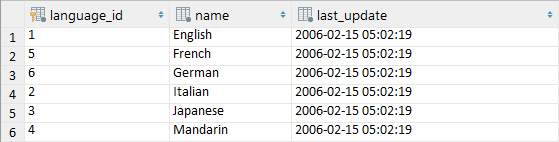
Databanken – SQL – oefeningen

Maak gebruik van de **sakila** database om volgende oefeningen op te lossen.

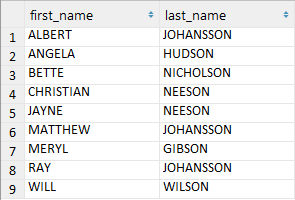
1. Welke talen komen voor in de databank (alfabetisch gesorteerd)? Toon alle velden.



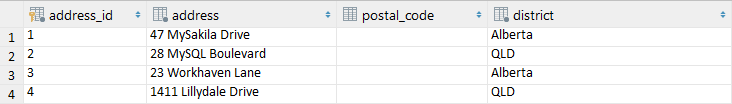
select \* from language

order by name;

1. Geef de volledige naam (voornaam en familienaam) van de acteurs/actrices met “SON” in hun familienaam, gesorteerd op voornaam.

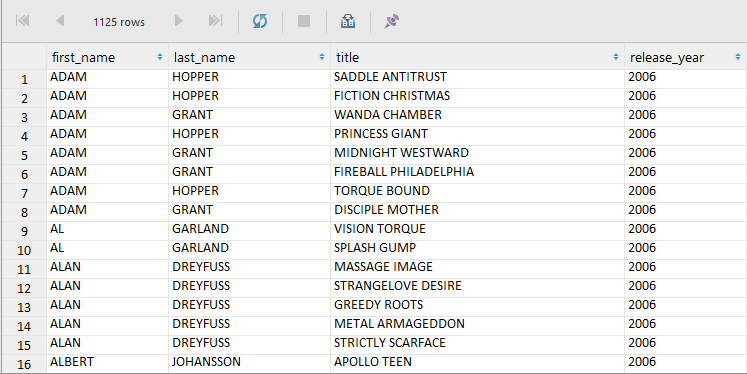


select first\_name, last\_name from actor  
where last\_name like '%SON%'  
order by first\_name;

1. Geef een lijst van alle adressen zonder postcode. Toon de velden zoals hieronder.  
     
   

select address\_id, address, postal\_code, district from address  
where postal\_code in (“”, null);

1. Geef voor- en achternaam van acteurs/actrices die speelden in een film met “Crocodile” of “Shark” in de beschrijving, samen met de titel en het releasejaar van de film, gesorteerd op voornaam.

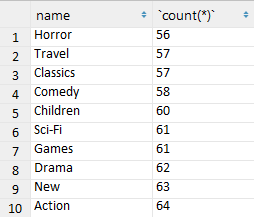


select first\_name, last\_name, title, release\_year  
from actor  
inner join film\_actor on actor.actor\_id = film\_actor.actor\_id  
inner join film on film\_actor.film\_id = film.film\_id  
where film.description like "%Crocodile%"  
or film.description like "%shark%"  
order by first\_name;

1. In hoeveel films komt een krokodil of een haai voor?

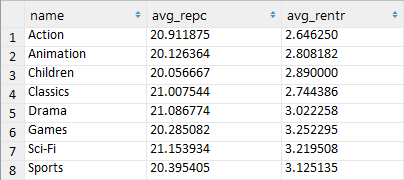
select count(film\_id)  
from film  
where description like '%krokodil%'  
or description like '%haai%'

select count(film\_id)  
from film  
where description like '%crocodile%'  
or description like '%shark%'

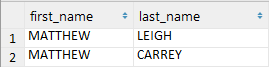
1. Zoek alle filmcategorieën waarin er tussen 55 en 65 films voorkomen. Geef de namen van deze categorieën, en het aantal films daarin, gesorteerd op het aantal films.  
     
   

select name, count(category.category\_id) as amount from film\_category  
inner join category on film\_category.category\_id = category.category\_id  
group by name  
having amount between 55 and 65  
order by amount

1. In hoeveel filmcategorieën is het verschil tussen de gemiddelde vervangkost (replacement cost) en de huurprijs (rental rate) groter dan 17? Geef een lijstje zoals hieronder.

