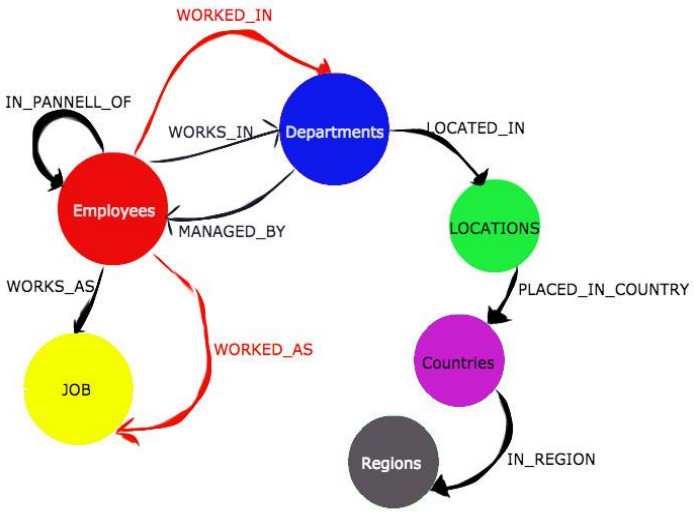
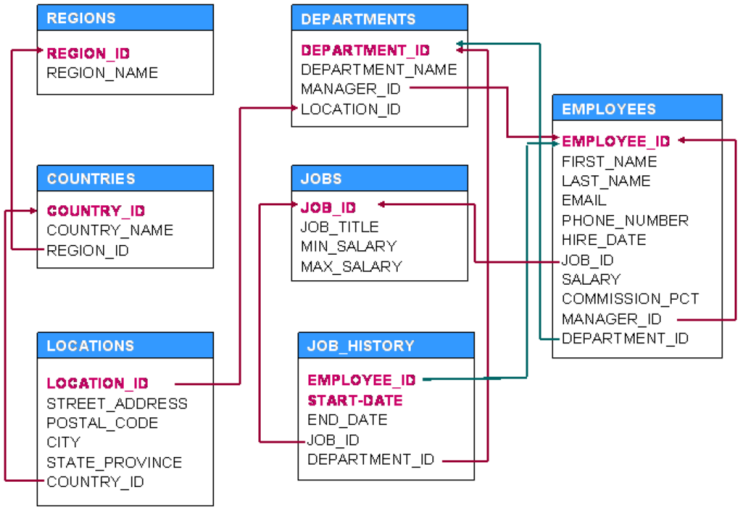
Practical Examination – Questions + Answers – Group 1





Note: **(Only 10)** of these questions below must be solved using SQL queries based on the standard Oracle HR schema tables. You are expected to use correct SQL syntax, apply joins, subqueries, group functions, and filtering conditions where needed.
Answers must directly reference the appropriate tables and fields from the HR schema.

Q1: Display the total salary paid to employees in each job title.

Q2: Find the number of employees who joined the company after the year 2010.

Q3: Find the average commission percentage (excluding NULLs).

Q4: Show the minimum salary and maximum salary for each department.

Q5: Find the employees who earn more than the min salary of any department except department 80.

Q6: List employees whose salary is higher than all salaries of employees who have the job 'IT_PROG'.

Q7: Find all departments that do not have any employees earning more than $10,000.

Q8: Find employees who joined earlier than their manager.

Q9: Display the department name and the highest salary in each department, but only for departments where the highest salary is above 10,000.

Q10: Display the list of all departments, including departments with no employees, along with the number of employees in each department.

Q11: List each employee and their manager's name; if the employee has no manager, display 'No Manager'.

Q12: Create a report showing employees' first name, job title, and their manager's job title.

