- E)

```
select customer.customer_id, sum(payment.amount) as TotalPagamentos from customer
inner join payment on (customer.customer_id = payment.customer_id) group by
customer.customer_id;
```

-- F)

```
select language.name as idioma, count(film.film_id) as QuantidadeFilme from film inner join
language on (film.language_id = language.language_id) where film.length between 100 and
150 group by language.name;
```

-- G)

```
select store.store_id, sum(payment.amount) as TotalPagamentos from store inner join staff
on (store.store_id = staff.store_id) inner join payment on (staff.staff_id = payment.staff_id)
where payment.payment_date like "2005-08%" and "2005-09%" group by store.store_id;
```

-- H)

```
select store.store_id, count(customer.customer_id) as QuantidadeClientes from store inner
join customer on (store.store_id = customer.store_id) where customer.Last_name like "R%"
group by store.store_id;
```

-- PÁGINA 42

-- QUESTÃO 4

-- A)

```
select * from city inner join address on (city.city_id = address.city_id) inner join customer on
(address.address_id = customer.address_id);
```

-- B)

```
select * from customer inner join payment on (customer.customer_id =  
payment.customer_id) inner join rental on (customer.customer_id = rental.customer_id);
```

-- C)

```
select * from film inner join film_category on (film.film_id = film_category.film_id) inner join  
category on (film_category.category_id = category.category_id);
```

-- D)

```
select * from actor inner join film_actor on (actor.actor_id = film_actor.actor_id) inner join film  
on (film_actor.film_id = film.film_id);
```

-- E)

```
select * from customer;  
select city, count(customer.customer_id) as QuantidadeClientes from city inner join address  
on (city.city_id = address.city_id) inner join customer on (address.address_id =  
customer.address_id) where customer.create_date like "2006-02%" group by city.city;
```

-- F)

```
select actor.actor_id, actor.first_name, actor.last_name, count(film.film_id) as  
QuantidadeFilme from actor inner join film_actor on (actor.actor_id = film_actor.actor_id)  
inner join film on (film_actor.film_id = film.film_id)  
where film.rental_duration in (3, 7) and film.length between 60 and 150 and  
film.replacement_cost > 12.00 group by actor.actor_id, actor.first_name, actor.last_name;
```

-- G)

```
select category.name as Categoria, sum(film.replacement_cost) as TotalMulta from category  
inner join film_category on (category.category_id = film_category.category_id) inner join film  
on (film_category.film_id = film.film_id)  
where film.rental_duration in (3, 7) and film.length between 60 and 150 and  
film.replacement_cost > 12.00 group by category.name;
```

-- QUESTÃO 5

-- A)

```
select * from city inner join address on (city.city_id = address.city_id) inner join customer on  
(address.address_id = customer.address_id)  
inner join payment on (customer.customer_id = payment.customer_id);
```

-- B)

```
select * from store inner join staff on (store.store_id = staff.store_id) inner join payment on  
(staff.staff_id = payment.staff_id)  
inner join rental on (payment.rental_id = rental.rental_id) inner join inventory on  
(rental.inventory_id = inventory.inventory_id) inner join film on (inventory.film_id =  
film.film_id);
```

-- C)

```
select * from actor inner join film_actor on (actor.actor_id = film_actor.actor_id) inner join film  
on (film_actor.film_id = film.film_id)  
inner join inventory on (film.film_id = inventory.film_id) inner join rental on  
(inventory.inventory_id = rental.inventory_id)  
inner join payment on (rental.rental_id = payment.rental_id) inner join customer on  
(payment.customer_id = customer.customer_id)  
inner join address on (customer.address_id = address.address_id) inner join city on  
(address.city_id = city.city_id) inner join country on (city.country_id = country.country_id);
```

-- PÁGINA 43

-- QUESTÃO 6

```
/*  
RENTAL, INVENTORY, PAYMENT, FILM e FILM_CATEGORY  
*/
```

-- QUESTÃO 7

```
/*  
CUSTOMER, PAYMENT, ADDRESS, COUNTRY e CITY  
*/
```

-- QUESTÃO 8

-- CÓDIGO Q6

```
select category.name as categoria, sum(payment.amount) as TotalPagamentos from  
payment inner join rental on (payment.rental_id = rental.rental_id)  
inner join inventory on (rental.inventory_id = inventory.inventory_id) inner join film_category  
on (inventory.film_id = film_category.film_id)  
inner join category on (film_category.category_id = category.category_id) group by  
category.name;
```

-- CÓDIGO Q7

```
select country.country as países, count(payment.payment_id) as QuantidadePagamentos  
from payment inner join customer on (payment.customer_id = customer.customer_id)  
inner join address on (customer.address_id = address.address_id) inner join city on  
(address.city_id = city.city_id)
```

```
inner join country on (city.country_id = country.country_id) group by country.country;
```