Database Programming with SQL

10-3: Multiple-Row Subqueries

Practice Activities Objectives

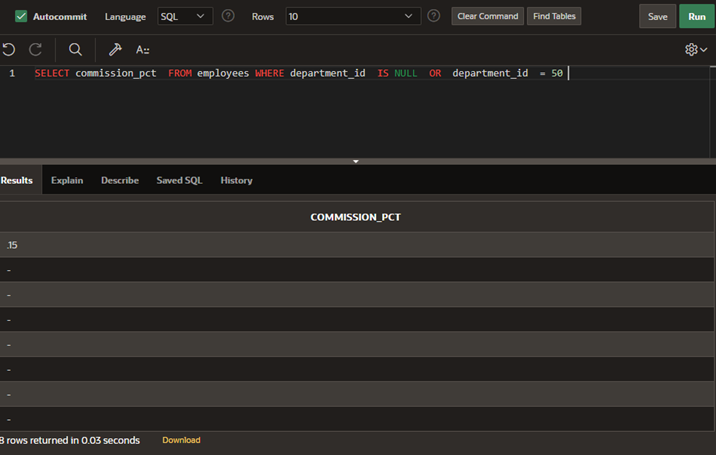
* Correctly use the comparison operators IN, ANY, and ALL in multiple-row subqueries
* Describe what happens if a multiple-row subquery returns a null value
* Construct and execute a multiple-row subquery in the WHERE clause or HAVING clause
* Understand when multiple-row subqueries should be used, and when it is safe to use a singlerow subquery
* Distinguish between pair-wise and non-pair-wise subqueries
* Create a query using the EXISTS and NOT EXISTS operators to test for returned rows from the subquery

# Try It / Solve It

1. What will be returned by a query if it has a subquery that returns a null ?

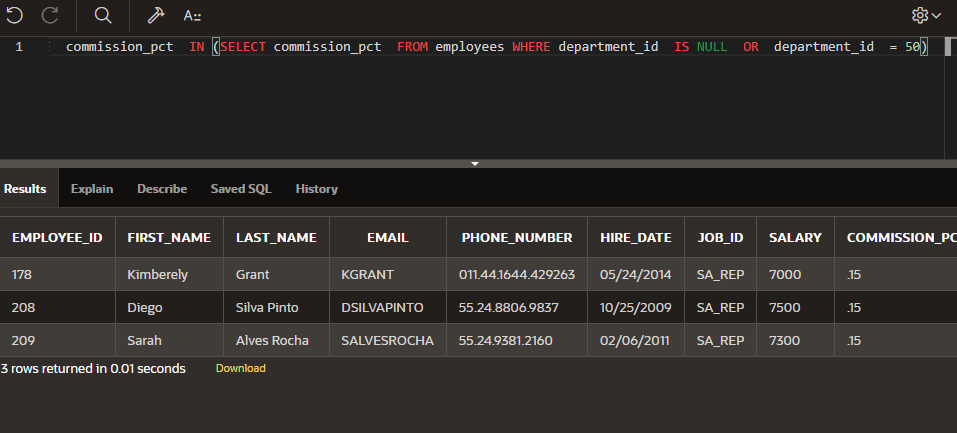
SELECT commission\_pct FROM employees WHERE department\_id IS NULL OR department\_id = 50

