Nested Subqueries

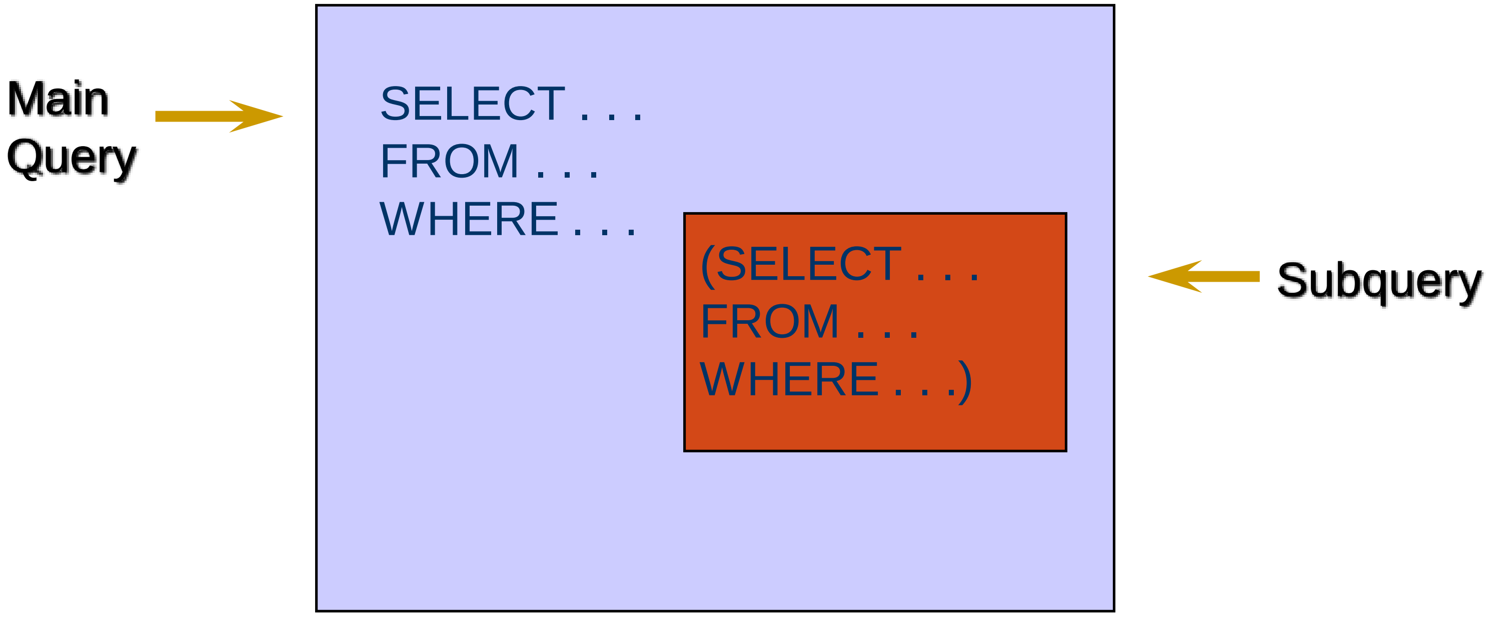
Subqueries and Their Uses

- A subquery is a query within another query. The outer query is called as **main query** and inner query is called as **subquery** Used when a query is based on an unknown value
- Must be enclosed in parentheses
- Place on right side of comparison operator
- Subquery can be placed in a number of SQL clauses: WHERE clause, HAVING clause, FROM clause.
- Subqueries can be used with SELECT, UPDATE, INSERT, DELETE statements along with expression operator. It could be equality operator or comparison operator such as =, >, =, <= and Like operator.

- The subquery generally executes first, and its output is used to complete the query condition for the main or outer query.
- ORDER BY command **cannot** be used in a Subquery. GROUPBY command can be used to perform same function as ORDER BY command.
- Use single-row operators with single row Subqueries. Use multiple-row operators with multiple-row Subqueries.

Subquery

- A subquery is a SELECT statement embedded in a clause of another SQL statement.



Main Query

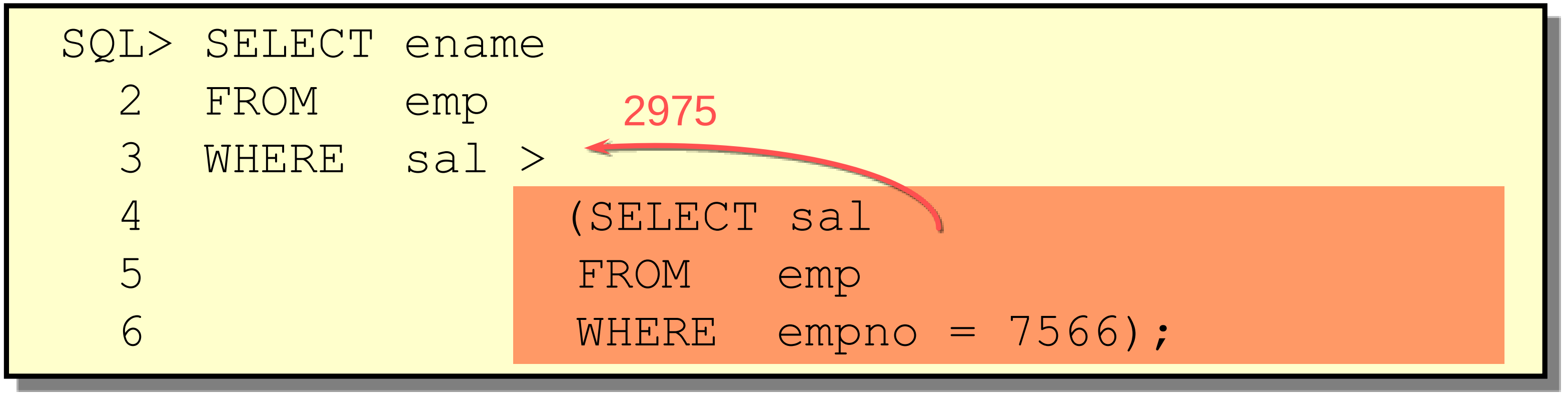
```
SELECT . . .
FROM . . .
WHERE . . .
          (SELECT . . .
          FROM . . .
          WHERE . . .)
```

