This lab is to be run on PostgreSQL

Window Functions

Regular Aggregate functions used as Window functions

1. **What is the most recent film (title and year) for every Asian country, ordered by country.**

```
select country_name, title, year_released
from (select c.country_name, m.title, m.year_released,
max(m.year_released) over (partition by c.country_name) as most_recent
    from movies m
        join countries c
          on c.country_code = m.country
      where c.continent = 'ASIA') x
where most_recent = year_released
order by country_name, title
```

2. **Countries in the database with more films than the average, with a common table expression and with a window function**

```
with country_films as
   (select country, count(*) as cnt
    from movies
     group by country)
select c.country_name, x.cnt
from (select country, cnt
      from country_films
      where cnt > (select avg(cnt)
                from country_films)) x
      join countries c
        on c.country_code = x.country
```

```
select c.country_name, cnt from
  ( select country, cnt, avg(cnt) over() as average
   from (select country, count(*) as cnt
      from movies
      group by country) m
```

```
    ) x
    join countries c
    on c.country_code=x.country
    where x.cnt>x.average
```

3. Which percentage of the films of the database does every country represent, ordered by increasing percentage? (use a Window function)

```
select c.country_name, round(100.0 * a.films / sum(a.films) over(), 2) as
pct
from (select country, count(*) as films
    from movies
    group by country) a
    join countries c
      on c.country_code = a.country
 order by pct
```

[Ranking](#)

4. What are the title and year of the ten most recent films from China?

```
select title, year_released
from (select title, year_released,
          rank() over (order by year_released desc) rnk
    from movies
    where country = 'cn') recent
where rnk <= 10
order by rnk
```

5. What is by continent the country with the most movies in the database.

```
select continent, country_name, cnt
from (select continent, country_name, cnt,
          rank() over (partition by continent order by cnt desc) as rnk
    from (select c.continent, c.country_name, count(m.movieid) as cnt
        from countries c
            join movies m
              on m.country = c.country_code
        group by c.continent, c.country_name) x) y
where rnk = 1
```

Can also be done with a common table expression:

```
with continent_count as
    (select c.country_name, c.continent, count(m.movieid) as cnt
     from countries c
         join movies m
           on m.country = c.country_code
     group by c.country_name, c.continent)
 select a.country_name, a.continent, a.cnt
 from continent_count a
     join (select continent, max(cnt) as maxcnt
           from continent_count
           group by continent) b
       on b.continent = a.continent
      and b.maxcnt = a.cnt
```

6. **What are, by country, the top three actors that are found most often in films from China, the United States, France, Italy and India**

```
select y.country_name, p.first_name, p.surname,  y.cnt as appearances
from (select x.country_name,
          x.peopleid,
          x.cnt,
          rank() over (partition by x.country_name
                    order by x.cnt desc) as rnk
    from (select co.country_name, c.peopleid, count(*) as cnt
          from credits c
              join movies m
                on m.movieid = c.movieid
              join countries co
                on co.country_code = m.country
          where c.credited_as = 'A'
            and co.country_name in ('China', 'United States',
                            'France', 'Italy', 'India')
          group by co.country_name, c.peopleid) x) y
    join people p
      on p.peopleid = y.peopleid
where y.rnk <= 3
order by country_name, appearances desc
```

7. **Modify the preceding query to get for the same countries the top three actors that have appeared more than 3 times since 2010 (included)**

```
select y.country_name, p.first_name, p.surname,  y.cnt as appearances
from (select x.country_name,
            x.peopleid,
            x.cnt,
            dense_rank() over (partition by x.country_name
                            order by x.cnt desc) as rnk
      from (select co.country_name, c.peopleid, count(*) as cnt
            from credits c
                join movies m
                  on m.movieid = c.movieid
                join countries co
                  on co.country_code = m.country
            where c.credited_as = 'A'
             and m.year_released >= 2010
             and co.country_name in ('China', 'United States',
                            'France', 'Italy', 'India')
            group by co.country_name, c.peopleid
            having count(*) > 3) x) y
        join people p
          on p.peopleid = y.peopleid
    where y.rnk <= 3
    order by country_name, appearances desc
```

8. **For countries for which we have at least 20 films released in 2010 or later, display for each year the year, the name of the country, the number of films, and the percentage variation since the preceding year (use the lag() function)**

```
select c.country_name,
     year_released,
     year_count,
     round(100.0*(year_count - previous_year_count)
        / case previous_year_count
            when 0 then null
            else previous_year_count
          end, 1) || '%' as variation
from (select country, year_released,
          year_count,
          lag(year_count, 1)
             over (partition by country
                   order by year_released) as previous_year_count,
          sum(year_count)
             over (partition by country) as country_count
    from (select country, year_released,
             count(*) as year_count
       from movies
       where year_released >= 2010
       group by country, year_released) a) b
   join countries c
   on c.country_code = b.country
where country_count >= 20
```