**Database Systems for Software Applications**

**Solutions to Section F**

|  |  |
| --- | --- |
| F1 | SELECT staff.ename, staff.empno, ' is managed by ', mngr.ename  FROM emp staff  INNER JOIN emp mngr  ON staff.mgr = mngr.empno  WHERE mngr.ename IN ('BLAKE','JONES');  -- NB: the third column (string literal) is not required, but helps to make the output clearer |
| F2 | SELECT mngr.ename, mngr.deptno, COUNT(\*)  FROM emp staff  INNER JOIN emp mngr  ON staff.mgr = mngr.empno  GROUP BY mngr.ename, mngr.deptno; |
| F3 | SELECT mngr.ename, mngr.deptno, COUNT(\*)  FROM emp staff  INNER JOIN emp mngr  ON staff.mgr = mngr.empno  GROUP BY mngr.ename, mngr.deptno  HAVING COUNT(\*) >2; |
|  |  |
| F6 | SELECT staff.ename, ' is managed by ', mngr.ename  FROM emp staff  INNER JOIN emp mngr  ON staff.mgr = mngr.empno;  -- this will return 13 rows, but there are 14 employees  -- KING is missing yet again  -- this time it’s because he has no manager, staff.mgr is NULL, so no match for the join. |
| F8 | SELECT staff.ename, staff.empno, ' is managed by ', mngr.ename, mngr.empno  FROM emp staff  LEFT OUTER JOIN emp mngr  ON staff.mgr = mngr.empno;  -- KING is included despite having no manager |
| F4 | SELECT ename, loc, emp.deptno  FROM emp INNER JOIN dept  ON emp.deptno = dept.deptno  WHERE sal > 1500;  -- this will return 6 rows, but there are 7 employees with sal > 1500  -- KING is missing even though his salary = 5000  -- KING is missing because emp.deptno is NULL, so no match for the join, and the row is discarded |
| F5 | SELECT ename, sal, loc  FROM emp INNER JOIN dept  ON emp.deptno = dept.deptno;  -- this will return 13 rows, but there are 14 employees  -- KING is still missing, same reason as above |
| F7 | SELECT ename, loc, emp.deptno  FROM emp LEFT OUTER JOIN dept  ON emp.deptno = dept.deptno  WHERE sal > 1500;  -- 7 rows instead of 6 in F4. KING is now included. His row in emp is not discarded despite the non-match |
| F9 | SELECT ename, ' is not in Research'  FROM emp LEFT OUTER JOIN dept  ON emp.deptno = dept.deptno  WHERE NVL(dname,'NONE') <> 'RESEARCH';  -- without the NVL on dname you will get only 8 rows, - no KING  -- because although the row for KING in emp is retained by the outer join, . . .  -- the WHERE clause will not match NULL as <> ‘RESEARCH’  -- alternatively, this query will also work:  SELECT ename, ' is not in Research'  FROM emp LEFT OUTER JOIN dept  ON emp.deptno = dept.deptno  WHERE dname <> 'RESEARCH' or dname IS NULL; |
| F10 | SELECT dname  FROM emp RIGHT OUTER JOIN dept  ON emp.deptno = dept.deptno  WHERE ename IS NULL; |
| F11 | SELECT dname, COUNT(ename)    FROM emp RIGHT OUTER JOIN dept  ON emp.deptno = dept.deptno  GROUP BY dname  HAVING COUNT(ename) <6;  -- we must use an outer join to include depts without any employees at all  -- but the count must be on ename, to avoid counting null values as employees.  -- using count(\*) here would give Operations a count of 1 (but there are none) |