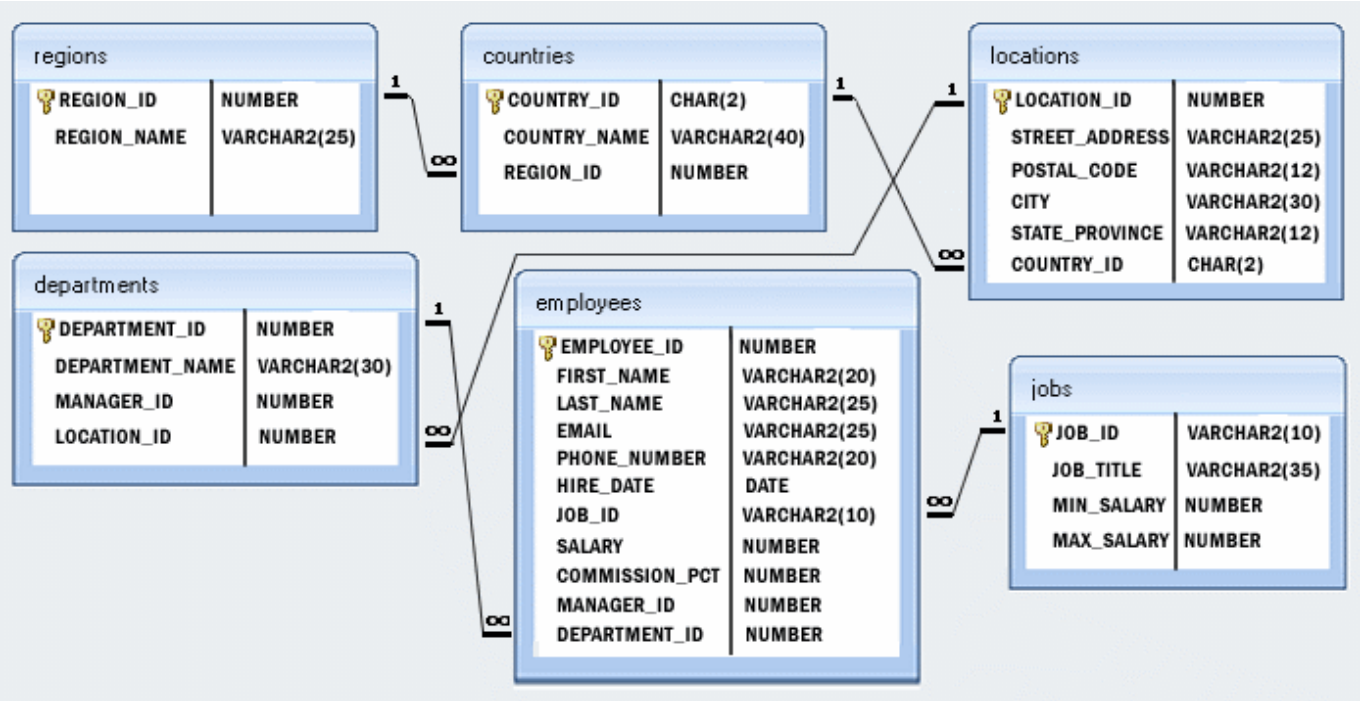
TD - Séance 2

La base de données associée au fichier `Lab12_TD1_SQL_humanresources.sql` contient 7 tables :



Question 1 : Write a query to find the addresses (location_id, street_address, city, state_province, country_name) in the database.

```
SELECT
    location_id,
    street_address,
    city,
    state_province,
    country_name
FROM
    locations
JOIN
    countries ON locations.country_id = countries.country_id;
```

Question 2 : Write a query to find the names (first_name, last name), department ID and depatment name of all the employees. Compare results with INNER JOIN and NATURAL JOIN. Explain Why.

```
SELECT
    first_name,
    last_name,
    departments.department_id,
    department_name
FROM
```

```
    employees
JOIN
    departments ON employees.department_id = departments.department_id;
```

```
SELECT
    first_name,
    last_name,
    departments.department_id,
    department_name
FROM
    employees
NATURAL JOIN
    departments;
```

Question 3 : Display job title, employee name, and the difference between salary of the employee and minimum salary for the job.

```
SELECT
    job_title,
    first_name,
    last_name,
    salary - min_salary AS salary_difference
FROM
    jobs
JOIN
    employees ON jobs.job_id = employees.job_id;
```

Question 4 : Write a query to display the department name, manager name, and city

```
SELECT
    department_name,
    first_name,
    city
FROM
    departments
JOIN
    employees ON departments.manager_id = employees.employee_id
JOIN
    locations ON departments.location_id = locations.location_id;
```

Question 5 : Write a query (using subquery) to find the names (first_name, last_name), the salary of the employees whose salary is greater than the average salary.

```

```

```sql
SELECT
    first_name,
    last_name,
    salary
FROM
    employees
WHERE
    salary > (SELECT avg(salary) FROM employees);
```

Question 6 : Write a query to find the cities without departments within the company.

```sql
SELECT
    city
FROM
    locations
LEFT JOIN
    departments ON locations.location_id = departments.location_id
WHERE
    department_id IS NULL;
```

