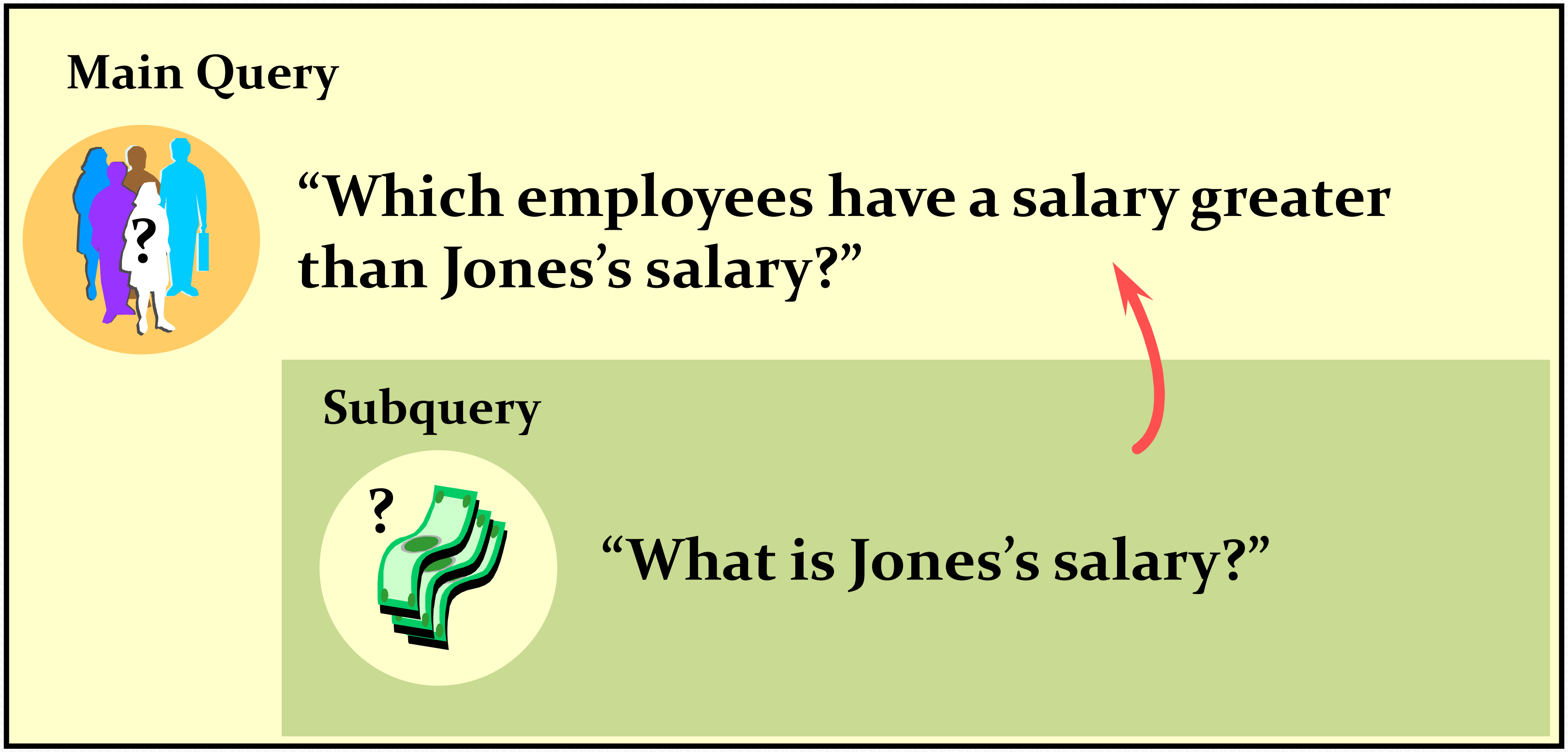
Subqueries

Objectives

- Using Subqueries
- Identifying different types of subqueries.
- Multiple row operators
- Multiple column subqueries.
- Subquery in from clause

Subquery to Solve a Problem

- "Who has a salary greater than Jones's?"



Main Query

"Which employees have a salary greater than Jones's salary?"