Subquery

Subqueries

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

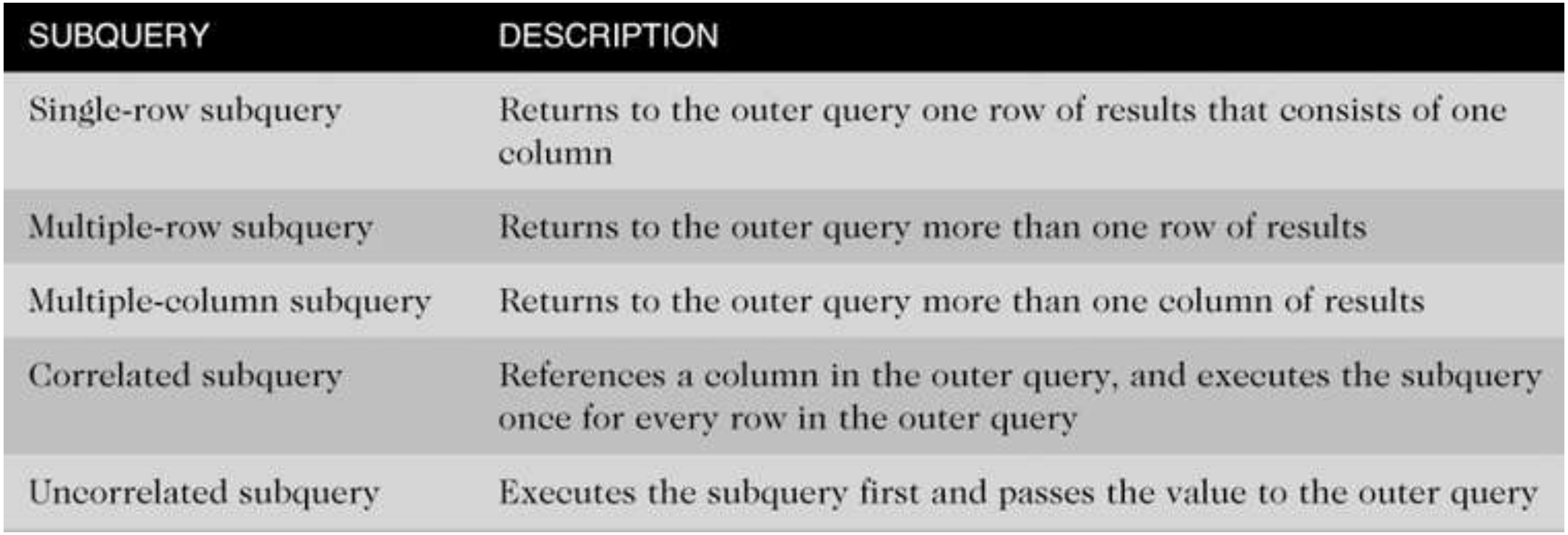
Using a Subquery



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

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

Types of Subqueries



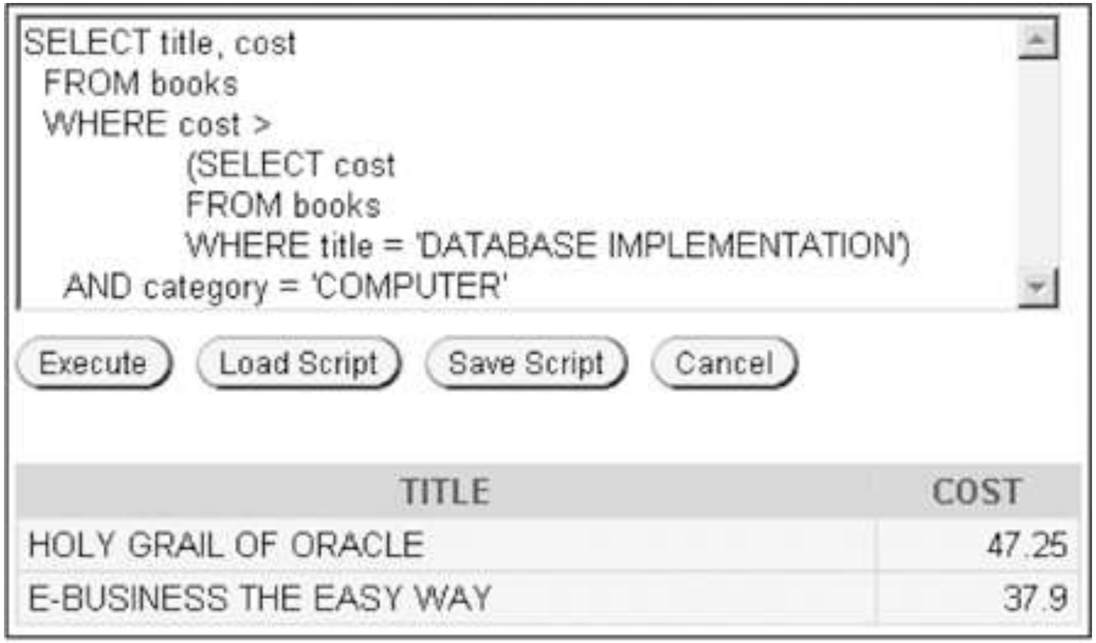
SUBQUERY	DESCRIPTION
Single-row subquery	Returns to the outer query one row of results that consists of one column
Multiple-row subquery	Returns to the outer query more than one row of results
Multiple-column subquery	Returns to the outer query more than one column of results
Correlated subquery	References a column in the outer query, and executes the subquery once for every row in the outer query
Uncorrelated subquery	Executes the subquery first and passes the value to the outer query