Получим

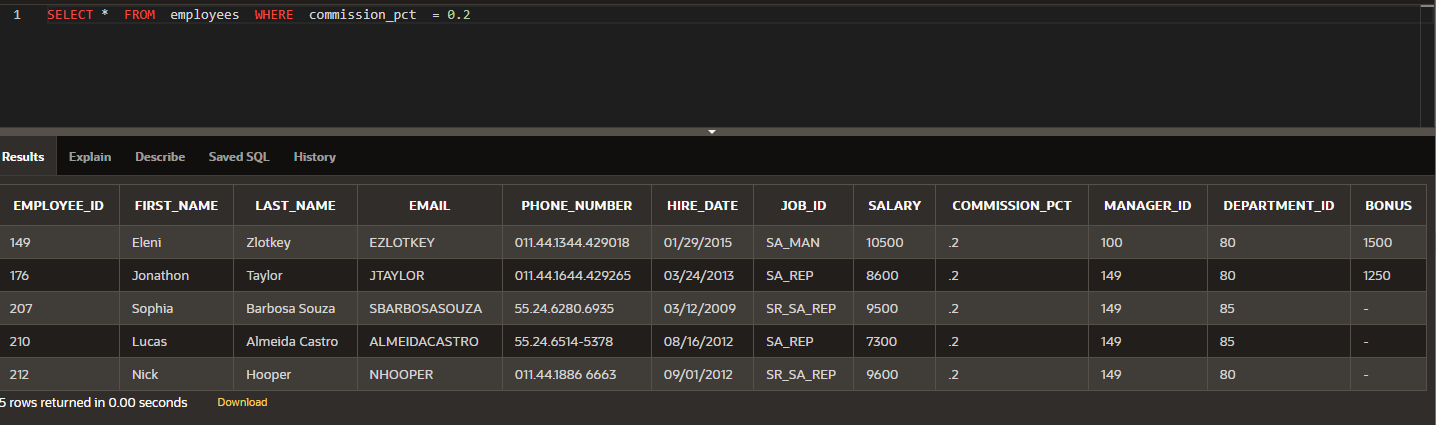


SELECT \* FROM employees WHERE commission\_pct IN (SELECT commission\_pct FROM employees WHERE department\_id IS NULL OR department\_id = 50)

выведет

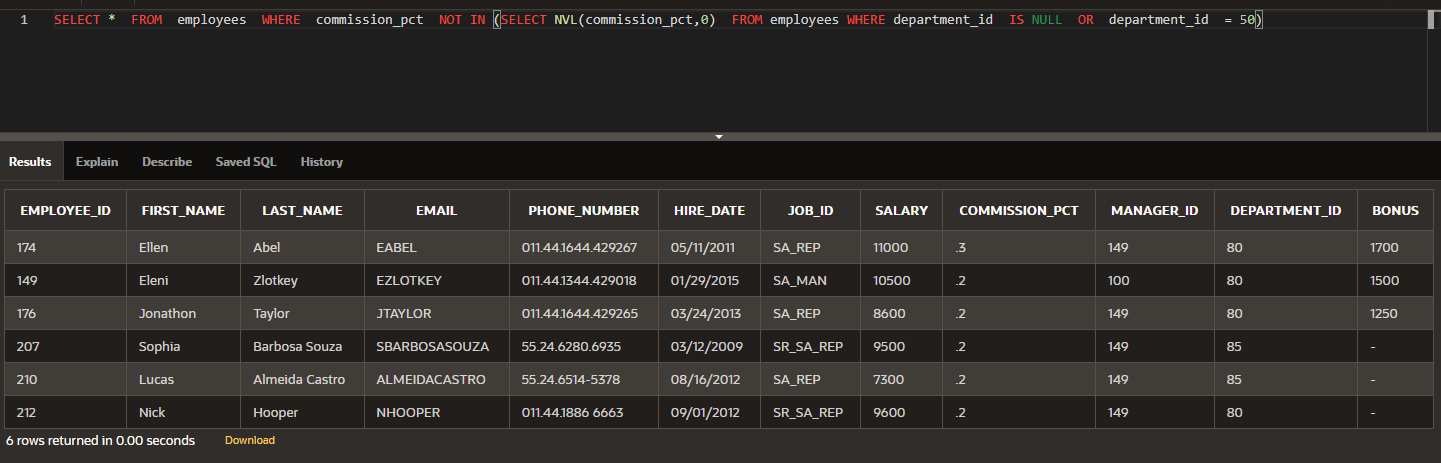


SELECT \* FROM employees WHERE commission\_pct = 0.2 дают



SELECT \* FROM employees WHERE commission\_pct NOT IN (SELECT NVL(commission\_pct,0) FROM employees WHERE department\_id IS NULL OR department\_id = 50)