Subquery

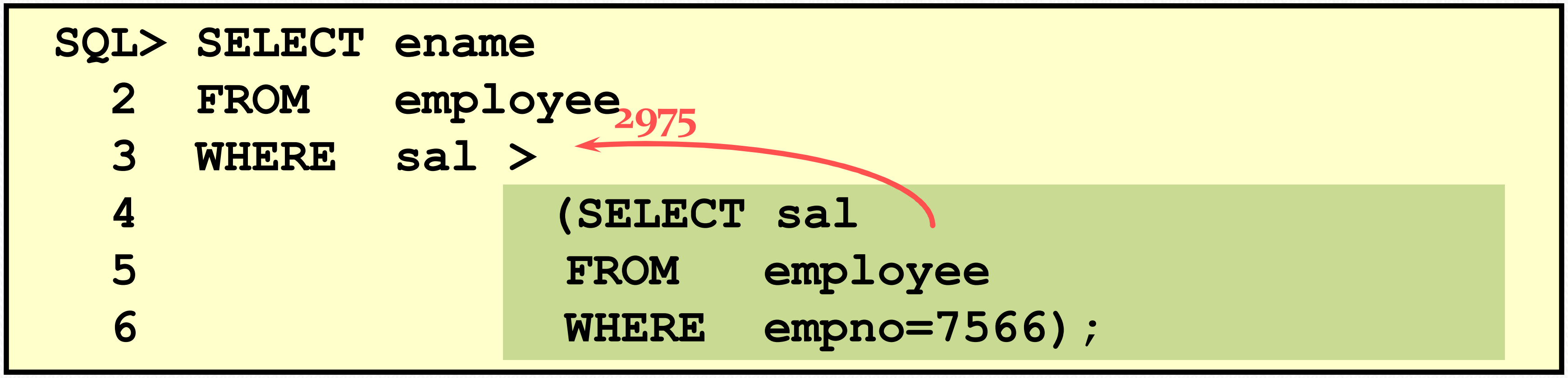
"What is Jones's salary?"

Subqueries

```
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).

Using a Subquery