Question 7 : Same query, but restricted to cities in Japan.

```sql
SELECT
    city
FROM
    locations
LEFT JOIN
    departments ON locations.location_id = departments.location_id
JOIN
    countries ON locations.country_id = countries.country_id
WHERE
    department_id IS NULL
    AND country_name = 'Japan';
```

Question 8 : Find the names (first_name, last_name), job, department number, and department name of the employees who work in London.

```sql
SELECT
    e.first_name,
    e.last_name,
    e.job_id,
    d.department_id,
    d.department_name
FROM
    employees e
```

```
JOIN
    departments d ON e.department_id = d.department_id
JOIN
    locations l ON d.location_id = l.location_id
WHERE
    l.city = 'London';
```

Question 9 : Write a query to find all the employees hired before all the employees of department « SALES ».

```
SELECT
    first_name,
    last_name,
    hire_date
FROM
    employees
JOIN
    departments ON employees.department_id = departments.department_id
WHERE
    hire_date < (SELECT MIN(hire_date) FROM employees
JOIN
    departments ON employees.department_id = departments.department_id
WHERE
    department_name = 'SALES')
ORDER BY
    hire_date
ASC;
```

Question 10 : Write a query to find the employee id, name (last_name) along with their manager_id, manager name (last_name).

```
SELECT
    e.employee_id,
    e.last_name,
    e.manager_id,
    m.last_name AS manager_last_name
FROM
    employees e
LEFT JOIN
    employees m ON e.manager_id = m.employee_id;
```

La requête pour la question 10 utilise un LEFT JOIN pour s'assurer que tous les employés sont inclus, même ceux qui n'ont pas de manager (comme King).

Question 11 : In the previous query, King(ID 100) does not appear in the employee list, how to integrate him ?

```sql
SELECT
    e.employee_id,
    e.last_name,
    e.manager_id,
    COALESCE(m.last_name, 'No Manager') AS manager_name
FROM
    employees e
LEFT JOIN
    employees m ON e.manager_id = m.employee_id;
```

Question 12 : Write a query to find the employees whose manager is not managing a department.

```sql
SELECT
    e.first_name,
    e.last_name
FROM
    employees e
LEFT JOIN
    departments d ON e.manager_id = d.manager_id
WHERE
    d.department_id IS NULL;
```

Question 13 : Write a query to display the name (first_name, last_name), hire date, salary of the manager for all managers whose experience is more than 15 years.

```sql
SELECT
    first_name,
    last_name,
    hire_date,
    salary
FROM
    employees
WHERE
    hire_date < CURRENT_DATE - INTERVAL '15 years';
```

Question 14 : Write a query to display all people hired before « Ernst » using both join and subqueries.

```sql
SELECT
    first_name,
    last_name
FROM
    employees
```

```
WHERE
    hire_date < (SELECT hire_date FROM employees WHERE last_name = 'Ernst');
```

Question 15 : Write a query (using join) to find all the employees hired before « Greenberg » and who have a lower salary than him.

```
SELECT
    e.first_name,
    e.last_name
FROM
    employees e
JOIN
    employees g ON e.hire_date < g.hire_date AND e.salary < g.salary
WHERE
    g.last_name = 'Greenberg';
```

Question 16 : Using a subquery, write a query to find the names of employees who are not managers. Hint: use the keyword EXISTS.

```
SELECT
    e.first_name,
    e.last_name
FROM
    employees e
WHERE
    NOT EXISTS (SELECT 1 FROM employees WHERE manager_id = e.employee_id);
```

Question 17 : Write a query to display the department ID, department name and manager first name.

```
SELECT
    department_id,
    department_name,
    (SELECT first_name FROM employees WHERE employee_id = manager_id) AS
manager_first_name
FROM
    departments;
```

Question 18 : Write a query to find the employees who have the same job as « Ernst » but a different manager.

```
SELECT
    e.first_name,
```

```
    e.last_name
FROM
    employees e
JOIN
    employees e2 ON e.job_id = e2.job_id
WHERE
    e2.last_name = 'Ernst' AND e.manager_id <> e2.manager_id;
```

Question 19 : Write a query to display the department name, manager name, and city.

```
SELECT
    department_name,
    (SELECT first_name FROM employees WHERE employee_id = departments.manager_id)
AS manager_name,
    city
FROM
    departments
JOIN
    locations ON departments.location_id = locations.location_id;
```

Question 20 : Write a query to find the firstname and lastname of people who have no job history.

```
SELECT
    first_name,
    last_name
FROM
    employees
WHERE
    employee_id NOT IN (SELECT employee_id FROM job_history);
```

Question 21 : Write a query to get department name, manager name, and salary for all managers that are employed for more than 5 years.

```
SELECT
    department_name,
    (SELECT first_name FROM employees WHERE employee_id = departments.manager_id)
AS manager_name,
    salary
FROM
    employees
JOIN
    departments ON employees.employee_id = departments.manager_id
WHERE
    hire_date < CURRENT_DATE - INTERVAL '5 years';
```

Question 22 : Write a query to return the cities, their country and region names where some employees earn more than 3000.

```
SELECT
    city,
    country_name,
    region_name
FROM
    locations
JOIN
    countries ON locations.country_id = countries.country_id
JOIN
    employees ON locations.location_id = departments.location_id
WHERE
    salary > 3000;
```