SQL Internship – Task 9 Solution
Writing Subqueries (Nested Queries)

This document contains the solution for Task 9 of the SQL Developer Internship. It demonstrates different types of subqueries including simple, correlated, and subqueries in SELECT, FROM, and WHERE clauses.

SQL Script:

```
-- 1. Find employees earning more than average salary
SELECT *
FROM employees
WHERE salary > (SELECT AVG(salary) FROM employees);

-- 2. Subquery in SELECT clause
SELECT emp_name,
       (SELECT AVG(salary) FROM employees) AS avg_salary
FROM employees;

-- 3. Subquery in FROM clause
SELECT dept_avg.department, dept_avg.avg_salary
FROM (
    SELECT department, AVG(salary) AS avg_salary
    FROM employees
    GROUP BY department
) AS dept_avg;

-- 4. Correlated Subquery
SELECT e1.emp_name, e1.salary
FROM employees e1
WHERE e1.salary > (
    SELECT AVG(e2.salary)
    FROM employees e2
    WHERE e2.department = e1.department
);

-- 5. Subquery vs JOIN Example
SELECT e.emp_name, e.salary
FROM employees e
JOIN (
    SELECT department, AVG(salary) AS avg_salary
    FROM employees
    GROUP BY department
) d
ON e.department = d.department
WHERE e.salary > d.avg_salary;
```

Interview Questions: 1. What is a subquery? A query written inside another query. 2. Difference between correlated and non-correlated subqueries? Correlated subqueries depend on outer query values, non-correlated do not. 3. Can subqueries return multiple rows? Yes, using IN, ANY, ALL operators. 4. Subquery vs JOIN – which is better? JOIN is usually faster, but subqueries are useful for complex logic. 5. Where are subqueries commonly used? In filtering, calculations, and nested data analysis queries.