```
SQL> SELECT ename
  2  FROM   employee
  3  WHERE  sal >
  4                   (SELECT  sal
  5                    FROM    employee
  6                    WHERE   empno=7566);
```

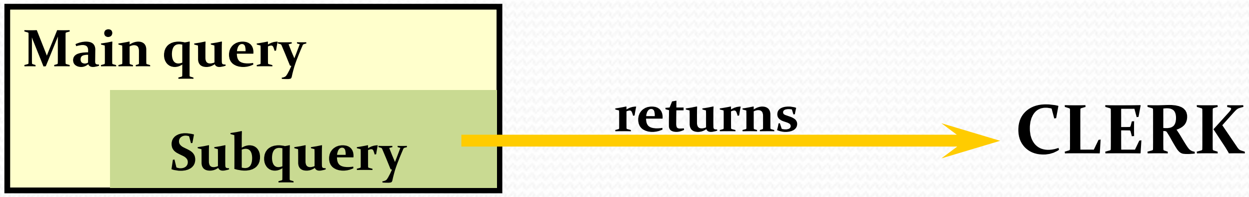
```
ENAME
----------
KING
FORD
SCOTT
```

Guidelines for Subqueries

- Enclose subqueries in parentheses.
- Place subqueries on the right side of the comparison operator.
- Do not add an ORDER BY clause to a subquery.
- Use single-row operators with single-row subqueries.
- Use multiple-row operators with multiple-row subqueries.

Types of Subqueries