Single-Row Subqueries

- Can only return one result to the outer query
- Operators include =, >, <, >=, <=, < >

Single-Row Subquery in a WHERE Clause

- Used for comparison against individual data



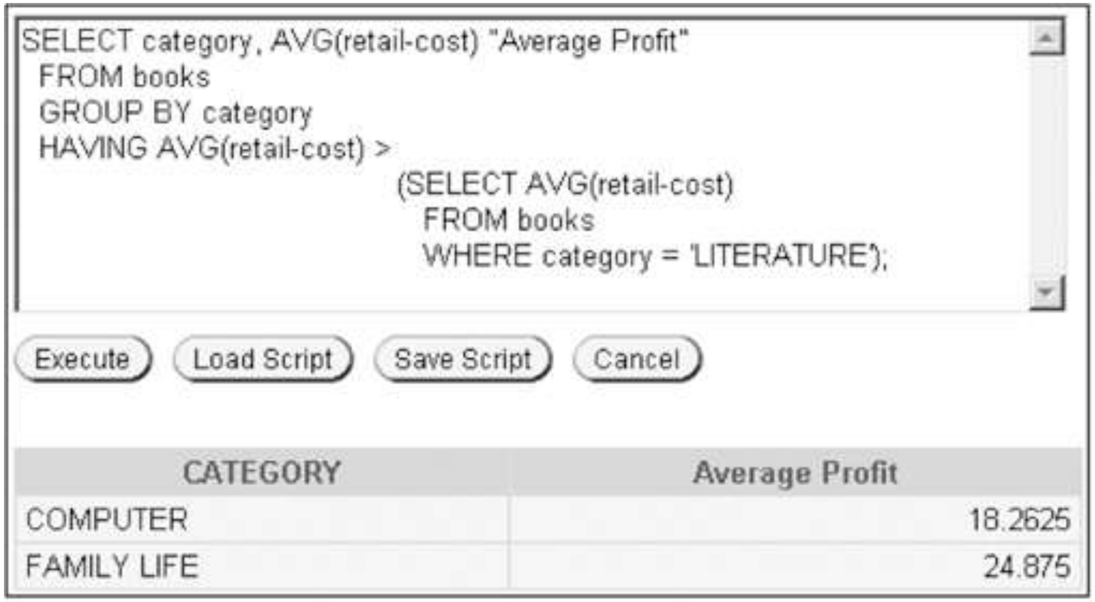
```
SELECT title, cost
 FROM books
 WHERE cost >
          (SELECT cost
          FROM books
          WHERE title = 'DATABASE IMPLEMENTATION')
  AND category = 'COMPUTER'
```

TITLE	COST
HOLY GRAIL OF ORACLE	47.25
E-BUSINESS THE EASY WAY	37.9

FIGURE 12-4 A single-row subquery

Single-Row Subquery in a HAVING Clause

- Required when returned value is compared to grouped data



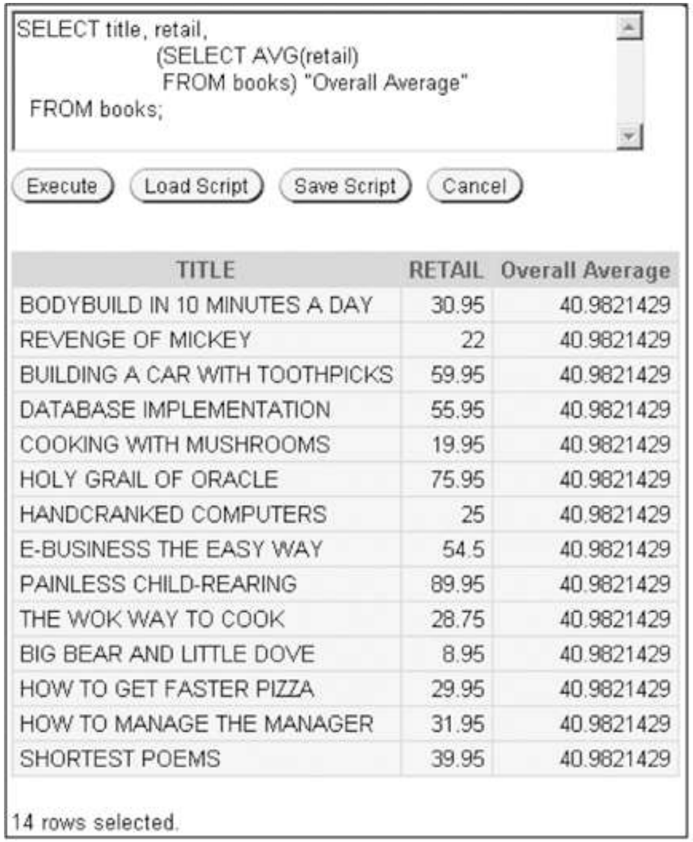
```
SELECT category, AVG(retail-cost) "Average Profit"
  FROM books
  GROUP BY category
  HAVING AVG(retail-cost) >
                                        (SELECT AVG(retail-cost)
                                           FROM books
                                           WHERE category = 'LITERATURE');
```

