

Using Subqueries to Solve Queries

Objectives

After completing this lesson, you should be able to do the following:

- Define subqueries
- Describe the types of problems that subqueries can solve
- Identify the types of subqueries
- Write single-row, multiple-row, multiple-column subqueries

Lesson Agenda

- Subquery: Types, syntax, and guidelines
- Single-row subqueries:
 - Group functions in a subquery
 - HAVING clause with subqueries
- Multiple-row subqueries
 - Using ALL or ANY operator
- Multiple-column subqueries
- Null values in a subquery

Using a Subquery to Solve a Problem



Subquery Syntax

- The subquery (inner query) executes before the main query (outer query).
- The result of the subquery is used by the main query.

```
SELECT select_list  
FROM table  
WHERE expr operator (SELECT select_list  
FROM table);
```

Main Query

Subquery

Using a Subquery



Main Query:

Determine the names of all employees who were hired after Davies?



Sub Query:

When was Davies hired?

```
SELECT last_name, hire_date
FROM   employees
WHERE  hire_date > (SELECT hire_date
                    FROM   employees
                    WHERE  last_name = 'Davies');
```


Rules and Guidelines for Using Subqueries

- Enclose subqueries in parentheses.
- Place subqueries on the right side of the comparison condition for readability.
(However, the subquery can appear on either side of the comparison operator.)
- Use single-row operators with single-row subqueries and multiple-row operators with multiple-row subqueries.



Subquery Syntax

- The subquery (inner query) executes before the main query (outer query).
- The result of the subquery is used by the main query.

```
SELECT select_list  
FROM table  
WHERE expr operator (SELECT select_list  
FROM table);
```

Main Query

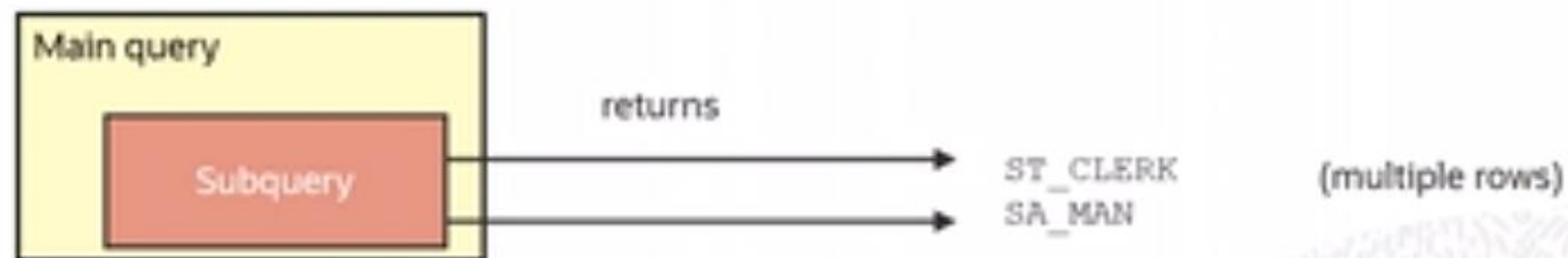
Subquery

Types of Subqueries

- Single-row subquery



- Multiple-row subquery





Single-Row Subqueries

- Return only one row
- Use single-row comparison operators

Operator	Meaning
=	Equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>	Not equal to

Executing Single-Row Subqueries

```
SELECT last_name, job_id, salary
FROM employees
WHERE job_id =  SA_REP
AND salary >  8600
      (SELECT salary
       FROM employees
       WHERE last_name = 'Taylor');
      (SELECT job_id
       FROM employees
       WHERE last_name = 'Taylor');
```



	LAST_NAME	JOB_ID	SALARY
1	Abel	SA_REP	11000



#	last_name	job_id	salary
1	Abel	SA_REP	11000.00

Using Group Functions in a Subquery

```
SELECT last_name, job_id, salary
FROM employees
WHERE salary = (SELECT MIN(salary)
                FROM employees);
```



	LAST_NAME	JOB_ID	SALARY
1	Vargas	ST_CLERK	2500



#	last_name	job_id	salary
1	Vargas	ST_CLERK	2500.00

HAVING Clause with Subqueries

The database server:

- Executes the subqueries first
- Returns the result into the HAVING clause of the main query

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

← 2500



	DEPARTMENT_ID	MIN(SALARY)
1	(null)	7000
2	90	17000
3	20	6000
4	110	8300
5	80	8600
6	60	4200
7	10	4400



#	department_id	MIN(salary)
1	90	7000.00
2	10	4400.00
3	20	6000.00
4	60	4200.00
5	80	8600.00
6	90	17000.00
7	110	8300.00

What Is Wrong with This Statement?

```
SELECT employee_id, last_name
FROM   employees
WHERE  salary =
      (SELECT  MIN(salary)
       FROM    employees
       GROUP BY department id);
```


What Is Wrong with This Statement?

```
SELECT employee_id, last_name
FROM   employees
WHERE  salary =
      (SELECT  MIN(salary)
       FROM    employees
       GROUP BY department id);
```



ORA-01427: single-row subquery returns more than one row
ORA-01427. 00000 - "single-row subquery returns more than one row"
*Cause:
*Action:

Single-row operator with multiple-row subquery



Error Code: 1242. Subquery returns more than 1 row

No Rows Returned by the Inner Query

```
SELECT last_name, job_id
FROM employees
WHERE job_id =
    (SELECT job_id
     FROM jobs
     WHERE job_title = 'Architect');
```



Query Result x	
SQL All Rows Fetched: 0 in 0.003 seconds	
LAST_N...	JOB_ID

The subquery returns no rows because there is no job with the title "Architect."