Answer Sheet:

Q1: Display the total salary paid to employees in each job title.

```
SELECT job_id, SUM(salary) AS total_salary
FROM employees
GROUP BY job_id;
```

Q2: Find the number of employees who joined the company after the year 2010

```
SELECT COUNT(*) AS employees_after_2010
FROM employees
WHERE hire_date > TO_DATE('2010-12-31', 'YYYY-MM-DD');
```

Q3: Find the average commission percentage (excluding NULLs).

```
SELECT AVG(commission_pct) AS avg_commission
FROM employees
WHERE commission_pct IS NOT NULL;
```

Q4: Show the minimum salary and maximum salary for each department.

```
SELECT department_id, MIN(salary) AS min_salary, MAX(salary) AS max_salary
FROM employees
GROUP BY department_id;
```

Q5: Find the employees who earn more than the min salary of any department except department 80.

```
SELECT first_name, last_name, salary
FROM employees
WHERE salary > (
                SELECT MIN(salary)
                FROM employees
                WHERE department_id != 80);
```

Q6: List employees whose salary is higher than all salaries of employees who have the job 'IT_PROG'.

```
SELECT first_name, last_name, salary
FROM employees
WHERE salary > ALL (
  SELECT salary
  FROM employees
  WHERE job_id = 'IT_PROG');
```

Q7: Find all departments that do not have any employees earning more than $10,000.

```
SELECT d.department_id, d.department_name
FROM departments d
WHERE NOT EXISTS (
  SELECT 1
  FROM employees e
  WHERE e.department_id = d.department_id
  AND e.salary > 10000);
```

Q8: Find employees who joined earlier than their manager.

```
SELECT e.employee_id, e.first_name, e.hire_date, m.first_name AS manager_name, m.hire_date AS
manager_hire_date
FROM employees e
JOIN employees m ON e.manager_id = m.employee_id
WHERE e.hire_date < m.hire_date;
```

Q9: Display the department name and the highest salary in each department, but only for departments where the highest salary is above 10,000.

```
SELECT d.department_name, MAX(e.salary) AS highest_salary
FROM departments d
JOIN employees e ON d.department_id = e.department_id
GROUP BY d.department_name
HAVING MAX(e.salary) > 10000;
```

Q10: Display the list of all departments, including departments with no employees, along with the number of employees in each department.

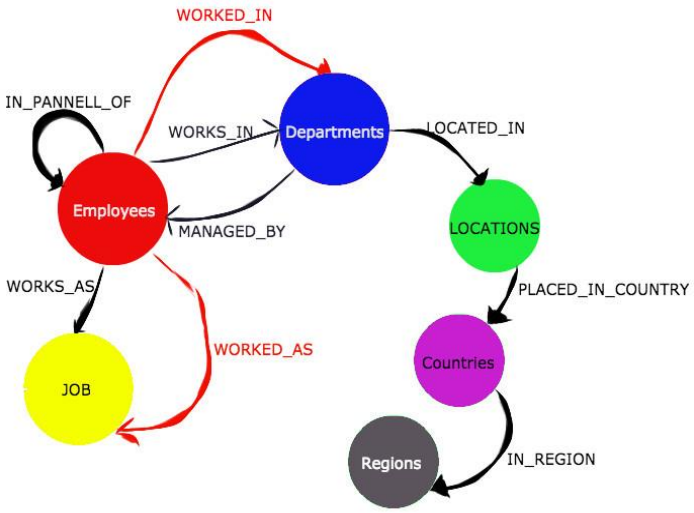
```
SELECT d.department_id, d.department_name, COUNT(e.employee_id) AS employee_count
FROM departments d
LEFT JOIN employees e ON d.department_id = e.department_id
GROUP BY d.department_id, d.department_name;
```