- Single-row subquery



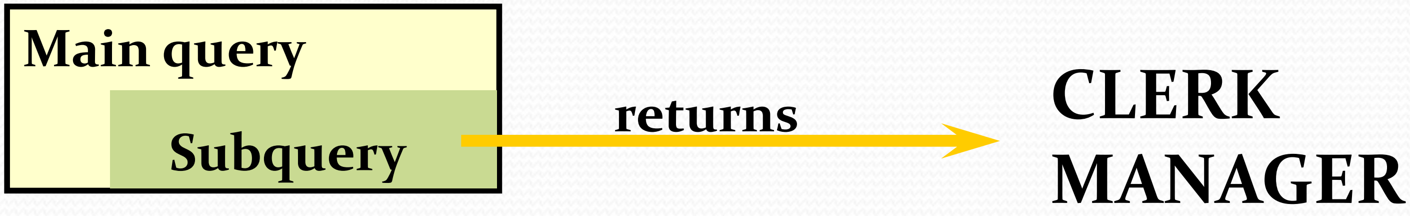
Main query

Subquery

returns

CLERK

- Multiple-row subquery

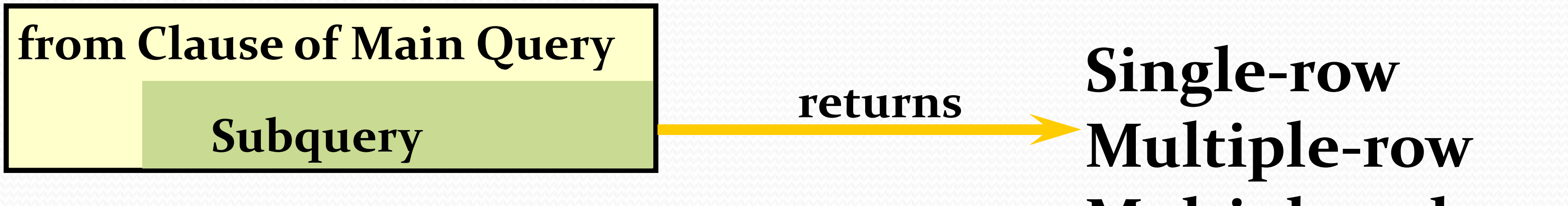


Main query

Subquery

returns

CLERK
MANAGER



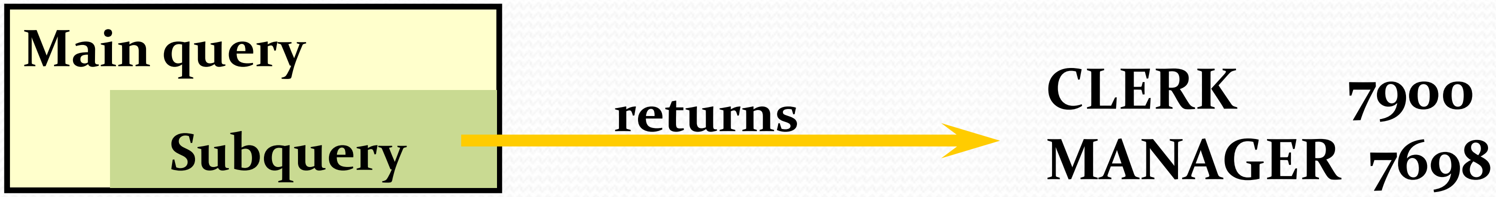
from Clause of Main Query

Subquery

returns

Single-row
Multiple-row
Multiple-column

- Multiple-column subquery



Main query

Subquery

returns

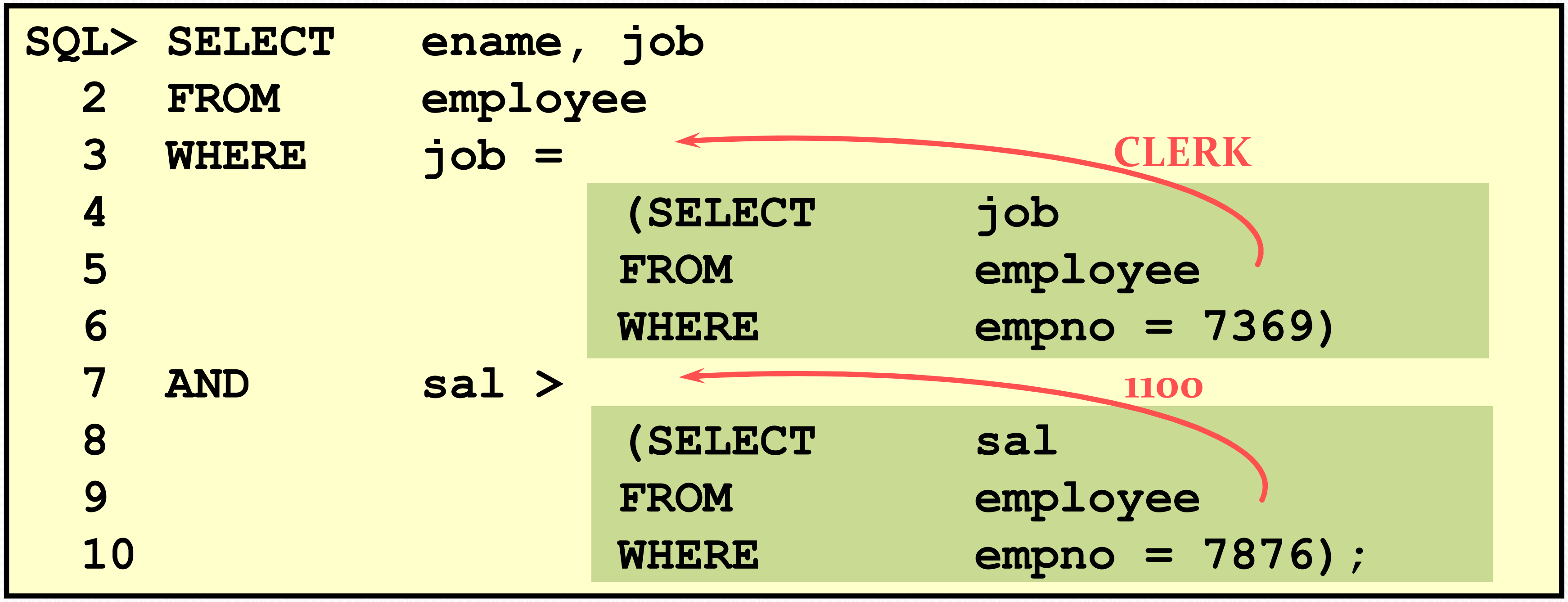
CLERK 7900
MANAGER 7698

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

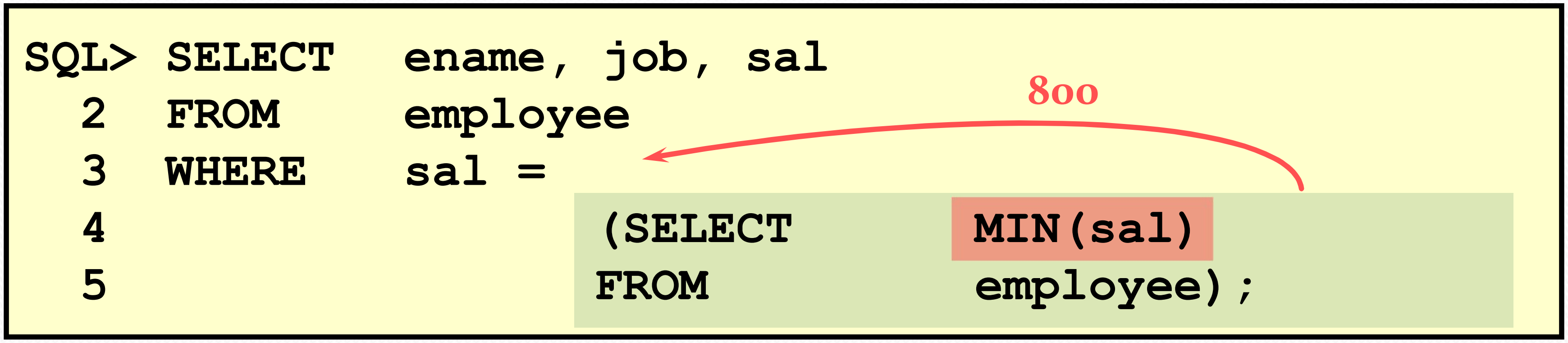
Single-Row Subqueries