#	last_name	job_id
---	-----------	--------

0 row(s) returned

Lesson Agenda

- Subquery: Types, syntax, and guidelines
- Single-row subqueries:
 - Group functions in a subquery
 - `HAVING` clause with subqueries
- **Multiple-row subqueries**
 - Use `IN`, `ALL`, or `ANY`
- **Multiple-column subqueries**
- Null values in a subquery

Multiple-Row Subqueries

- Return more than one row
- Use multiple-row comparison operators

Operator	Meaning
IN	Equal to any member in the list
ANY	Must be preceded by =, !=, >, <, <=, >=. This returns TRUE if at least one element exists in the result set of the subquery for which the relation is TRUE.
ALL	Must be preceded by =, !=, >, <, <=, >=. This returns TRUE if the relation is TRUE for all elements in the result set of the subquery.

Using the ANY Operator in Multiple-Row Subqueries

```
SELECT employee_id, last_name, job_id, salary
FROM   employees
WHERE  salary < ANY (SELECT salary
                     FROM   employees
                     WHERE  job_id = 'IT_PROG')
AND    job_id <> 'IT_PROG';
```



#	EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
1	144	Vargas	ST_CLERK	2500
2	143	Matos	ST_CLERK	2600
3	142	Davies	ST_CLERK	3100
4	141	Rajs	ST_CLERK	3500
5	200	Whalen	AD_ASST	4400

...

9	206	Gietz	AC_ACCOUNT	8300
10	176	Taylor	SA_REP	8600



#	employee_id	last_name	job_id	salary
1	124	Moungos	ST_MAN	5800.00
2	141	Rajs	ST_CLERK	3500.00
3	142	Davies	ST_CLERK	3100.00
4	143	Matos	ST_CLERK	2600.00
5	144	Vargas	ST_CLERK	2500.00
6	176	Taylor	SA_REP	8600.00
7	178	Grant	SA_REP	7000.00
8	200	Whalen	AD_ASST	4400.00
9	202	Fay	MK_REP	6000.00
10	206	Gietz	AC_ACC...	8300.00
*	***	***	***	***



Using the ALL Operator in Multiple-Row Subqueries

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

9000, 6000, 4200



	EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
1	141	Rajs	ST_CLERK	3500
2	142	Davies	ST_CLERK	3100
3	143	Matos	ST_CLERK	2600
4	144	Vargas	ST_CLERK	2500



#	employee_id	last_name	job_id	salary
1	141	Rajs	ST_CLERK	3500.00
2	142	Davies	ST_CLERK	3100.00
3	143	Matos	ST_CLERK	2600.00
4	144	Vargas	ST_CLERK	2500.00
*	max	max	max	max



Multiple-Column Subqueries

- A multiple-column subquery returns more than one column to the outer query.
- Column comparisons in multiple column comparisons can be pairwise or nonpairwise.
- A multiple-column subquery can also be used in the `FROM` clause of a `SELECT` statement.

Syntax:

```
SELECT column, column, ...  
FROM table  
WHERE (column1, column2, ...) IN  
      (SELECT column1, column2, ...  
       FROM table  
       WHERE condition);
```

Multiple-Column Subquery: Example

Display all the employees with the lowest salary in each department.

```
SELECT first_name, department_id, salary
FROM employees
WHERE (salary, department_id) IN
    (SELECT min(salary), department_id
     FROM employees
     GROUP BY department_id)
ORDER BY department_id;
```



#	FIRST_NAME	DEPARTMENT_ID	SALARY
1	Jennifer	10	4400
2	Pat	20	6000
3	Peter	50	2500
4	Diana	60	4200
5	Jonathon	80	8600
6	Neena	90	17000
7	Lex	90	17000
8	William	110	8300



#	first_name	department_id	salary
1	Jennifer	10	4400.00
2	Pat	20	6000.00
3	Peter	50	2500.00
4	Diana	60	4200.00
5	Jonathon	80	8600.00
6	Neena	90	17000.00
7	Lex	90	17000.00
8	William	110	8300.00

Null Values in a Subquery

```
SELECT emp.last_name  
FROM   employees emp  
WHERE  emp.employee_id NOT IN  
        (SELECT mgr.manager_id  
         FROM   employees mgr);
```



Query Result x	
SQL All Rows Fetched: 0 in 0.051 seconds	
LAST_NAME	



#	last_name
---	-----------

0 row(s) returned

The subquery returns no rows because one of the values returned by a subquery is null.

Summary

In this lesson, you should have learned how to:

- Define subqueries
- Identify the types of problems that subqueries can solve
- Identify the types of subqueries
- Write single-row, multiple-row, multiple-column subqueries



Practice 8: Overview

This practice covers the following topics:

- Creating subqueries to query values based on unknown criteria
- Using subqueries to find out the values that exist in one set of data and not in another