CATEGORY	Average Profit
COMPUTER	18.2625
FAMILY LIFE	24.875

FIGURE 12-8 Single-row subquery nested in a HAVING clause

Single-Row Subquery in a SELECT Clause

- Replicates subquery value for each row displayed



```
SELECT title, retail,
               (SELECT AVG(retail)
                FROM books) "Overall Average"
 FROM books;
```

TITLE	RETAIL	Overall Average
BODYBUILD IN 10 MINUTES A DAY	30.95	40.9821429
REVENGE OF MICKEY	22	40.9821429
BUILDING A CAR WITH TOOTHPICKS	59.95	40.9821429
DATABASE IMPLEMENTATION	55.95	40.9821429
COOKING WITH MUSHROOMS	19.95	40.9821429
HOLY GRAIL OF ORACLE	75.95	40.9821429
HANDCRANKED COMPUTERS	25	40.9821429
E-BUSINESS THE EASY WAY	54.5	40.9821429
PAINLESS CHILD-REARING	89.95	40.9821429
THE WOK WAY TO COOK	28.75	40.9821429
BIG BEAR AND LITTLE DOVE	8.95	40.9821429
HOW TO GET FASTER PIZZA	29.95	40.9821429
HOW TO MANAGE THE MANAGER	31.95	40.9821429
SHORTEST POEMS	39.95	40.9821429

14 rows selected.

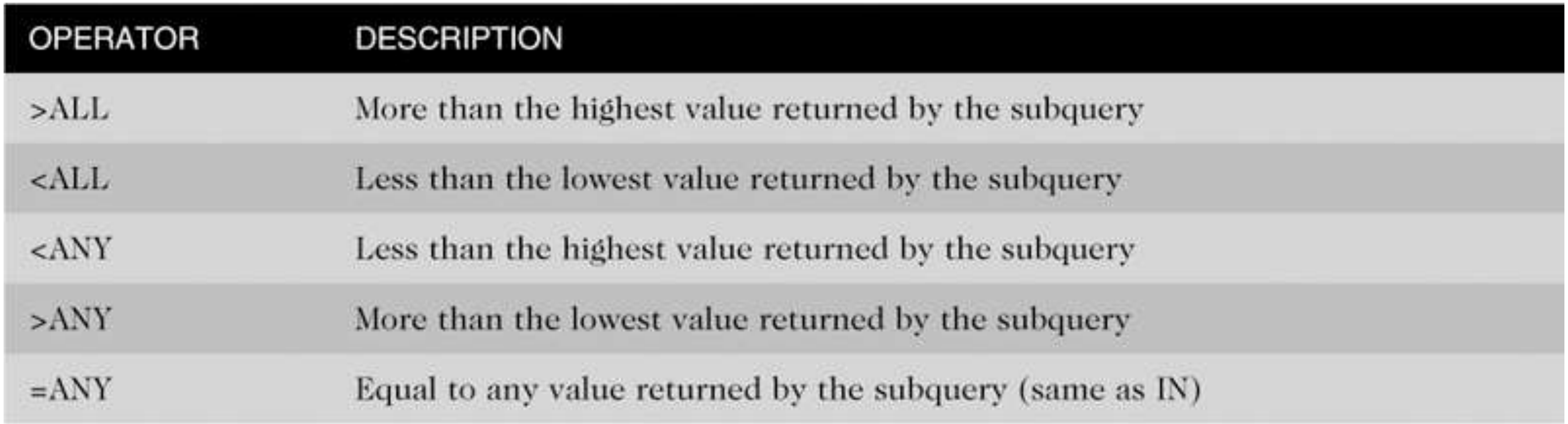
FIGURE 12-9 Single-row subquery in a SELECT clause

Multiple-Row Subqueries

- Return more than one row of results
- Require use of IN, ANY, ALL, or EXISTS operators

