--challenge 1

SELECT\*,

case when 'Trailers' = any(special\_features) then 1 else 0

end as Trailers,

case when 'Commentaries' = any(special\_features) then 1 else 0

end as Commentaries,

case when 'Deleted Scenes' = any(special\_features) then 1 else 0

end as deleted\_scenes,

case when 'Behind the Scenes' = any(special\_features) then 1 else 0

end as behind\_the\_scenes

FROM FILM;

--challenge 2

select c.name, count(r.rental\_id)

from category c

full join film\_category fc on fc.category\_id = c.category\_id

full join inventory i on i.film\_id = fc.film\_id

full join rental r on r.inventory\_id = i.inventory\_id

group by c.name

order by count(r.rental\_id) desc

--challenge 3: how many rentals have not been paid for?

--Just return the number in one cell in the result set

select count(rental\_id)

from rental where rental\_id not in (select rental\_id from payment)

--challenge 4: which city has brought in the most revenue to the dvd store

--(both stores, assume this is a fun virtual store) thinking about where the customers are from

with city\_sums as (

select ci.city, sum(p.amount) as amount

from payment p

join customer c on p.customer\_id = c.customer\_id

join address a on a.address\_id = c.address\_id

join city ci on ci.city\_id = a.city\_id

group by ci.city

order by sum(p.amount) desc

)

select city, amount from city\_sums

where amount = (select max(amount) from city\_sums)

--challenge 5: Considering every film in the film table, return a result set that gives the number of actors in one column and the frequency of that number of actors in the next column. Order the result by the number of actors.

--the table should be like this hypothetically:

with actor\_count as

(select count(fa.actor\_id) actor\_count, f.film\_id

from film f

full join film\_actor fa on fa.film\_id = f.film\_id

group by f.film\_id

)

select actor\_count, count(actor\_count)

from actor\_count

group by actor\_count

order by actor\_count