```
SQL> SELECT   ename, job
  2  FROM     employee
  3  WHERE    job =
  4                   (SELECT      job
  5                   FROM         employee
  6                   WHERE        empno = 7369)
  7  AND      sal >
  8                   (SELECT      sal
  9                   FROM         employee
 10                   WHERE        empno = 7876);
```

```
ENAME      JOB
---------- ---------
MILLER     CLERK
```

Group Functions in Subquery

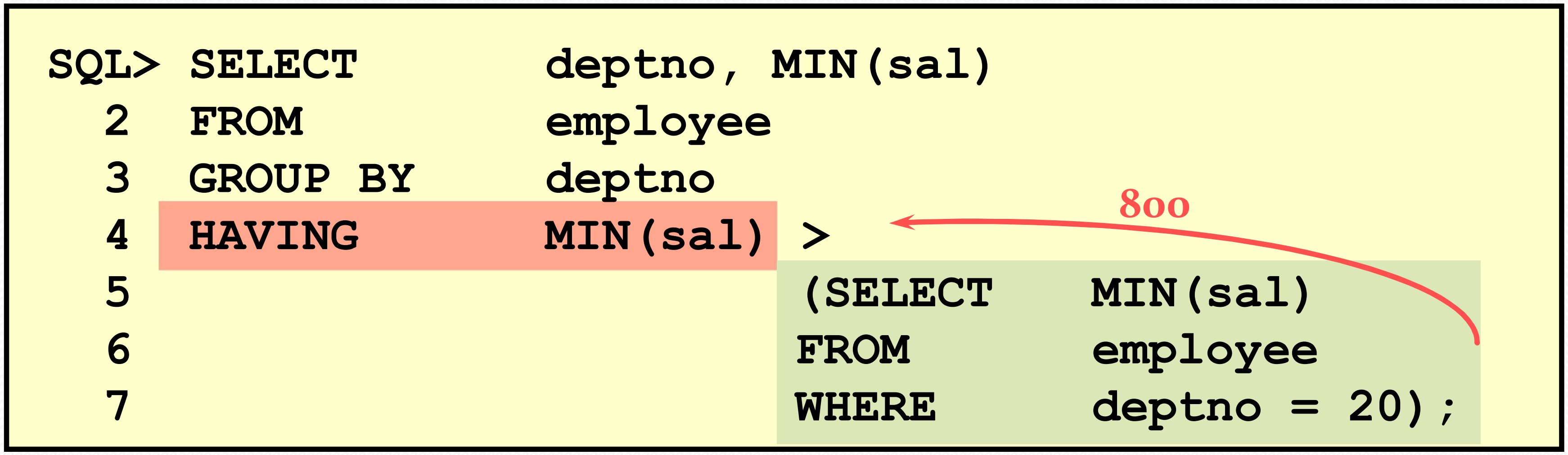


```
SQL> SELECT   ename, job, sal
  2  FROM     employee
  3  WHERE    sal =
  4                (SELECT      MIN(sal)
  5                FROM         employee);
```

```
ENAME      JOB              SAL
---------- --------- ----------
SMITH      CLERK            800
```

HAVING with Subqueries

- The Oracle Server executes subqueries first.



```
SQL> SELECT       deptno, MIN(sal)
  2  FROM         employee
  3  GROUP BY     deptno
  4  HAVING       MIN(sal) >
  5                          (SELECT    MIN(sal)
  6                          FROM       employee
  7                          WHERE      deptno = 20);
```

What Is Wrong ?

```
SQL> SELECT empno, ename
  2  FROM   employee
  3  WHERE  sal =
  4                     (SELECT   MIN(sal)
  5                     FROM      employee
  6                     GROUP BY  deptno);
```

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

no rows selected
```

Single-row operator with multiple-row subquery

Will This Statement Work?

```
SQL> SELECT ename, job
  2  FROM   employee
  3  WHERE  job =
  4                  (SELECT job
  5                  FROM    employee
  6                  WHERE   ename='SMYTHE');
```

```
no rows selected
```

