

7

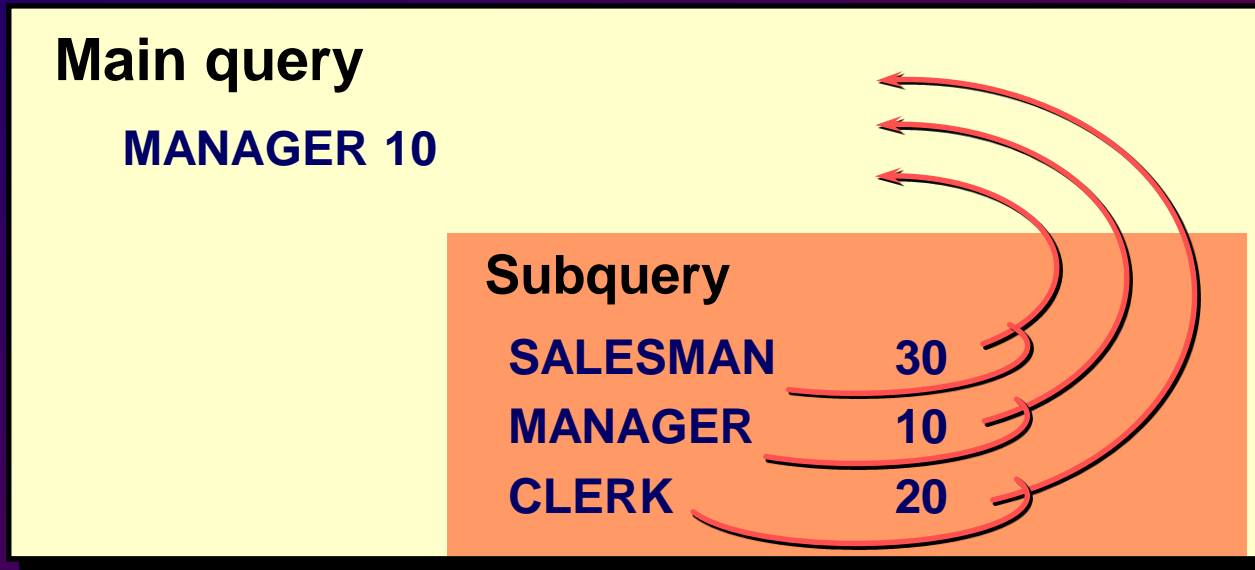
Multiple-Column Subqueries

Objectives

At the end of this lesson, you should be able to:

- **Write a multiple-column subquery**
- **Describe and explain the behavior of subqueries when null values are retrieved**
- **Write a subquery in a FROM clause**

Multiple-Column Subqueries



**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 name, department number, salary, and commission of any employee whose salary and commission matches **both** the commission and salary of any employee in department 30.

```
SQL> SELECT      ename, deptno, sal, comm
  2  FROM          emp
  3  WHERE         (sal, NVL(comm,-1)) IN
  4                (SELECT sal, NVL(comm,-1)
  5                  FROM    emp
  6                  WHERE   deptno = 30) ;
```

Column Comparisons

Pairwise

SAL		COMM
1600	↔	300
1250	↔	500
1250	↔	1400
2850		
1500	↔	0
950		

Nonpairwise

SAL		COMM
1600	↔	300
1250	↔	500
1250	↔	1400
2850	↔	0
1500	↔	0
950		

Nonpairwise Comparison Subquery

Display the name, department number, salary, and commission of any employee whose salary and commission matches the commission and salary of any employee in department 30.

```
SQL> SELECT      ename, deptno, sal, comm
  2  FROM          emp
  3  WHERE          sal IN              (SELECT  sal
  4                                         FROM    emp
  5                                         WHERE   deptno = 30)
  6  AND
  7              NVL(comm, -1) IN      (SELECT  NVL(comm, -1)
  8                                         FROM    emp
  9                                         WHERE   deptno = 30) ;
```

Modifying the EMP Table

- Assume that salary and commission for Clark are modified.
- Salary is changed to \$1500 and commission to \$300.

ENAME	SAL	COMM
...		
CLARK	1500	300
...		
ALLEN	1600	300
TURNER	1500	0
...		

14 rows selected.

The diagram illustrates the modification of the EMP table. Red boxes highlight the updated values for Clark: SAL=1500 and COMM=300. Red arrows indicate the flow of data: one arrow points from the new SAL value (1500) to the old SAL value (1600) for Clark, and another arrow points from the new COMM value (300) to the old COMM value (0) for Clark. Additionally, a red arrow points from the new SAL value (1500) to the old SAL value (1500) for Allen, and another red arrow points from the new COMM value (300) to the old COMM value (300) for Allen.

Pairwise Subquery

```
SQL> SELECT  ename, deptno, sal, comm
  2  FROM      emp
  3  WHERE      (sal, NVL(comm,-1)) IN
  4              (SELECT sal, NVL(comm,-1)
  5                  FROM      emp
  6                  WHERE      deptno = 30) ;
```

ENAME	DEPTNO	SAL	COMM
JAMES	30	950	
WARD	30	1250	500
MARTIN	30	1250	1400
TURNER	30	1500	0
ALLEN	30	1600	300
BLAKE	30	2850	

6 rows selected.

Nonpairwise Subquery

```

SQL> SELECT      ename,deptno, sal, comm
      2 FROM      emp
      3 WHERE      sal IN      (SELECT sal
      4                                FROM      emp
      5                                WHERE      deptno = 30)
      6 AND
      7            NVL(comm,-1) IN (SELECT NVL(comm,-1)
      8                                FROM      emp
      9                                WHERE      deptno = 30);
  
```

ENAME	DEPTNO	SAL	COMM
JAMES	30	950	
BLAKE	30	2850	
TURNER	30	1500	0
CLARK	10	1500	300
...			

7 rows selected.

Null Values in a Subquery

```
SQL> SELECT  employee.ename
      2  FROM    emp employee
      3  WHERE   employee.empno NOT IN
                (SELECT manager.mgr
                 FROM    emp manager) ;
```

no rows selected.

Using a Subquery in the FROM Clause

```
SQL> SELECT  a.ename, a.sal, a.deptno, b.salavg
  2  FROM    emp a, (SELECT  deptno, avg(sal) salavg
  3                      FROM    emp
  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

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

Practice Overview

Creating multiple-column subqueries