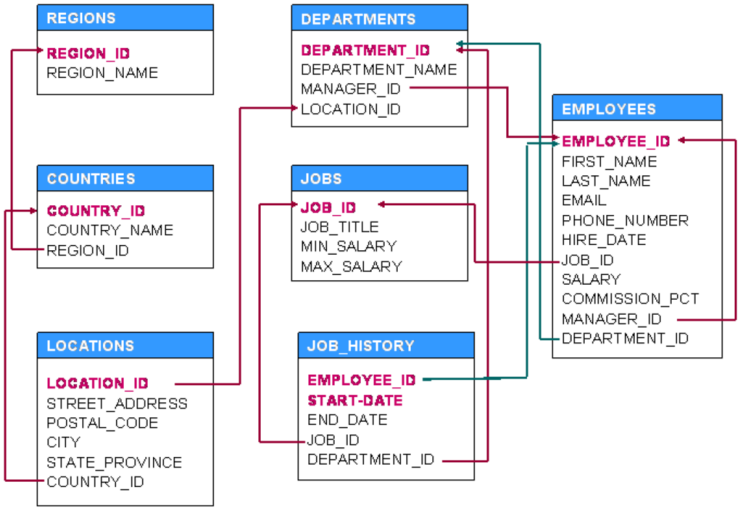
Q11: List each employee and their manager's name; if the employee has no manager, display 'No Manager'.

```
SELECT e.first_name || ' ' || e.last_name AS employee_name, NVL(m.first_name || ' ' || m.last_name, 'No
Manager') AS manager_name
FROM employees e
LEFT JOIN employees m ON e.manager_id = m.employee_id;
```

Q12: Create a report showing employees' first name, job title, and their manager's job title.

```
SELECT e.first_name AS employee_name,
    e.job_id AS employee_job,
    m.job_id AS manager_job
FROM employees e
LEFT JOIN employees m ON e.manager_id = m.employee_id;
```





Note: **(Only 10)** of these questions below must be solved using SQL queries based on the standard Oracle HR schema tables. You are expected to use correct SQL syntax, apply joins, subqueries, group functions, and filtering conditions where needed.
Answers must directly reference the appropriate tables and fields from the HR schema.

Q1: List the departments where the average salary is more than 7000.

Q2: Find the total number of employees in the HR department.

Q3: For each job title, show the total salary paid, but exclude jobs with 'REP' in the job ID.

Q4: Display the average salary across the whole company, but treat NULL commission_pct as 0 while calculating average salary.

Q5: Display the last name, department name, and city for employees who work in departments located in Toronto.

Q6: List employees who have the same salary as 'King'.

Q7: Find employees who earn more than any employee in department 50.

Q8: Display employees who do not have a manager.

Q9: List the employees and their department name and location city who are working in departments located outside the USA.

Q10: Display a list of all employees who do NOT have any subordinates (they are not managers).

Q11: Write a query to create a report showing each department name and the total salary paid to employees in that department.

Q12: Find the employee who has the third highest salary.

Answer Sheet:

Q1: List the departments where the average salary is more than 7000.

```
SELECT department_id, AVG(salary) AS avg_salary
FROM employees
GROUP BY department_id
HAVING AVG(salary) > 7000;
```

Q2: Find the total number of employees in the HR department.

```
SELECT COUNT(*) AS total_hr_employees
FROM employees e
JOIN departments d ON e.department_id = d.department_id
WHERE d.department_name = 'Human Resources';
```

Q3: For each job title, show the total salary paid, but exclude jobs with 'REP' in the job ID.

```
SELECT job_id, SUM(salary) AS total_salary
FROM employees
WHERE job_id NOT LIKE '%REP%'
GROUP BY job_id;
```

Q4: Display the average salary across the whole company, but treat NULL commission_pct as 0 while calculating average salary.

```
SELECT AVG(salary + NVL(commission_pct, 0)) AS avg_total_compensation
FROM employees;
```

Q5: Display the last name, department name, and city for employees who work in departments located in Toronto.

```
SELECT e.last_name, d.department_name, l.city
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 = 'Toronto';
```

Q6: List employees who have the same salary as 'King'.

```
SELECT first_name, last_name, salary
FROM employees
WHERE salary = (
  SELECT salary
  FROM employees
  WHERE last_name = 'King');
```

Q7: Find employees who earn more than any employee in department 50.

```
SELECT first_name, last_name, salary
FROM employees
WHERE salary > ANY (
  SELECT salary
  FROM employees
  WHERE department_id = 50);
```

Q8: Display employees who do not have a manager.

```
SELECT employee_id, first_name, last_name
FROM employees
WHERE manager_id IS NULL;
```

Q9: List the employees and their department name and location city who are working in departments located outside the USA.

```
SELECT e.first_name, e.last_name, d.department_name, l.city
FROM employees e
JOIN departments d ON e.department_id = d.department_id
JOIN locations l ON d.location_id = l.location_id
JOIN countries c ON l.country_id = c.country_id
WHERE c.country_name != 'United States of America';
```

Q10: Display a list of all employees who do NOT have any subordinates (they are not managers).