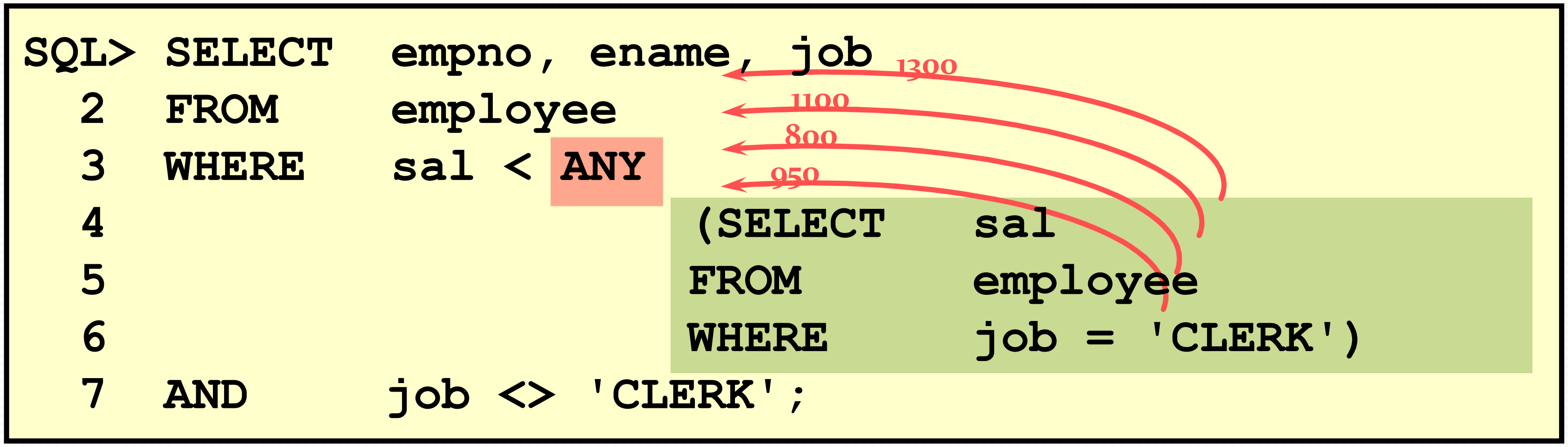
Subquery returns no values

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

ANY: Multiple-Row Subqueries



```
SQL> SELECT   empno, ename, job
  2  FROM     employee
  3  WHERE    sal < ANY
  4                    (SELECT    sal
  5                    FROM       employee
  6                    WHERE      job = 'CLERK')
  7  AND      job <> 'CLERK';
```

```
     EMPNO ENAME      JOB
---------- ---------- ---------
      7654 MARTIN     SALESMAN
      7521 WARD       SALESMAN
```

ALL: Multiple-Row Subqueries

```
SQL> SELECT   empno, ename, job
  2  FROM     employee
  3  WHERE    sal > ALL
  4                 (SELECT       avg(sal)
  5                 FROM          employee
  6                 GROUP BY      deptno);
```

