1. **List the employee names, job titles, department names, and city of all employees.**

   SQL

```
SELECT
  e.first_name || ' ' || e.last_name AS "Employee Name",
  j.job_title AS "Job Title",
  d.department_name AS "Department Name",
  l.city AS "City"
FROM
  employees e
JOIN
  jobs j ON e.job_id = j.job_id
JOIN
  departments d ON e.department_id = d.department_id
JOIN
  locations l ON d.location_id = l.location_id;
```

---

2. **Display employee_id, name, salary, department id, and average salary of their department.**

   SQL

```
SELECT
  e.employee_id,
  e.first_name || ' ' || e.last_name AS "Employee Name",
  e.salary,
  e.department_id,
  (SELECT AVG(salary) FROM employees WHERE department_id = e.department_id) AS "Department Avg Salary"
FROM
  employees e;
```

---

3. **Display the name of employees who have the same department as 'Steven King', but exclude Steven King himself.**

   SQL

```
SELECT
  first_name || ' ' || last_name AS "Employee Name"
FROM
  employees
WHERE
  department_id = (
    SELECT department_id
    FROM employees
    WHERE first_name = 'Steven' AND last_name = 'King'
  )
  AND NOT (first_name = 'Steven' AND last_name = 'King');
```

---

4. **Display employee name, salary, and commission percentage of employees working in the 'Sales' department.**

SQL

```sql
SELECT
  e.first_name || ' ' || e.last_name AS "Employee Name",
  e.salary,
  e.commission_pct
FROM
  employees e
JOIN
  departments d ON e.department_id = d.department_id
WHERE
  d.department_name = 'Sales';
```

---

5. **Find the department that has the maximum number of employees, using the employees, departments, and locations tables.**

SQL

```sql
SELECT
  d.department_name
FROM
  employees e
JOIN
  departments d ON e.department_id = d.department_id
GROUP BY
  d.department_name
ORDER BY
  COUNT(e.employee_id) DESC
FETCH FIRST 1 ROW ONLY;
-- Note: 'FETCH FIRST 1 ROW ONLY' is standard SQL; 'ROWNUM = 1' is Oracle-specific.
```

---

6. **Find the manager name and the number of employees working under each manager.**

SQL

```sql
SELECT
  m.first_name || ' ' || m.last_name AS "Manager Name",
  COUNT(e.employee_id) AS "Number of Employees"
FROM
  employees e
JOIN
  employees m ON e.manager_id = m.employee_id
GROUP BY
  m.first_name, m.last_name
ORDER BY
  "Number of Employees" DESC;
```

---

7. **Find the employees who were hired before their department's manager.**

SQL

```sql
SELECT
```

```
    e.first_name || ' ' || e.last_name AS "Employee Name",
    e.hire_date AS "Employee Hire Date",
    m.first_name || ' ' || m.last_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;
```

---

8. **Display employee name, hire date, and manager name.**

SQL

```
SELECT
    e.first_name || ' ' || e.last_name AS "Employee Name",
    e.hire_date,
    m.first_name || ' ' || m.last_name AS "Manager Name"
FROM
    employees e
LEFT JOIN
    employees m ON e.manager_id = m.employee_id;
```

---

9. **Display job id for each job where the average salary exceeds $10,000.**

SQL

```
SELECT
    job_id
FROM
    employees
GROUP BY
    job_id
HAVING
    AVG(salary) > 10000;
```

---

10. **Find the departments that do not have any employees assigned.**

SQL

```
SELECT
    d.department_id,
    d.department_name
FROM
    departments d
LEFT JOIN
    employees e ON d.department_id = e.department_id
WHERE
    e.employee_id IS NULL;
```