получим



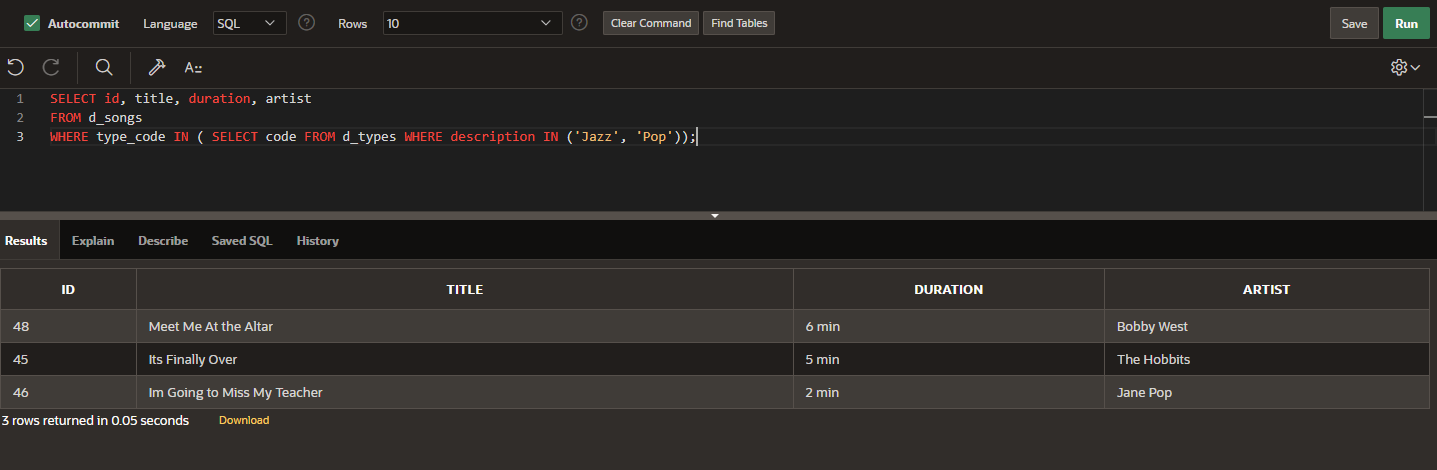
1. Write a query that returns jazz and pop songs. Write a multi-row subquery and use the d\_songs and d\_types tables. Include the id, title, duration, and the artist name.



SELECT id, title, duration, artist

FROM d\_songs

WHERE type\_code IN ( SELECT code FROM d\_types WHERE description IN ('Jazz', 'Pop'));

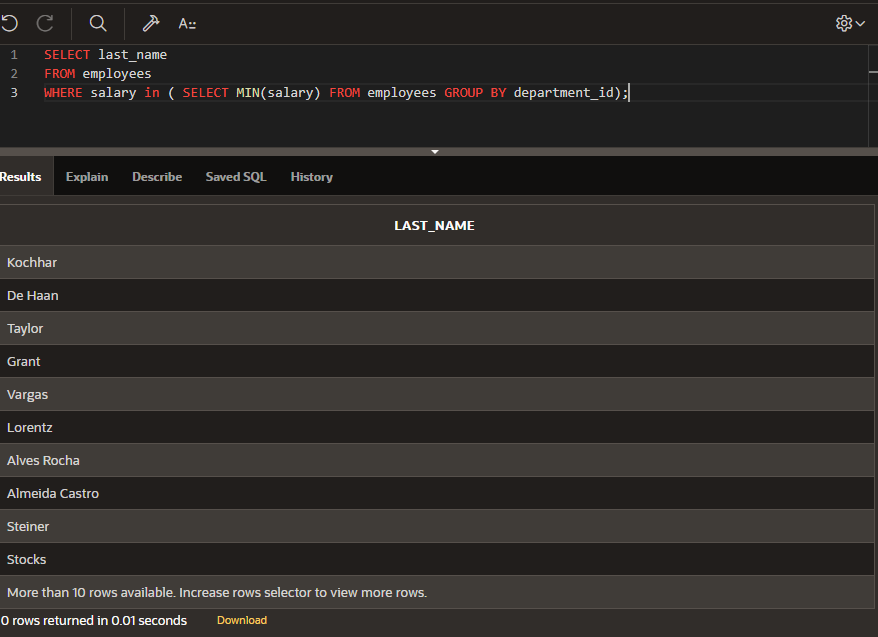


1. Find the last names of all employees whose salaries are the same as the minimum salary for any department.

SELECT last\_name

FROM employees

WHERE salary in ( SELECT MIN(salary) FROM employees GROUP BY department\_id);