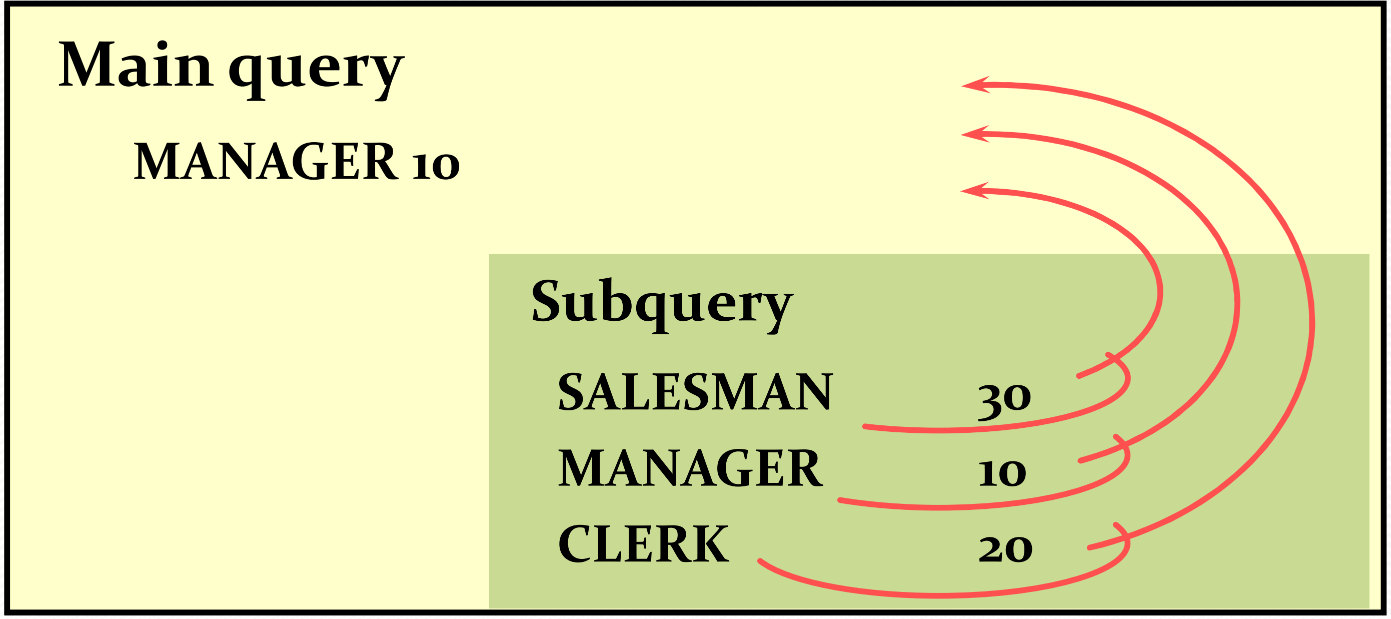
1566.6667

2175

2916.6667

```
     EMPNO ENAME      JOB
---------- ---------- ---------
      7839 KING       PRESIDENT
      7566 JONES      MANAGER
      7902 FORD       ANALYST
      7788 SCOTT      ANALYST
```

Multiple-Column Subqueries



Main query

MANAGER 10

Subquery

SALESMAN 30

MANAGER 10

CLERK 20

Main query compares **to** **Values from a multiple-row and multiple-column subquery**

MANAGER 10

SALESMAN	30
MANAGER	10
CLERK	20

Using Multiple-Column Subqueries

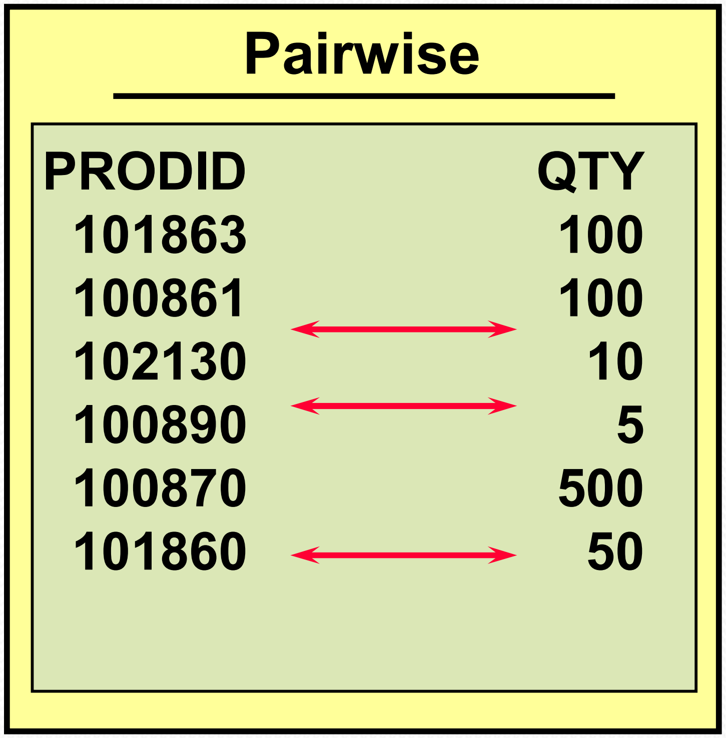
- Display the order number, product number, and quantity of any item in which the product number and quantity match both the product number and quantity of an item in order 605.

```
SQL> SELECT   ordid, prodid, qty
  2  FROM     item
  3  WHERE    (prodid, qty) IN
  4                         (SELECT prodid, qty
  5                          FROM   item
  6                          WHERE  ordid = 605)
  7  AND      ordid <> 605;
```

