More Joins and Views – LAB 5

1. List all managers with number of managed employees, where the number of managed employees is bigger than 4.

```
SELECT s.FIRST_NAME, s.LAST_NAME, COUNT(1) AS numb_managed
from employee as e
join employee as s on e.MANAGER_ID=s.EMPLOYEE_ID
group by s.EMPLOYEE_ID
having numb_managed > 4;
```

2. List former job titles, start and end dates of the employees sorted by start date

```
SELECT j.JOB_TITLE, e.FIRST_NAME, e.LAST_NAME, jh.START_DATE, jh.END_DATE
FROM job_history jh
JOIN employee e on jh.EMPLOYEE_ID=e.EMPLOYEE_ID
JOIN job j on jh.JOB_ID=j.JOB_ID
ORDER BY jh.START_DATE;
```

3. Count employees by regions where there are employees

```
select r.REGION_NAME, COUNT(1) AS numb_emp
from region r
join country c on c.REGION_ID=r.REGION_ID
join location l on l.COUNTRY_ID=c.COUNTRY_ID
join department d on d.LOCATION_ID=l.LOCATION_ID
join employee e on e.DEPARTMENT_ID=d.DEPARTMENT_ID
GROUP BY r.REGION_ID;
```

4. Create a view jobtitle_salary with two columns:
- salary
- job title

(No personal information should be included; The President's salary should not be listed)

```
create view jobtitle_salary as
select salary, j.JOB_TITLE from employee e
left join job j on e.JOB_ID=j.JOB_ID
where not j.JOB_TITLE = 'President';
```

5. Create a view jobtitle_salary_avg with averaged salaries by Job
- average salary
- job title

(Again, the President's salary should not be listed)

```
create view jobtitle_salary_avg as
select AVG(salary) as salary_avg, j.JOB_TITLE from employee e
left join job j on e.JOB_ID=j.JOB_ID
where not j.JOB_TITLE = 'President'
group by j.JOB_ID;
```

6. Use the view jobtitle_salary to calculate the average salary by job title

```
select avg(salary) as salary_avg, job_title from jobtitle_salary
group by JOB_TITLE;
```

7. Is the result from above equivalent of the one from jobtitle_salary_avg ignoring the order? And why?

The content of the resulting tables may be the same but is not equivalent.
The grouping in the last query is done over Job Title, which is not unique, while the view jobtitle_salary_avg has grouping over job_id which is unique (as primary key). So, it is possible to add a new job_id with already existing job title and then the result will not be the same.

8. Create a view employee_country that contains only the employees that belong to a department and that department has been assigned to a location and country. The view should show two columns:

- full employee's name
- county name where his department is located

```
create view employee_country as
select concat(e.FIRST_NAME,' ',e.LAST_NAME) as name, c.COUNTRY_NAME as
country from employee as e
join department d on e.DEPARTMENT_ID =d.DEPARTMENT_ID
join location l on d.LOCATION_ID =l.LOCATION_ID
join country c on l.COUNTRY_ID =c.COUNTRY_ID;
```