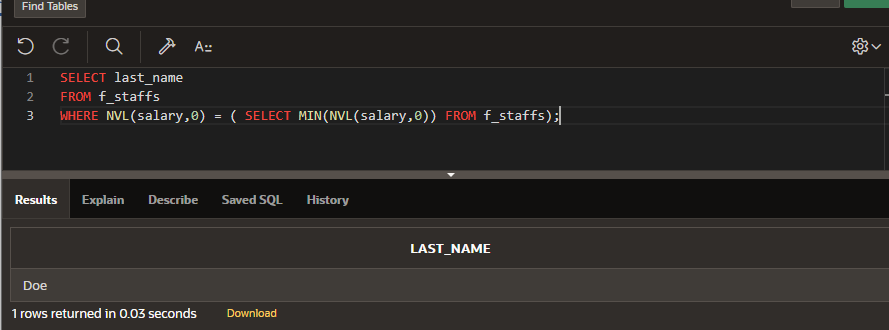


1. Which Global Fast Foods employee earns the lowest salary? Hint: You can use either a singlerow or a multiple-row subquery.

SELECT last\_name

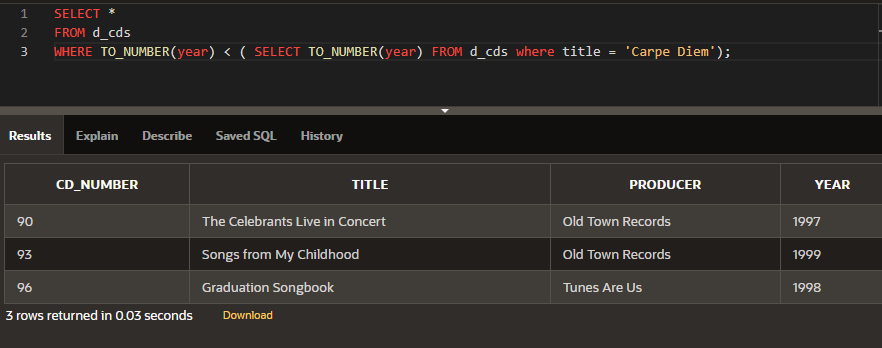
FROM f\_staffs

WHERE NVL(salary,0) = ( SELECT MIN(NVL(salary,0)) FROM f\_staffs);



1. Place the correct multiple-row comparison operators in the outer query WHERE clause of each of the following:
   1. Which CDs in our d\_cds collection were produced before “Carpe Diem” was produced? WHERE year \_\_\_\_\_\_\_\_\_\_ (SELECT year ...

YEAR is VARCHAR2(4), but results with TO\_NUMBER and without it are the same.



* 1. Which employees have salaries lower than any one of the programmers in the IT department? WHERE salary \_\_\_\_\_\_\_\_\_\_(SELECT salary ...

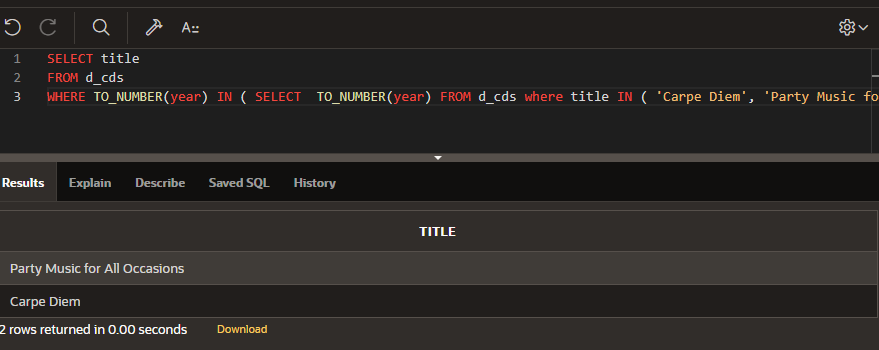
* 1. What CD titles were produced in the same year as “Party Music for All Occasions” or “Carpe Diem”?

