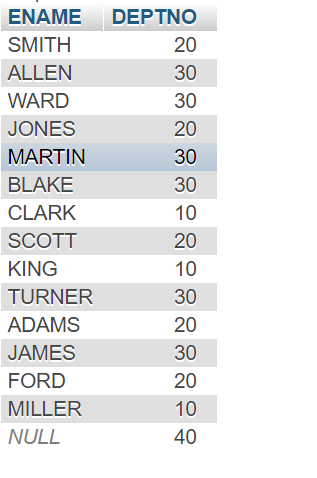
 **Tutorial -SQL Exercises 3**

* **(INNER) JOIN**: Returns records that have matching values in both tables
* **LEFT (OUTER) JOIN**: Returns all records from the left table, and the matched records from the right table
* **RIGHT (OUTER) JOIN**: Returns all records from the right table, and the matched records from the left table
* **FULL (OUTER) JOIN**: Returns all records when there is a match in either left or right table

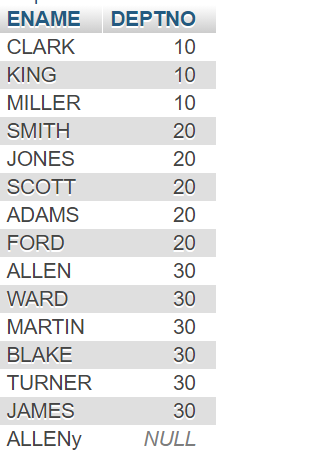
**Right Join**

**SELECT emp.ENAME, dept.DEPTNO**

**FROM `emp`**

**RIGHT JOIN dept**

**ON emp.DEPTNO = dept.DEPTNO**



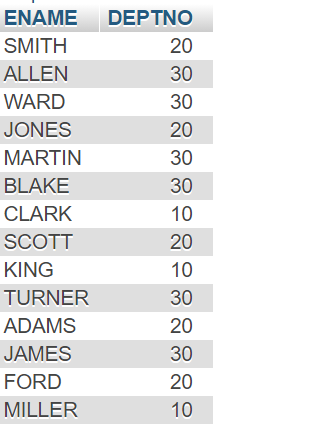
**Left Join**

**SELECT emp.ENAME, dept.DEPTNO**

**FROM `emp`**

**LEFT JOIN dept**

**ON emp.DEPTNO = dept.DEPTNO**

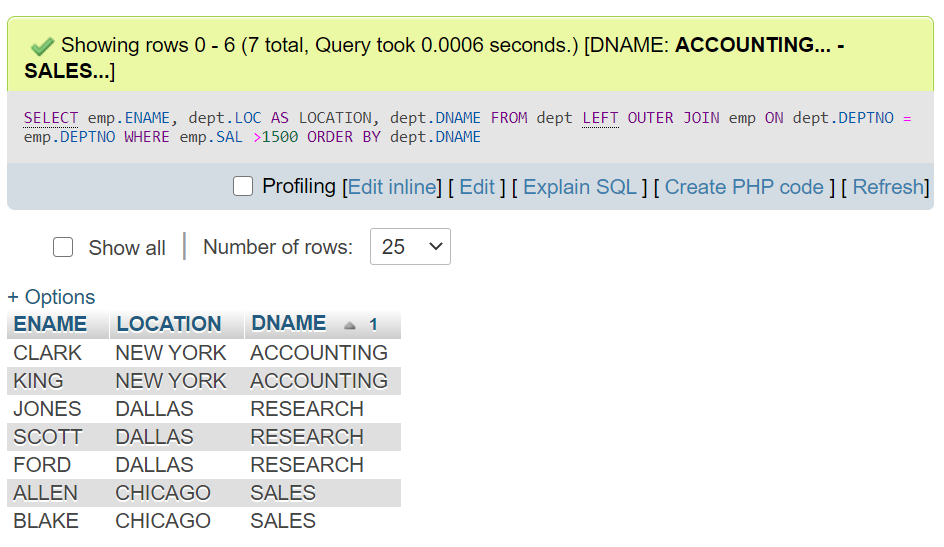
**Inner Join**

**SELECT emp.ENAME, dept.DEPTNO**

**FROM `emp`**

**INNER JOIN dept**

**ON emp.DEPTNO = dept.DEPTNO**

1. **Display employee name, location and department or those whose salary is greater than 1500**