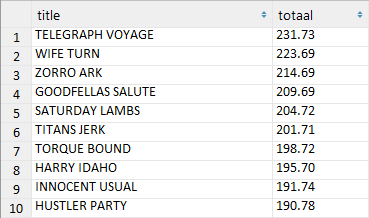


select name, avg(replacement\_cost) as avg\_repc, avg(f.rental\_rate) as avg\_rentr from film\_category fc  
inner join film f on fc.film\_id = f.film\_id  
inner join category c on fc.category\_id = c.category\_id  
group by name  
having (avg\_repc - avg\_rentr) > 17

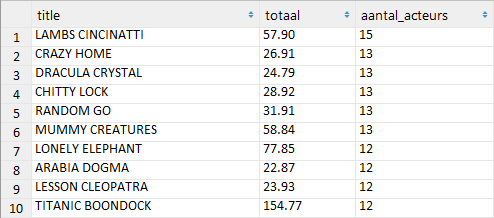
1. Geef de namen (voor- en achternaam) van alle acteurs en klanten wiens voornaam dezelfde is als de voornaam van de acteur met id 8. Neem de acteur met id 8 niet op in het lijstje. Je mag de naam van de acteur met id 8 niet gebruiken in je SQL statement, wel het id (het cijfer 8).  
     
   

select first\_name, last\_name from actor  
where first\_name = (select first\_name from actor where actor\_id = 8)  
and not actor\_id = 8  
union(  
select customer.first\_name, customer.last\_name from customer  
where customer.first\_name = (select first\_name from actor where actor\_id = 8));

1. Geeft de top 10 van de best verhurende films (die het meest opbrachten volgens het totaal van betalingen in de tabel *payments*). Geef titel van de film, en totale opbrengst. Sorteer op totale opbrengst, aflopend.



select f.title, sum(amount) as totaal from payment  
inner join rental r on payment.rental\_id = r.rental\_id  
inner join inventory i on r.inventory\_id = i.inventory\_id  
inner join film f on i.film\_id = f.film\_id  
group by f.title  
order by totaal desc  
limit 10

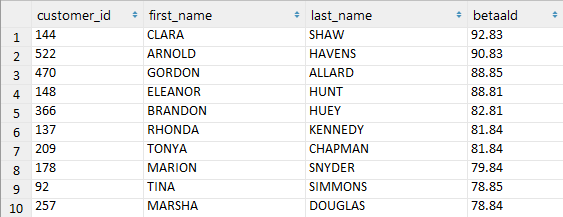
1. Geeft de top 10 van de films met de meeste acteurs. Geef titel van de film, aantal acteurs, en totale opbrengst. Sorteer op het aantal acteurs, aflopend.  
     
   

select title, sum(amount) as totaal, (  
 select count(actor\_id) from film\_actor  
 where film\_actor.film\_id = film.film\_id  
 ) as aantal\_acteurs  
from film  
inner join inventory i on film.film\_id = i.film\_id  
inner join rental r on i.inventory\_id = r.inventory\_id  
inner join payment p on r.rental\_id = p.rental\_id  
group by title  
order by aantal\_acteurs desc

limit 10

1. Geef de top 10 van landen met de meeste klanten.  
     
   

select country, count(customer\_id) as aantal\_klanten from country  
inner join city c on country.country\_id = c.country\_id  
inner join address a on c.city\_id = a.city\_id  
inner join customer c2 on a.address\_id = c2.address\_id  
group by country  
order by aantal\_klanten desc  
limit 10

1. Geef de top 10 klanten volgens hun betalingen in de maand juli 2005. Kolommen zoals hieronder, sorteer op ‘betaald’, aflopend.  
     
   

select c.customer\_id, c.first\_name, c.last\_name, sum(amount) as betaald from payment p  
inner join customer c on p.customer\_id = c.customer\_id  
where date(payment\_date) between '2005-06-01' and '2005-06-31'  
group by c.customer\_id  
order by betaald desc  
limit 10