 



Homework

10-4: Correlated Subqueries Practice Activities

# Try It / Solve It

1. Explain the main difference between correlated and non-correlated subqueries?

Dacă o subinterogare necorelată este executată o singură dată, subinterogarea corelată va fi executată în mod repetat o dată pentru fiecare valoare produsă de SELECT-ul exterior.

1. Write a query that lists the highest earners for each department. Include the last\_name, department\_id, and the salary for each employee.

SELECT o.last\_name, o.department\_id, o.salary

FROM employees o

WHERE o.salary =

(SELECT MAX(i.salary)

FROM employees i

WHERE i.department\_id = o.department\_id)

1. Examine the following select statement and finish it so that it will return the last\_name, department\_id, and salary of employees who have at least one person reporting to them. So we are effectively looking for managers only. In the partially written SELECT statement, the WHERE clause will work as it is. It is simply testing for the existence of a row in the subquery.

SELECT (enter columns here)

FROM (enter table name here) outer WHERE 'x' IN (SELECT 'x'

FROM (enter table name here) inner

WHERE inner(enter column name here) = inner(enter column name here) Finish off the statement by sorting the rows on the department\_id column.

SELECT last\_name, department\_id, salary

FROM employees outer

WHERE employee\_id NOT IN

(SELECT DISTINCT inner.manager\_id

FROM employees inner

WHERE inner.manager\_id = outer.employee\_id)

ORDER BY department\_id

1. Using a WITH clause, write a SELECT statement to list the job\_title of those jobs whose maximum salary is more than half the maximum salary of the entire company. Name your subquery MAX\_CALC\_SAL. Name the columns in the result JOB\_TITLE and JOB\_TOTAL, and sort the result on JOB\_TOTAL in descending order.

Hint: Examine the jobs table. You will need to join JOBS and EMPLOYEES to display the job\_title.

WITH max\_calc\_sal AS

(SELECT jobs.job\_id, jobs.job\_title, MAX(NVL(employees.salary, 0)) AS job\_actual\_max

FROM employees RIGHT OUTER JOIN jobs

ON employees.job\_id = jobs.job\_id

GROUP BY jobs.job\_id, jobs.job\_title)

SELECT job\_title, job\_actual\_max AS job\_total

FROM max\_calc\_sal

WHERE job\_actual\_max > (SELECT MAX(job\_actual\_max)/2

FROM max\_calc\_sal)

ORDER BY job\_total DESC;