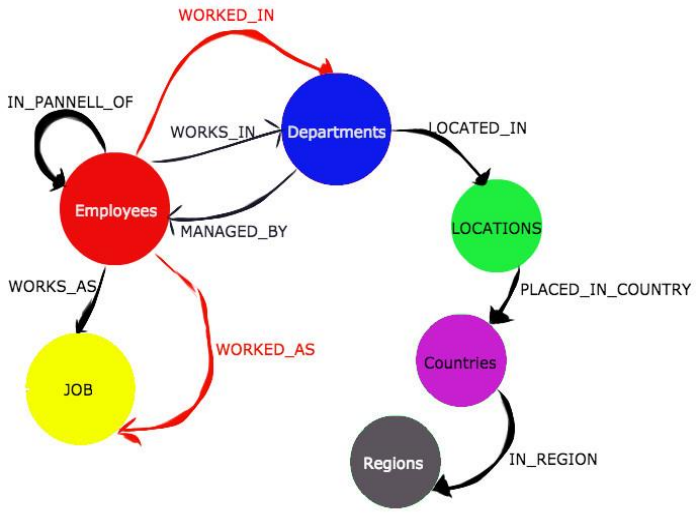
```
SELECT employee_id, first_name, last_name
FROM employees
WHERE employee_id NOT IN (
   SELECT DISTINCT manager_id
   FROM employees
   WHERE manager_id IS NOT NULL);
```

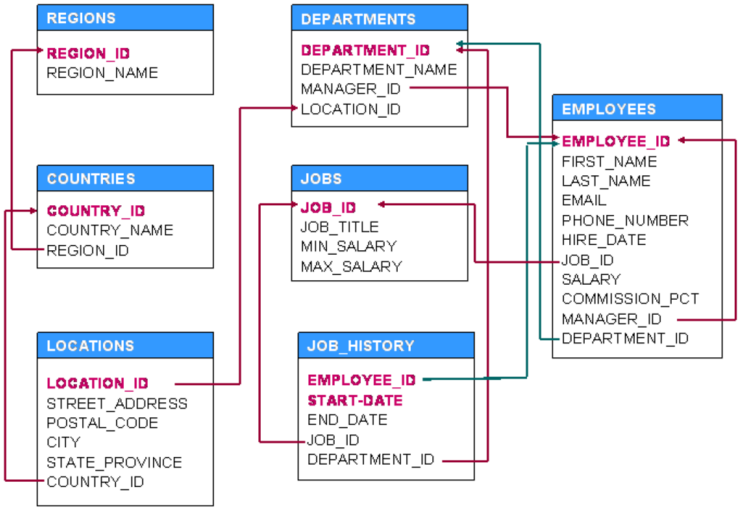
Q11: Create a report showing each department name and the total salary paid to employees in that department.

```
SELECT d.department_name, SUM(e.salary) AS total_salary
FROM departments d
LEFT JOIN employees e ON d.department_id = e.department_id
GROUP BY d.department_name;
```

Q12: Find the employee who has the third highest salary.

```
SELECT *
FROM (
   SELECT employee_id, first_name, last_name, salary,
        DENSE_RANK() OVER (ORDER BY salary DESC) AS rank
   FROM employees
)
WHERE rank = 3;
```





Note: **(Only 10)** of these questions below must be solved using SQL queries based on the standard Oracle HR schema tables. You are expected to use correct SQL syntax, apply joins, subqueries, group functions, and filtering conditions where needed.
Answers must directly reference the appropriate tables and fields from the HR schema.