**You can use LEFT OUTER JOIN or LEFT JOIN**

**SELECT emp.ENAME, dept.LOC AS LOCATION, dept.DNAME**

**FROM dept**

**LEFT OUTER JOIN emp**

**ON dept.DEPTNO = emp.DEPTNO**

**WHERE emp.SAL >1500**

**ORDER BY dept.DNAME**

**2nd Method**

**SELECT emp.ENAME, dept.LOC AS LOCATION, dept.DNAME**

**FROM emp**

**LEFT OUTER JOIN dept**

**ON emp.DEPTNO = dept.DEPTNO**

**WHERE emp.SAL >1500**

**ORDER BY dept.DNAME**

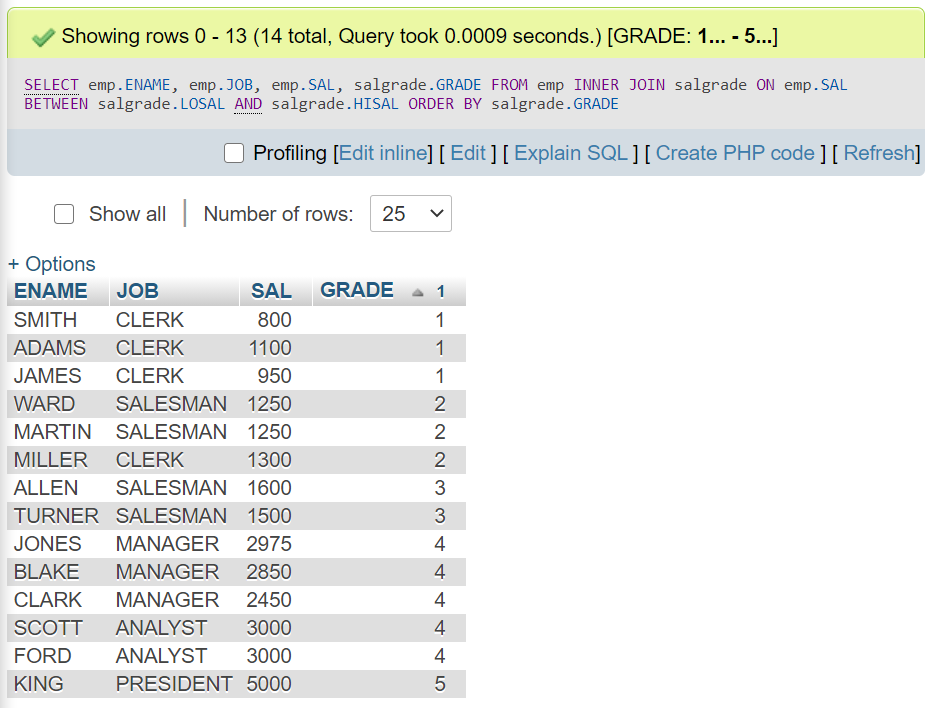
**3rd Method**

**SELECT emp.ENAME, dept.LOC AS "LOCATION" , dept.DNAME**

**FROM dept , emp**

**WHERE dept.DEPTNO = emp.DEPTNO AND emp.SAL >1500**

**ORDER BY dept.DNAME**

1. **Produce a list which will show the grade that each employee is in**

**1st Method**

**SELECT emp.ENAME, emp.JOB, emp.SAL, salgrade.GRADE**

**FROM emp**

**INNER JOIN salgrade**

**ON emp.SAL**

**BETWEEN salgrade.LOSAL AND salgrade.HISAL**

**ORDER BY salgrade.GRADE;**

**2nd Method**

**SELECT ENAME, JOB, SAL,salgrade.GRADE**

**FROM `emp`**

**LEFT OUTER JOIN salgrade**

**ON EMP.SAL**

**BETWEEN salgrade.LOSAL AND salgrade.HISAL**

**ORDER BY