ANY and ALL Operators

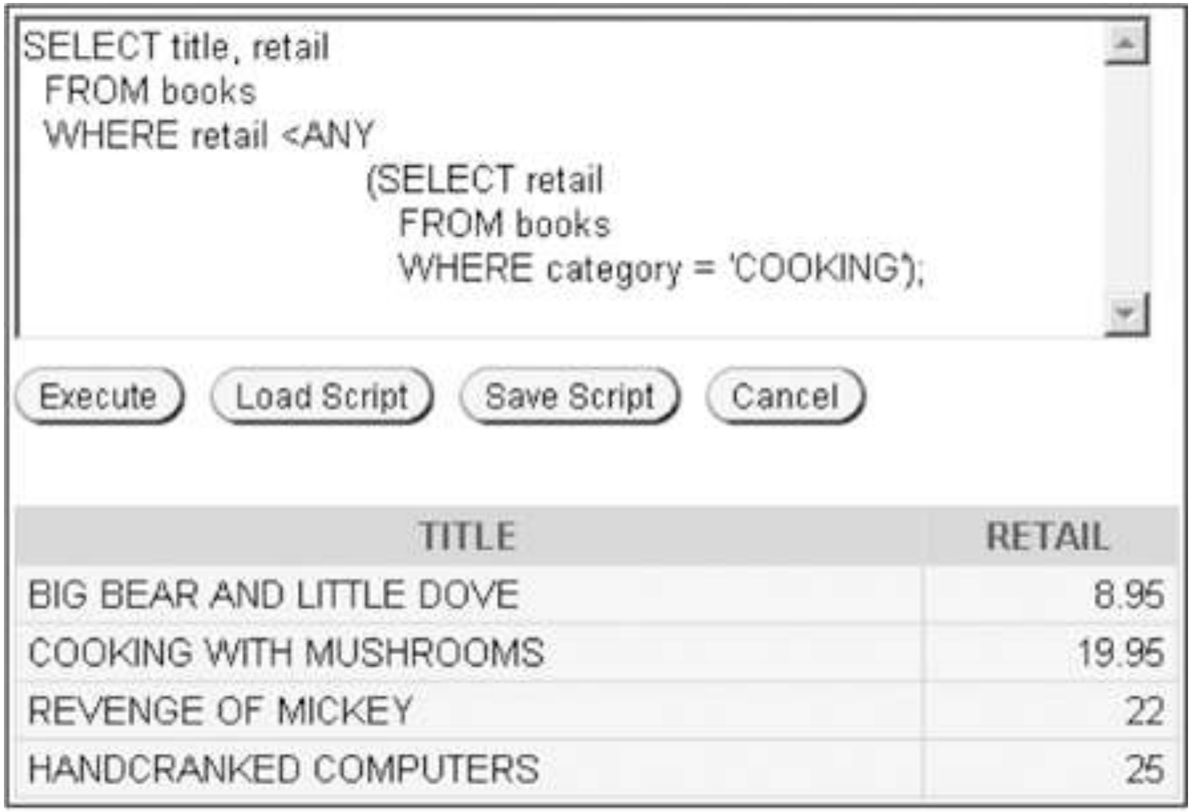
- Combine with arithmetic operators



OPERATOR	DESCRIPTION
>ALL	More than the highest value returned by the subquery
<ALL	Less than the lowest value returned by the subquery
<ANY	Less than the highest value returned by the subquery
>ANY	More than the lowest value returned by the subquery
=ANY	Equal to any value returned by the subquery (same as IN)

FIGURE 12-11 Descriptions of ALL and ANY operator combinations

Multiple-Row Subquery in a WHERE Clause (continued)



```
SELECT title, retail
  FROM books
  WHERE retail <ANY
                        (SELECT retail
                           FROM books
                           WHERE category = 'COOKING');
```

TITLE	RETAIL
BIG BEAR AND LITTLE DOVE	8.95
COOKING WITH MUSHROOMS	19.95
REVENGE OF MICKEY	22
HANDCRANKED COMPUTERS	25

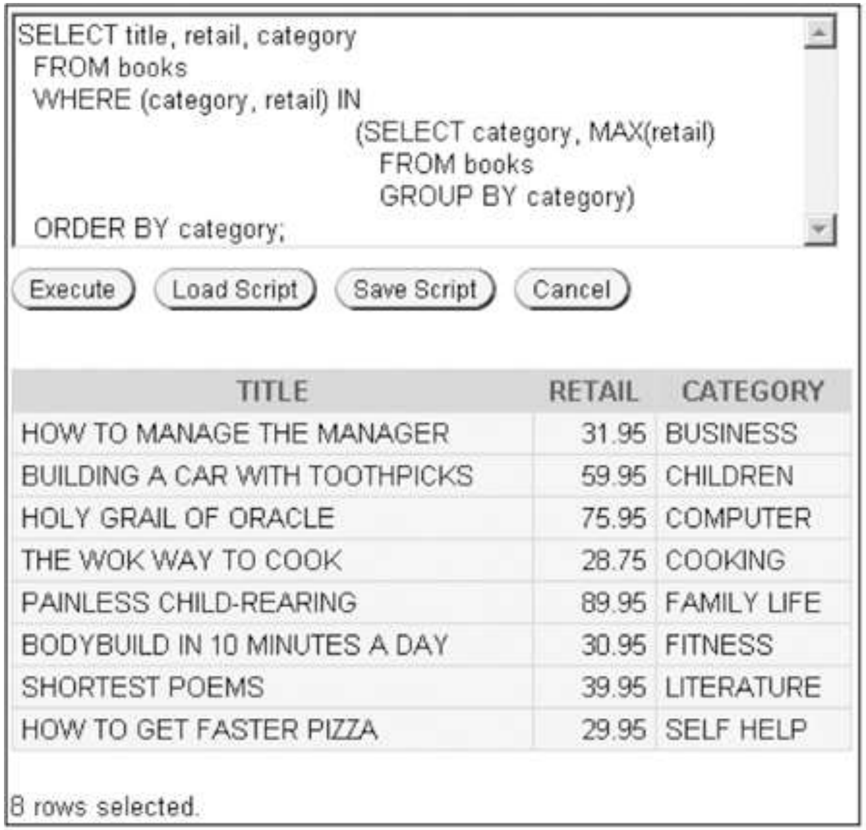
FIGURE 12-15 The <ANY operator

Multiple-Column Subqueries

- Return more than one column in results
- Can return more than one row
- Column list on the left side of operator must be in parentheses
- Use the IN operator for WHERE and HAVING clauses