Q1: Find the maximum average salary across all departments.

Q2: Find the highest and lowest hire date in the employees table.

Q3: Find the number of distinct jobs in the employees table.

Q4: Display the department IDs and the sum of salaries for each, but only for departments where the total salary is more than $40,000.

Q5: Find employees whose salary is less than all employees in 'IT' department.

Q6: Find the department(s) where no employee earns a salary greater than $8000.

Q7: Find employees who work in the same location as 'Sales' department.

Q8: Find the departments where the highest salary is less than 9000.

Q9: Create a query that shows, for each department, the average salary, maximum salary, and number of employees, and only show those departments where the number of employees is more than 5.

Q10: Find the employee who has the third highest salary.

Q11: Display the department name and the highest salary in each department, but only for departments where the highest salary is above 10,000.

Q12: Display the list of all departments, including departments with no employees, along with the number of employees in each department.

Answer Sheet:

Q1: Find the maximum average salary across all departments.

```
SELECT MAX(avg_salary) AS max_avg_salary
FROM (
   SELECT AVG(salary) AS avg_salary
   FROM employees
   GROUP BY department_id);
```

Q2: Find the highest and lowest hire date in the employees table.

```
SELECT MAX(hire_date) AS latest_hire_date,
     MIN(hire_date) AS earliest_hire_date
FROM employees;
```

Q3: Find the number of distinct jobs in the employees table.

```
SELECT COUNT(DISTINCT job_id) AS distinct_jobs
FROM employees;
```

Q4: Display the department IDs and the sum of salaries for each, but only for departments where the total salary is more than $40,000.

```
SELECT department_id, SUM(salary) AS total_salary
FROM employees
GROUP BY department_id
HAVING SUM(salary) > 40000;
```

Q5: Find employees whose salary is less than all employees in 'IT' department.

```
SELECT first_name, last_name, salary
FROM employees
WHERE salary < ALL (
   SELECT salary
   FROM employees
   WHERE department_id = (
      SELECT department_id
      FROM departments
      WHERE department_name = 'IT' ));
```

Q6: Find the department(s) where no employee earns a salary greater than $8000.

```
SELECT department_id
FROM employees
GROUP BY department_id
HAVING MAX(salary) <= 8000;
```

Q7: Find employees who work in the same location as 'Sales' department.

```
SELECT e.first_name, e.last_name, d.department_name
FROM employees e
JOIN departments d ON e.department_id = d.department_id
WHERE d.location_id = (
   SELECT location_id
   FROM departments
   WHERE department_name = 'Sales');
```

Q8: Find the departments where the highest salary is less than 9000.

```
SELECT department_id
FROM employees
GROUP BY department_id
HAVING MAX(salary) < 9000;
```

Q9: Show, for each department, the average salary, maximum salary, and number of employees, only if the department has more than 5 employees.

```
SELECT department_id,
    AVG(salary) AS avg_salary,
    MAX(salary) AS max_salary,
    COUNT(*) AS employee_count
FROM employees
GROUP BY department_id
HAVING COUNT(*) > 5;
```

Q10: Find the employee who has the third highest salary.

```
SELECT *
FROM (
  SELECT employee_id, first_name, last_name, salary,
      DENSE_RANK() OVER (ORDER BY salary DESC) AS rank
  FROM employees
)
WHERE rank = 3;
```

Q11: Display the department name and the highest salary in each department, but only for departments where the highest salary is above 10,000.

```
SELECT d.department_name, MAX(e.salary) AS highest_salary
FROM departments d
JOIN employees e ON d.department_id = e.department_id
GROUP BY d.department_name
HAVING MAX(e.salary) > 10000;
```

Q12: Display the list of all departments, including departments with no employees, along with the number of employees in each department.

```
SELECT d.department_id, d.department_name, COUNT(e.employee_id) AS employee_count
FROM departments d
LEFT JOIN employees e ON d.department_id = e.department_id
GROUP BY d.department_id, d.department_name;
```