salgrade.GRADE**

**3rd Method**

**SELECT emp.ENAME, emp.JOB , emp.SAL, salgrade.GRADE**

**FROM salgrade, emp**

**WHERE emp.SAL BETWEEN salgrade.LOSAL AND salgrade.HISAL**

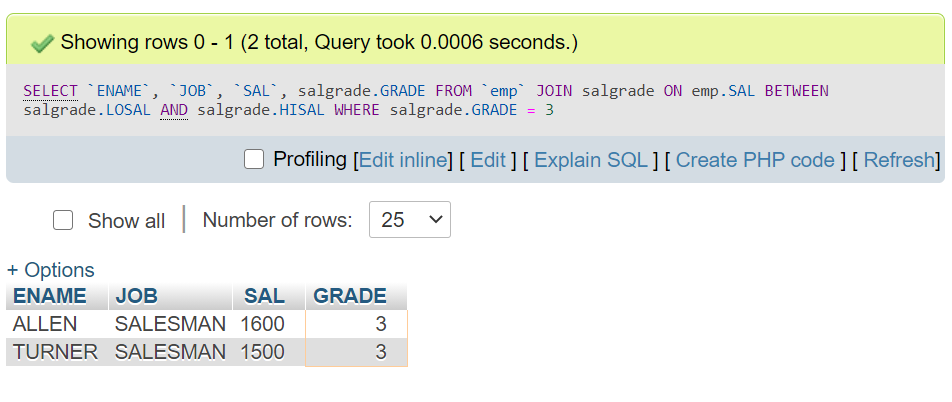
**ORDER BY salgrade.GRADE**

1. **List employee name, job, salary and grade for those in grade 3**

**SELECT `ENAME`, `JOB`, `SAL`, salgrade.GRADE FROM `emp`**

**JOIN salgrade**

**ON emp.SAL BETWEEN salgrade.LOSAL AND salgrade.HISAL**

 **WHERE salgrade.GRADE = 3;**

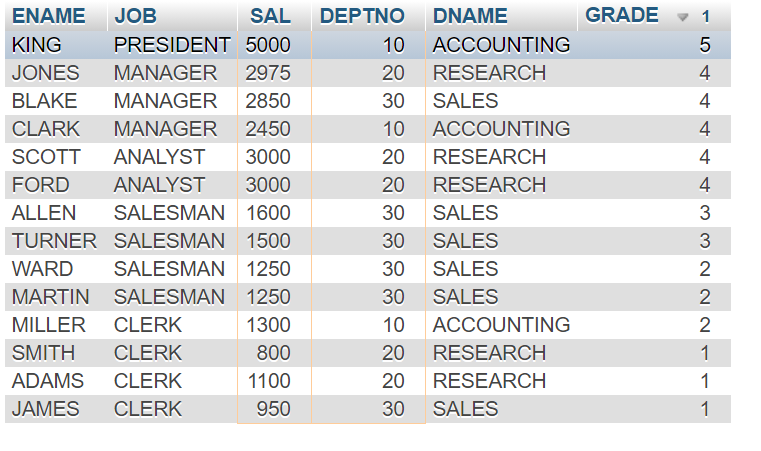
1. **As 2 above, but also give their department name, in descending order of their salary grade.**

**SELECT emp.ENAME, emp.JOB, emp.SAL, emp.DEPTNO,dept.DNAME, salgrade.GRADE**

**FROM emp, salgrade, dept**

**WHERE (emp.SAL BETWEEN salgrade.LOSAL AND salgrade.HISAL) AND EMP.DEPTNO = dept.DEPTNO**

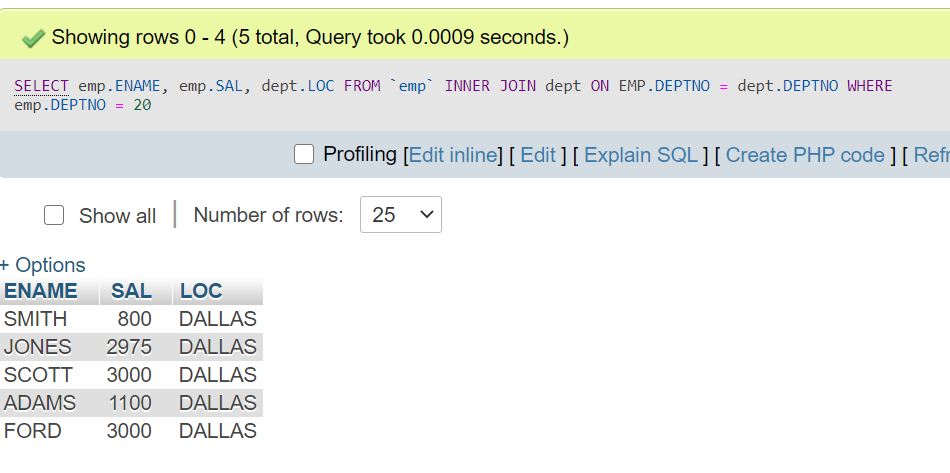
**ORDER BY salgrade.GRADE DESC**



1. **Find the name and salary of employees in Dallas**

**SELECT emp.ENAME, emp.SAL, dept.LOC**

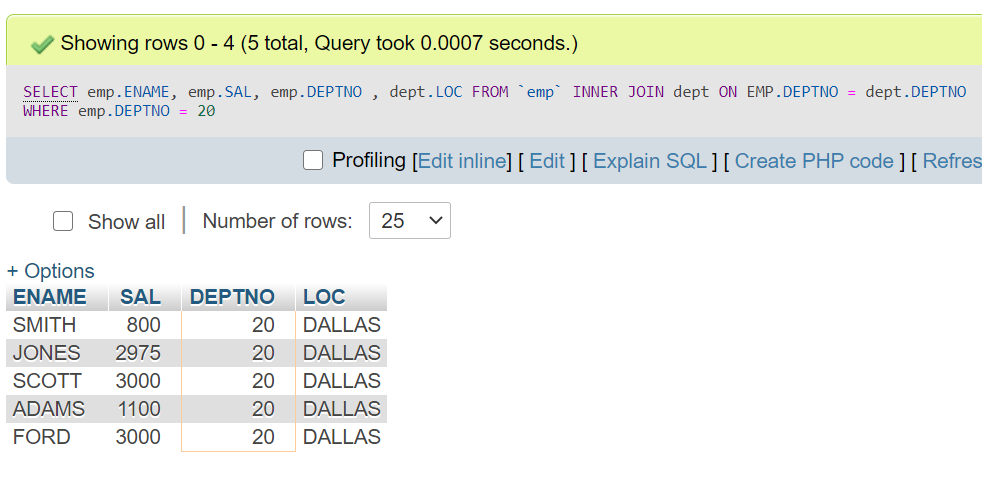
**FROM `emp` INNER JOIN dept ON EMP.DEPTNO = dept.DEPTNO**