Multiple-Column Subquery in a WHERE Clause

- Returns multiple columns for evaluation



```sql
SELECT title, retail, category
  FROM books
  WHERE (category, retail) IN
                                          (SELECT category, MAX(retail)
                                             FROM books
                                             GROUP BY category)
  ORDER BY category;
```

TITLE	RETAIL	CATEGORY
HOW TO MANAGE THE MANAGER	31.95	BUSINESS
BUILDING A CAR WITH TOOTHPICKS	59.95	CHILDREN
HOLY GRAIL OF ORACLE	75.95	COMPUTER
THE WOK WAY TO COOK	28.75	COOKING
PAINLESS CHILD-REARING	89.95	FAMILY LIFE
BODYBUILD IN 10 MINUTES A DAY	30.95	FITNESS
SHORTEST POEMS	39.95	LITERATURE
HOW TO GET FASTER PIZZA	29.95	SELF HELP

```
8 rows selected.
```

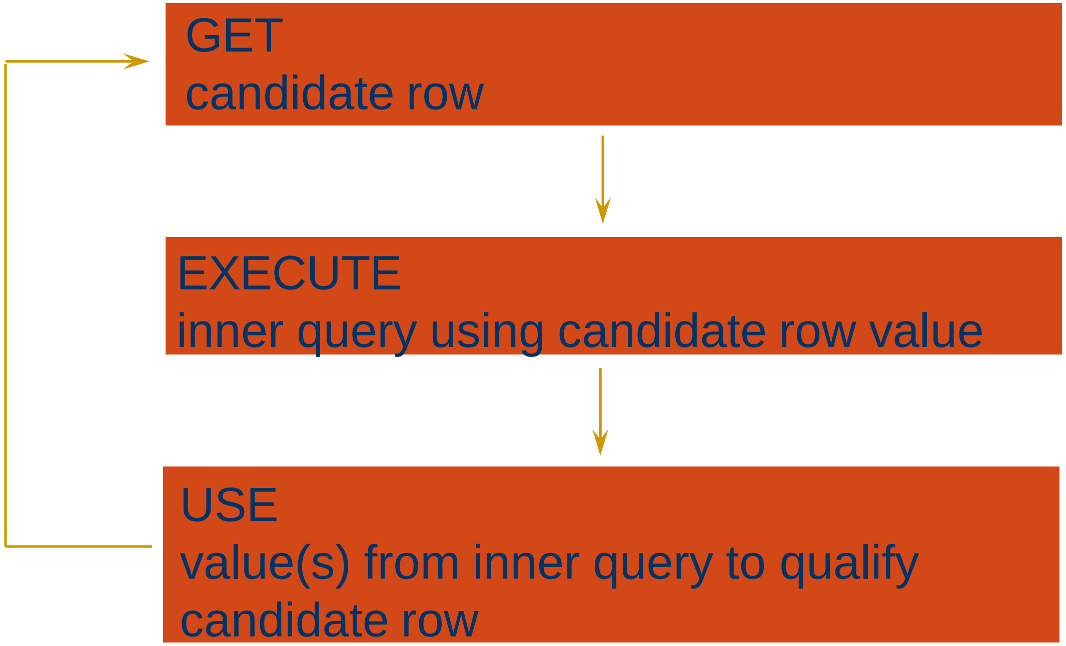
FIGURE 12-24 Multiple-column subquery in a WHERE clause

Uncorrelated Subqueries

- Processing sequence:
  - Inner query is executed first
  - Result is passed to outer query
  - Outer query is executed

Correlated Subqueries

- Used to affect row-by-row processing, each subquery is executed once for every row of the outer query.



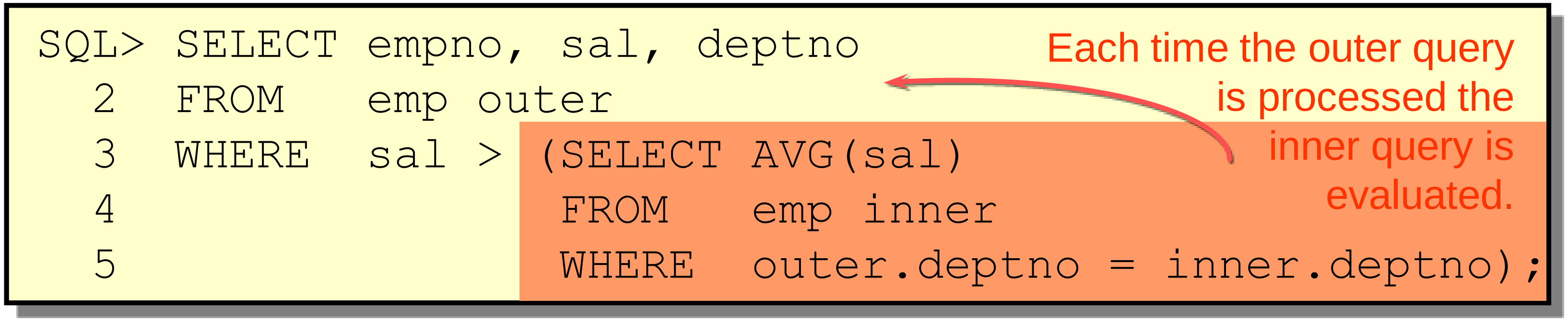
GET
candidate row

EXECUTE
inner query using candidate row value

USE
value(s) from inner query to qualify candidate row

Using Correlated Subqueries

- Find all employees who make more than the average salary in their department.



```
SQL> SELECT empno, sal, deptno
  2  FROM   emp outer
  3  WHERE  sal > (SELECT AVG(sal)
  4                FROM   emp inner
  5                WHERE  outer.deptno = inner.deptno);
```

Each time the outer query is processed the inner query is evaluated.