WHERE year \_\_\_\_\_\_\_\_\_\_(SELECT year ...

SELECT title

FROM d\_cds

WHERE TO\_NUMBER(year) IN ( SELECT TO\_NUMBER(year) FROM d\_cds where title IN ( 'Carpe Diem', 'Party Music for All Occasions'));



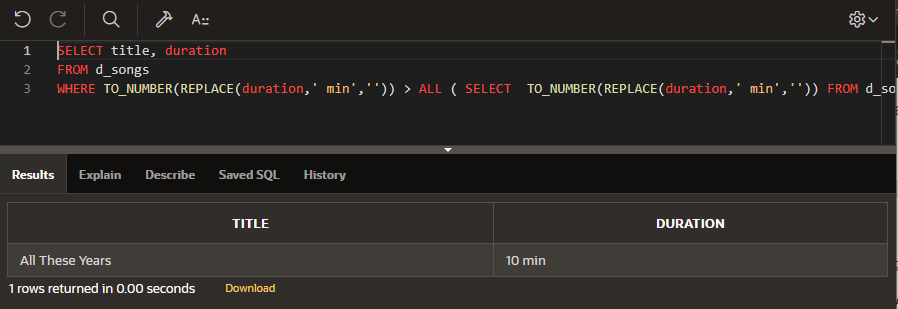
* 1. What song title has a duration longer than every type code 77 title?

WHERE duration \_\_\_\_\_\_\_\_\_(SELECT duration ...

SELECT title, duration

FROM d\_songs

WHERE TO\_NUMBER(REPLACE(duration,' min','')) > ALL ( SELECT TO\_NUMBER(REPLACE(duration,' min','')) FROM d\_songs where type\_code = 77);



Copyright © 2020, Oracle and/or its affiliates. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

1. If each WHERE clause is from the outer query, which of the following are true?

\_True\_a. WHERE size > ANY -- If the inner query returns sizes ranging from 8 to 12, the value 9 could be returned in the outer query.

\_False\_b. WHERE book\_number IN -- If the inner query returns books numbered 102, 105, 437, and 225 then 325 could be returned in the outer query.

\_False\_c. WHERE score <= ALL -- If the inner query returns the scores 89, 98, 65, and 72, then 82 could be returned in the outer query.

\_True\_d. WHERE color NOT IN -- If the inner query returns red, green, blue, black, and then the outer query could return white.

\_False\_e. WHERE game\_date = ANY -- If the inner query returns 05-Jun-1997, 10-Dec-2002, and 2-Jan-2004, then the outer query could return 10-Sep-2002.

1. The goal of the following query is to display the minimum salary for each department whose minimum salary is less than the lowest salary of the employees in department 50. However, the subquery does not execute because it has five errors. Find them, correct them, and run the query.

SELECT department\_id

FROM employees

WHERE MIN(salary)

HAVING MIN(salary) >

GROUP BY department\_id

SELECT MIN(salary)

WHERE department\_id < 50;

a)

ORA-00934: group function is not allowed here

Remove WHERE MIN(salary)

b)

ORA-00936: missing expression

This is from HAVING

Move the subquery in having and change the sign.

c)

ORA-00923: FROM keyword not found where expected

Put FROM employees in subquery.

d)

But I want minimum salary of department 50, change the where clause in subquery.

e)

But I want minimum salary instead of department no.

Change SELECT of outer query.

SELECT department\_id, MIN(salary)

FROM employees

GROUP BY department\_id

HAVING MIN(salary) < (SELECT MIN(salary) FROM employees WHERE department\_id = 50);

1. Which statements are true about the subquery below?

SELECT employee\_id, last\_name

FROM employees

WHERE salary =

(SELECT MIN(salary)

FROM employees

GROUP BY department\_id);

\_False\_ a. The inner query could be eliminated simply by changing the WHERE clause to WHERE MIN(salary).

\_True\_ b. The query wants the names of employees who make the same salary as the smallest salary in any department.

\_False\_ c. The query first selects the employee ID and last name, and then compares that to the salaries in every department.

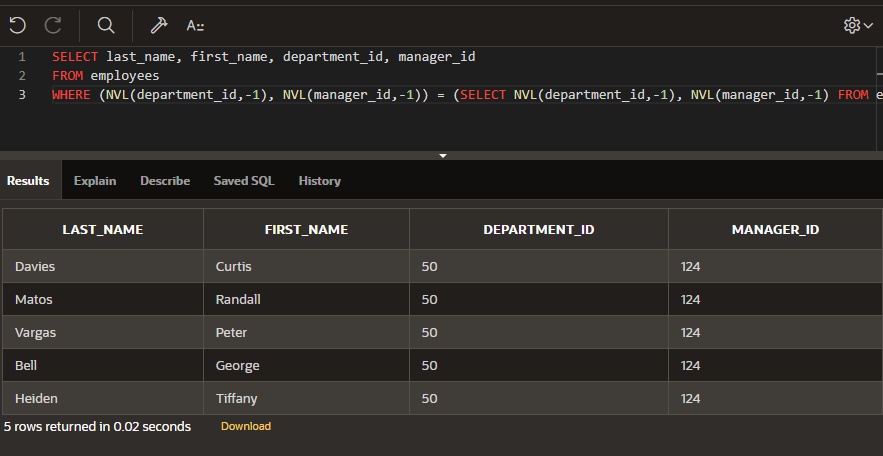
\_True\_ d. This query will not execute.

1. Write a pair-wise subquery listing the last\_name, first\_name, department\_id, and manager\_id for all employees that have the same department\_ id and manager\_id as employee 141. Exclude employee 141 from the result set.

SELECT last\_name, first\_name, department\_id, manager\_id

FROM employees

WHERE (NVL(department\_id,-1), NVL(manager\_id,-1)) = (SELECT NVL(department\_id,-1), NVL(manager\_id,-1) FROM employees WHERE employee\_id = 141) AND employee\_id != 141



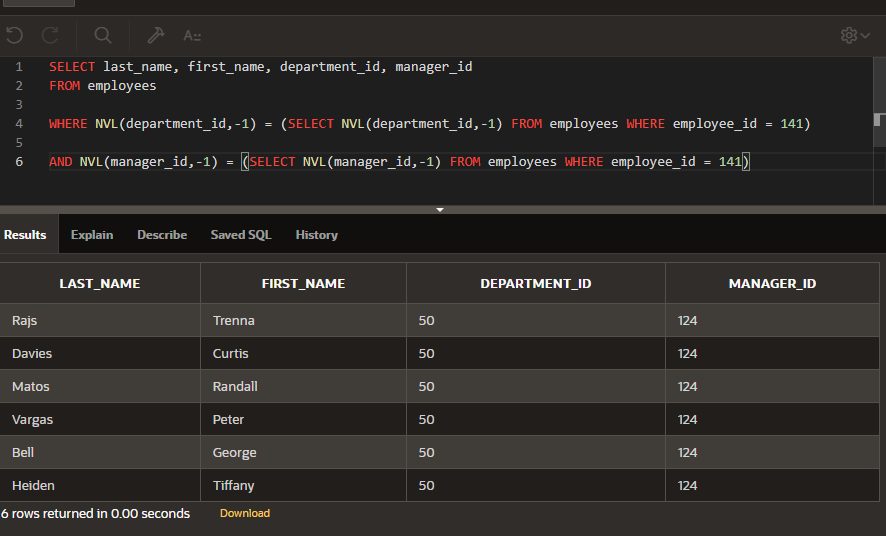
1. Write a non-pair-wise subquery listing the last\_name, first\_name, department\_id, and manager\_id for all employees that have the same department\_ id and manager\_id as employee 141.

SELECT last\_name, first\_name, department\_id, manager\_id

FROM employees

WHERE NVL(department\_id,-1) = (SELECT NVL(department\_id,-1) FROM employees WHERE employee\_id = 141)

AND NVL(manager\_id,-1) = (SELECT NVL(manager\_id,-1) FROM employees WHERE employee\_id = 141)

Copyright © 2020, Oracle and/or its affiliates. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. 

2