Pair wise

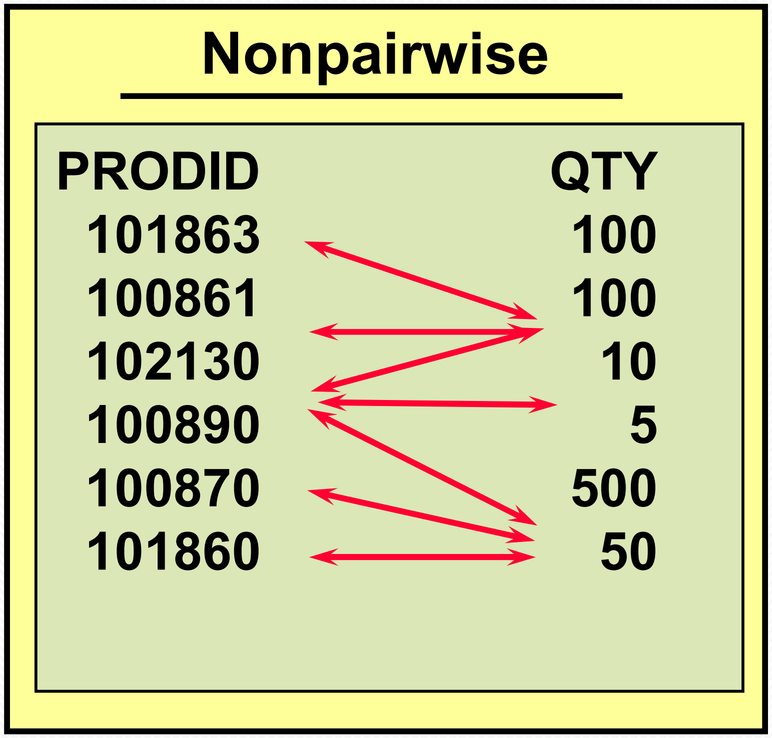
```
select ename,deptno,sal
from emp
where (deptno,sal) in (select deptno,sal from emp
where comm is not null);
```

Column Comparisons



Pairwise

PRODID	QTY
101863	100
100861	100
102130	10
100890	5
100870	500
101860	50



Nonpairwise

PRODID	QTY
101863	100
100861	100
102130	10
100890	5
100870	500
101860	50

Nonpairwise Comparison Subquery

- Display the order number, product number, and quantity of any item in which the product number and quantity match any product number and any quantity of an item in order 605.

```
SQL> SELECT   ordid, prodid, qty
  2  FROM     item
  3  WHERE    prodid IN  (SELECT          prodid
  4                        FROM           item
  5                        WHERE          ordid = 605)
  6  AND      qty    IN  (SELECT          qty
  7                       FROM            item
  8                       WHERE           ordid = 605)
  9  AND      ordid <> 605;
```

Non Pair wise

```
select ename,deptno,sal
from emp
where deptno in (select deptno from emp where comm
   is not null)
and sal in (select sal from emp where comm is not
   null);
```

Nonpairwise Subquery

```
     ORDID     PRODID        QTY
---------- ---------- ----------
       609     100870          5
       616     100861         10
       616     102130         10
       621     100861         10
       618     100870         10
       618     100861         50
       616     100870         50
       617     100861        100
       619     102130        100
       615     100870        100
       617     101860        100
       621     100870        100
       617     102130        100
       .  .  .
16 rows selected.
```

Correlated Subquery

- The outer Query is executed first and then the inner query is executed.
- Find the employee list who earn more than the avg salary of their own department
- Example:

```
Select * from emp X
Where sal > (Select avg(sal)
    from emp where x.deptno=deptno)
```

Exists operator

- Find the list of employees who has subordinates.

```
Select * from emp e
Where exists(select 'X' from emp
Where mgr=e.empno);
```

Exists alternative

```
Select * from emp
Where empno in (select mgr from emp where mgr is not null);
select distinct a.ename from emp a,emp b
where a.empno=b.mgr;
 select empno from emp where empno in (select mgr from emp);
```

Not Exists

- Find the list of employees who has manager

```
Select * from emp e
Where not exists(select 'X' from emp
Where mgr=e.empno)
```

Non Exists Alternative

```
 Select * from emp
Where empno not in (select mgr from emp where
   mgr is not null)
```

Subquery in FROM Clause

```
SQL> SELECT  a.ename, a.sal, a.deptno, b.salavg
  2  FROM    employee a, (SELECT   deptno, avg(sal) salavg
  3                       FROM     employee
  4                       GROUP BY deptno) b
  5  WHERE   a.deptno = b.deptno
  6  AND     a.sal > b.salavg;
```

```
ENAME            SAL    DEPTNO     SALAVG
---------- --------- --------- ----------
KING            5000        10  2916.6667
JONES           2975        20       2175
SCOTT           3000        20       2175
...
6 rows selected.
```

Summary

- Writing a query using subquery.
- Use single row operator with single-row subqueries
- Use multiple-row operators with multiple-row subqueries.
- Using group function in subqueries.
- Using an inline subquery.(subquery in from clause).