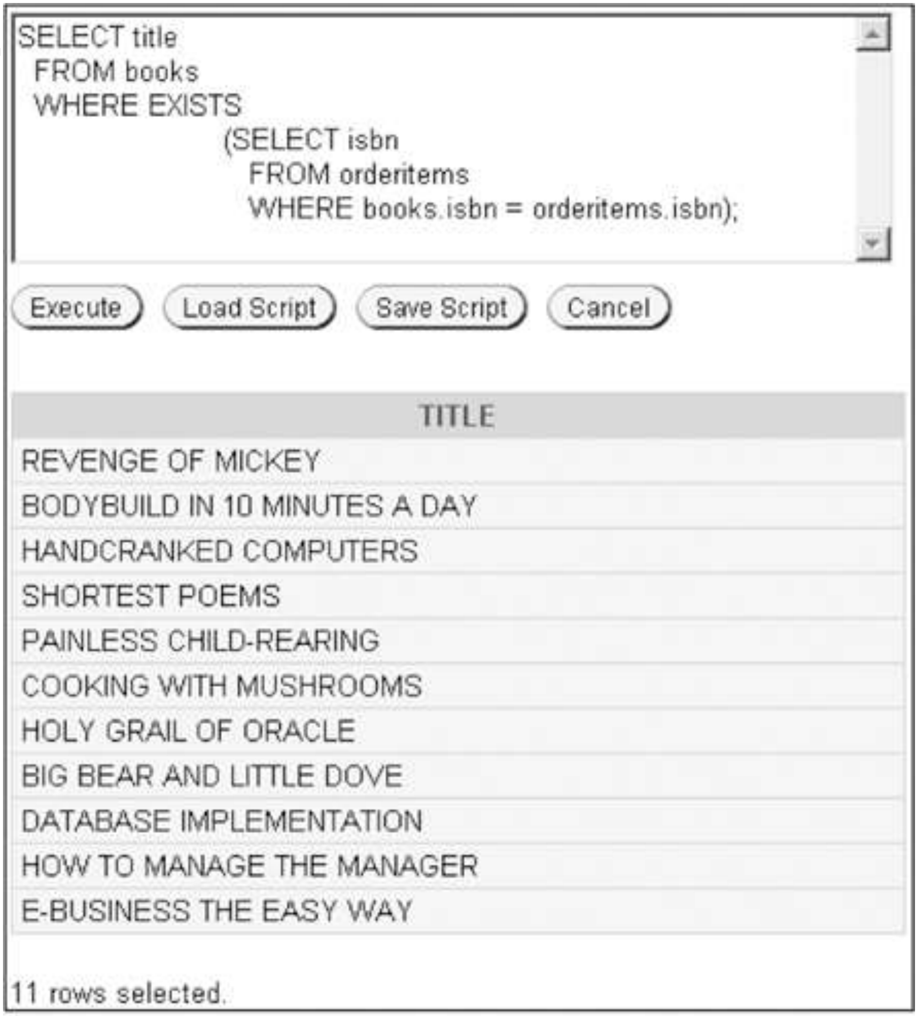
```
   EMPNO       SAL    DEPTNO
-------- --------- ---------
    7839      5000        10
    7698      2850        30
    7566      2975        20
 ...
 6 rows selected.
```

Using the EXISTS Operator

- If a subquery row value is found:
  - The search does not continue in the inner query.
  - The condition is flagged TRUE.
- If a subquery row value is not found:
  - The condition is flagged FALSE.
  - The search continues in the inner query.

EXISTS Operator

- Determines whether condition exists in subquery



```
SELECT title
  FROM books
  WHERE EXISTS
                (SELECT isbn
                   FROM orderitems
                   WHERE books.isbn = orderitems.isbn);
```

TITLE
REVENGE OF MICKEY
BODYBUILD IN 10 MINUTES A DAY
HANDCRANKED COMPUTERS
SHORTEST POEMS
PAINLESS CHILD-REARING
COOKING WITH MUSHROOMS
HOLY GRAIL OF ORACLE
BIG BEAR AND LITTLE DOVE
DATABASE IMPLEMENTATION
HOW TO MANAGE THE MANAGER
E-BUSINESS THE EASY WAY

```
11 rows selected.
```

FIGURE 12-18 Subquery using the EXISTS operator

Using the EXISTS Operator

Find employees who have at least one person reporting to them.

```
SQL> SELECT empno, ename, job, deptno
  2  FROM   emp outer
  3  WHERE  EXISTS (SELECT empno
  4                 FROM   emp inner
  5                 WHERE  inner.mgr = outer.empno);
```

```
    EMPNO ENAME      JOB          DEPTNO
--------- ---------- --------- ---------
     7839 KING       PRESIDENT        10
     7698 BLAKE      MANAGER          30
     7782 CLARK      MANAGER          10
     7566 JONES      MANAGER          20
...
6 rows selected.
```

Using the NOT EXISTS Operator

Find all departments that do not have any employees.

```
SQL> SELECT   deptno, dname
  2  FROM     dept d
  3  WHERE    NOT EXISTS (SELECT '1'
  4                       FROM   emp e
  5                       WHERE  d.deptno = e.deptno);
```

```
   DEPTNO DNAME
--------- ----------
       40 OPERATIONS
```