 **WHERE emp.DEPTNO = 20**

**SELECT emp.ENAME, emp.SAL, emp.DEPTNO , dept.LOC**

**FROM `emp` INNER JOIN dept ON EMP.DEPTNO = dept.DEPTNO**

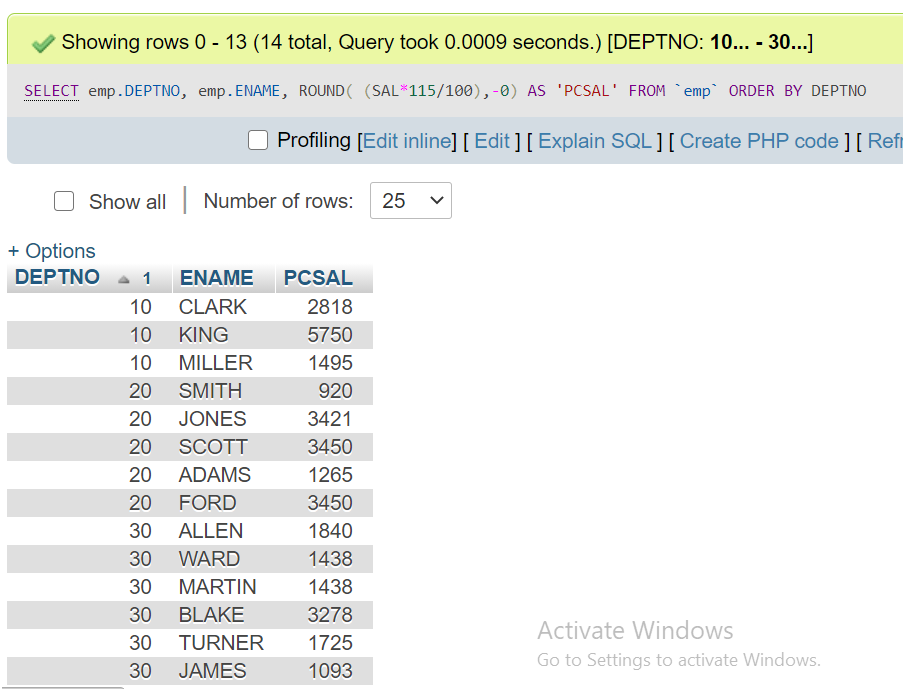
**WHERE emp.DEPTNO = 20**



1. **List in order of the department number, the department number, employee name and salary increased by 5% and expressed as a whole of pounds.**

**SELECT emp.DEPTNO, emp.ENAME, ROUND( (SAL\*115/100),-0) AS 'PCSAL'**

**FROM `emp`**

 **ORDER BY DEPTNO**

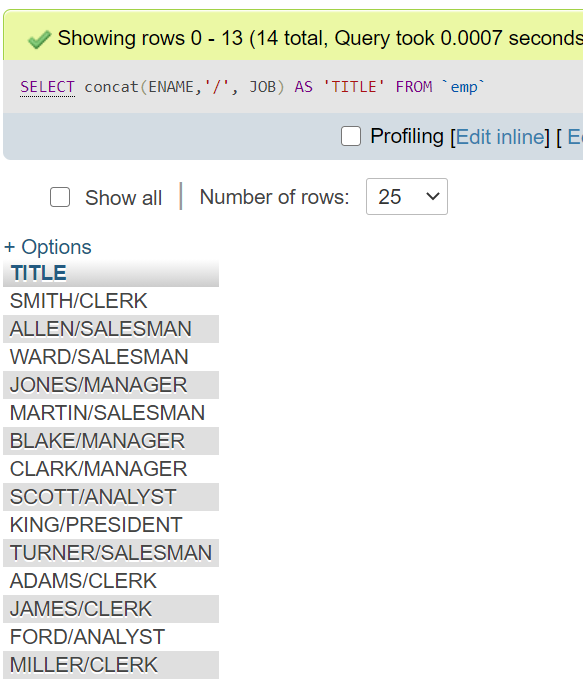
1. **Print the employee name and the job as one field with a / between the two fields. Also try getting the ