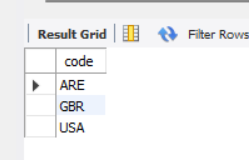
use world;

-- task 1--

select code from country where code in('USA', 'GBR', 'ARE');



Conclusion:-Country names according to their code.

-- task 2--

select avg(sum\_population\_of\_continent) as avg\_of\_sum\_of\_population\_of\_continent

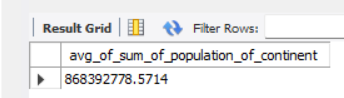
from

(select continent,

sum(population) as sum\_population\_of\_continent

from country

group by continent) as pop;



Conclusion:-The average of the sum of the continental population.

* Task 3--

select name country\_name, Continent, language

from

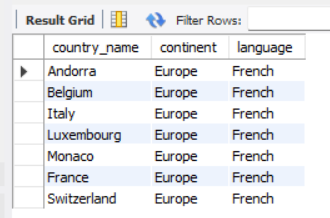
(select name, continent, language

from country

join countrylanguage on country.code = countrylanguage.countrycode) as x

where continent like 'Europe'

and language like 'french';



Conclusion:-The list of countries from the continent of Europe the French is spoken.

-- task 4--

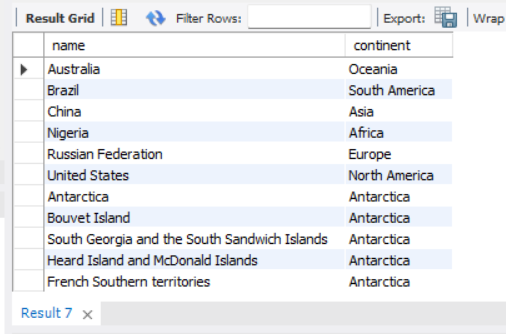
create view view3 as select continent, max(population) as max\_pop from country

group by continent;

select \* from view3;

select name, country.continent from country join view3 on country.continent = view3.continent

and population =view3.max\_pop;



Conclusion:-Country from each continent according to their highest population.

use sakila;

-- task 1--

select movie\_name,

count(counts) no\_of\_copies

from

(select title movie\_name,

a.film\_id counts

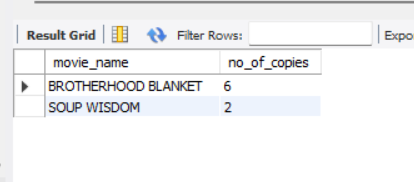
from inventory a

inner join film\_text b on a.film\_id = b.film\_id) exp

where movie\_name like 'BROTHERHOOD BLANKET'

or movie\_name like 'SOUP WISDOM'

group by movie\_name;



Conclusion:-The most popular and trending movies are “BROTHERHOOD BLANKET”, “SOUP WISDOM”.

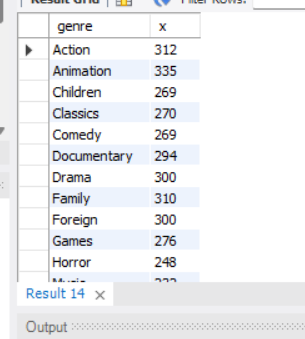
-- task 2--

select name as genre, count(inventory\_id) as x from category

join film\_category using (category\_id)

join inventory using (film\_id)

group by category\_id;



Conclusion:-The total no.of movies in each genre.

-- task 3--

select concat(first\_name,' ',last\_name) full\_name,

c.actor\_id x,

sum(replacement\_cost) replacement\_costs

from film\_actor a

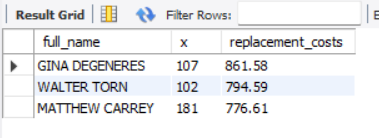
inner join film b on a.film\_id = b.film\_id

inner join actor c on a.actor\_id = c.actor\_id

group by x

order by replacement\_costs desc

limit 3;



Conclusion:-Top 3 actors according to their highest replacement cost.

-- task 4--

select a.name movie\_name,

sum(p.amount) total\_sales

from category a

inner join film\_category b

on a.category\_id=b.category\_id

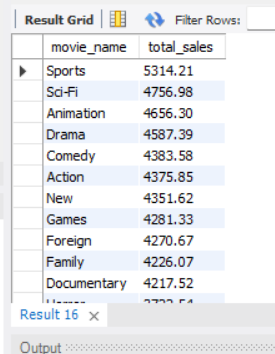
inner join inventory c on b.film\_id=c.film\_id

inner join rental d on c.inventory\_id=d.inventory\_id

inner join payment p on p.rental\_id = d.rental\_id

group by name

order by total\_sales desc;



Conclusion:-Total sales for each movie type.

-- task 5--

select a.name movie\_name,

sum(p.amount) total\_sales

from category a

inner join film\_category b

on a.category\_id=b.category\_id

inner join inventory c on b.film\_id=c.film\_id

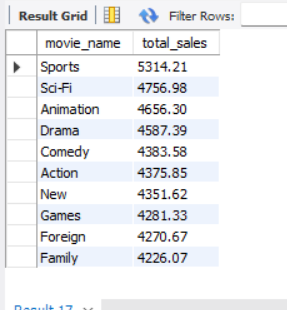
inner join rental d on c.inventory\_id=d.inventory\_id

inner join payment p on p.rental\_id = d.rental\_id

group by name

order by total\_sales desc

limit 10;



Conclusion:-The board wishes to see the list of the top 10 genre.