Correlated UPDATE

```
UPDATE table1 alias1
SET    column = (SELECT expression
                 FROM   table2 alias2
                 WHERE  alias1.column = alias2.column);
```

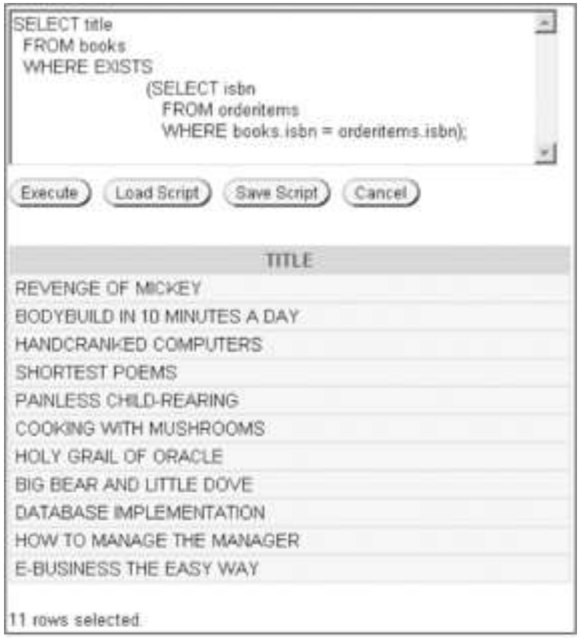
- Use a correlated subquery to update rows in one table based on rows from another table.

Correlated DELETE

```
DELETE FROM table1 alias1
WHERE  column operator
            (SELECT expression
            FROM   table2 alias2
            WHERE  alias1.column = alias2.column);
```

Use a correlated subquery to delete only those rows that also exist in another table.

Correlated Subqueries



```
SELECT title
  FROM books
  WHERE EXISTS
                    (SELECT isbn
                       FROM orderitems
                       WHERE books.isbn = orderitems.isbn);
```

TITLE
REVENGE OF MICKEY
BODYBUILD IN 10 MINUTES A DAY
HANDCRANKED COMPUTERS
SHORTEST POEMS
PAINLESS CHILD-REARING
COOKING WITH MUSHROOMS
HOLY GRAIL OF ORACLE
BIG BEAR AND LITTLE DOVE
DATABASE IMPLEMENTATION
HOW TO MANAGE THE MANAGER
E-BUSINESS THE EASY WAY

11 rows selected.

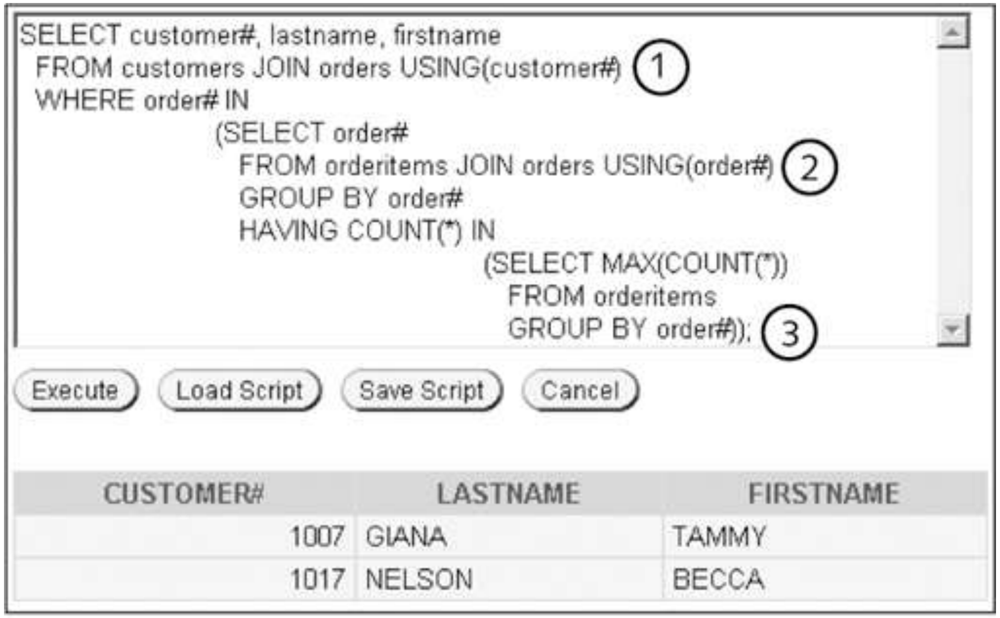
FIGURE 12-28 Correlated subquery

Nested Subqueries

- Maximum of 255 subqueries if nested in the WHERE clause
- No limit if nested in the FROM clause
- Innermost subquery is resolved first, then the next level, etc.

Nested Subqueries

- Innermost is resolved first (3), then the second level (2), then the outer query (1)



```
SELECT customer#, lastname, firstname
  FROM customers JOIN orders USING(customer#)
  WHERE order# IN
                    (SELECT order#
                       FROM orderitems JOIN orders USING(order#)
                       GROUP BY order#
                       HAVING COUNT(*) IN
                                          (SELECT MAX(COUNT(*))
                                             FROM orderitems
                                             GROUP BY order#));
```

CUSTOMER#	LASTNAME	FIRSTNAME
1007	GIANA	TAMMY
1017	NELSON	BECCA

FIGURE 12-30 Nested subqueries