SQL Part 4

Perancangan Basis Data Relasional

Outline

- Overview
- Subquery syntax
- Single row subquery
- Multiple row subquery

Using a Subquery to Solve a Problem

Who has a salary greater than Abel's?

Main Query:

Which employees have salaries greater than Abel's salary?

Subquery

What is Abel's salary?

Using a Subquery to Solve a Problem

- To solve this problem, you need *two* queries: one to find what Abel earns, and a second query to find who earns more than that amount.
- You can solve this problem by combining the two queries, placing one query *inside* the other query.
- The inner query or the *subquery* returns a value that is used by the outer query or the main query.
- Using a subquery is equivalent to performing two sequential queries and using the result of the first query as the search value in the second query.

Subquery Syntax

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

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

Subquery Syntax

– You can place the subquery in a number of SQL clauses, including:
  - The `WHERE` clause
  - The `HAVING` clause
  - The `FROM` clause
– In the syntax:
  `operator` includes a comparison condition such as >, =, or `IN`

Using a Subquery

```
SELECT last_name
FROM   employees 11000
WHERE  salary >
                (SELECT salary
                 FROM   employees
                 WHERE  last_name = 'Abel');
```

LAST_NAME
King
Kochhar
De Haan
Hartstein
Higgins

Guidelines for Using Subqueries

- Enclose subqueries in parentheses.
- Place subqueries on the right side of the comparison condition.
- Use single-row operators with single-row subqueries and use multiple-row operators with multiple-row subqueries.

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 =
                (SELECT job_id
                 FROM   employees
                 WHERE  employee_id = 141)
AND    salary >
                (SELECT salary
                 FROM   employees
                 WHERE  employee_id = 143);
```

ST_CLERK

2600

LAST_NAME	JOB_ID	SALARY
Rajs	ST_CLERK	3500
Davies	ST_CLERK	3100

Using Group Functions in a Subquery

- You can display data from a main query by using a group function in a subquery to return a single row.
- The subquery is in parentheses and is placed after the comparison condition.

Using Group Functions in a Subquery

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

2500

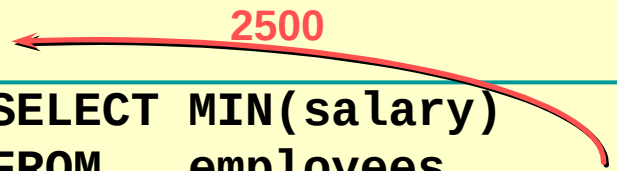
LAST_NAME	JOB_ID	SALARY
Vargas	ST_CLERK	2500

The HAVING Clause with Subqueries

- You can use subqueries not only in the WHERE clause, but also in the HAVING clause.
- DBMS executes the subquery, and the results are returned into the HAVING clause of the main query.

The HAVING Clause with Subqueries

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



What is Wrong with this Statement?

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

```
ERROR at line 4:
ORA-01427: single-row subquery returns more than
one row
```

Single-row operator with multiple-row subquery

Errors with Subqueries

- One common error with subqueries is more than one row returned for a single-row subquery.
- The `WHERE` clause contains an equal (=) operator, a single-row comparison operator expecting only one value.
- The = operator cannot accept more than one value from the subquery and therefore generates the error.

Will this Statement Return Rows?

```
SELECT last_name, job_id
FROM   employees
WHERE  job_id =
                (SELECT job_id
                 FROM   employees
                 WHERE  last_name = 'Haas');
```

```
no rows selected
```

Subquery returns no values

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);

no rows selected
```

Multiple-Row Subqueries

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

Operator	Meaning
IN	Equal to any member in the list
ANY	Compare value to each value returned by the subquery
ALL	Compare value to every value returned by 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';
```

9000, 6000, 4200

EMPLOYEE_ID	LAST_NAME	JOB_ID	SALARY
124	Mourgos	ST_MAN	5800
141	Rajs	ST_CLERK	3500
142	Davies	ST_CLERK	3100
143	Matos	ST_CLERK	2600
144	Vargas	ST_CLERK	2500

...

10 rows selected.

<ANY means less than the maximum.

>ANY means more than the minimum.

=ANY is equivalent to IN.

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
141	Rajs	ST_CLERK	3500
142	Davies	ST_CLERK	3100
143	Matos	ST_CLERK	2600
144	Vargas	ST_CLERK	2500

>ALL means more than the maximum,

<ALL means less than the minimum.

Review

- Overview
- Subquery syntax
- Single row subquery
- Multiple row subquery