Database Programming

Single-Row Subqueries

ORACLE ACADEMY

Objectives

This lesson covers the following objectives:

- Construct and execute a single-row subquery in the WHERE clause or HAVING clause
- Construct and execute a SELECT statement using more than one subquery
- Construct and execute a SELECT statement using a group function in the subquery

Purpose

As you have probably realized, subqueries are a lot like Internet search engines. They are great at locating the information needed to accomplish another task.

In this lesson, you will learn how to create even more complicated tasks for subqueries to do for you. Keep in mind that subqueries save time in that you can accomplish two tasks in one statement.

Facts About Single-row Subqueries

They:

- Return only one row
- Use single-row comparison operators (=, >,>=, <, <=, <>)

Always:

- Enclose the subquery in parentheses.
- Place the subquery on the right hand side of the comparison condition.

Additional Subquery Facts

- The outer and inner queries can get data from different tables.
- Only one ORDER BY clause can be used for a SELECT statement, and if specified, it must be the last clause in the main SELECT statement.
- The only limit on the number of subqueries is the buffer size that the query uses.

Subqueries from Different Tables

The outer and inner queries can get data from different tables.

Who works in the Marketing department?

DEPARTMENT_ID	DEPARTMENT_NAME
10	Administration
20	Marketing
50	Shipping
…	…

```
SELECT last_name, job_id,
department_id
FROM employees
WHERE department_id =
    (SELECT department_id
     FROM departments
     WHERE department_name =
‘Marketing’)
ORDER BY job_id;
```

LAST_NAME	JOB_ID	DEPARTMENT_ID
Hartstein	MK_MAN	20
Fay	MK_REP	20

Subqueries from Different Tables (cont.)

More than one subquery can return information to the outer query.

```
SELECT last_name, job_id,
salary, department_id
FROM  employees
WHERE job_id =
          (SELECT job_id
        FROM   employees
        WHERE  employee_id = 141)
AND department_id =
            (SELECT department_id
      FROM   departments
      WHERE  location_id =1500);
```

LAST_NAME	JOB_ID	DEPARTMENT_ID
Lorentz	IT_PROG	60
Mourgos	ST_MAN	50
Rajs	ST_CLERK	50
Davies	ST_CLERK	50
Matos	ST_CLERK	50

DEPARTMENT_ID	LOCATION_ID
10	1700
20	1800
50	1500
60	1400

LAST_NAME	JOB_ID	SALARY	DEPARTMENT_ID
Rajs	ST_CLERK	3500	50
Davies	ST_CLERK	3100	50
Matos	ST_CLERK	2600	50

Group Functions in Subqueries

Group functions can be used in subqueries. A group function without a GROUP BY clause in the subquery returns a single row.

This query answers the question, “Which Global Fast Foods staff earn less than the maximum salary?”

```
SELECT last_name,
first_name, salary
FROM f_staffs
WHERE salary <
(SELECT MAX(salary)
      FROM f_staffs);
```

F_STAFFS

MAX (SALARY)
60

F_STAFFS

LAST_NAME	FIRST_NAME	SALARY
Doe	Sue	6.75
Miller	Bob	10

Subqueries in the HAVING Clause

Subqueries can also be placed in the HAVING clause.

Remember that the HAVING clause is similar to the WHERE clause, except that the HAVING clause is used to restrict groups and always includes a group function such as MIN, MAX, or AVG.

Because the HAVING clause always includes a group function, the subquery will nearly always include a group function as well.

Subquery Example

Which departments have a lowest salary that is greater than the lowest salary in department 50? In this example, the subquery selects and returns the lowest salary in department 50.

```
SELECT department_id,
MIN(salary)
FROM employees
GROUP BY department_id
HAVING MIN(salary) >
     (SELECT MIN(salary)
     FROM employees
     WHERE department_id = 50);
```

DEPARTMENT_ID	MIN(SALARY)
10	4400
20	6000
60	4200
80	8600
90	17000
110	8300

EMPLOYEES

MIN(SALARY)
2500

Subquery Example (cont.)

The outer query uses this value to select the department ID and lowest salaries of all the departments whose lowest salary is greater than that number.

The HAVING clause eliminated those departments whose MIN salary was less than department 50’s MIN salary.

```
SELECT department_id,
MIN(salary)
FROM employees
GROUP BY department_id
HAVING MIN(salary) >
     (SELECT MIN(salary)
     FROM employees
     WHERE department_id = 50);
```

DEPARTMENT_ID	MIN(SALARY)
10	4400
20	6000
60	4200
80	8600
90	17000
110	8300

EMPLOYEES

MIN(SALARY)
2500

Summary

In this lesson, you should have learned how to:

- Construct and execute a single-row subquery in the WHERE clause or HAVING clause
- Construct and execute a SELECT statement using more than one subquery
- Construct and execute a SELECT statement using a group function in the subquery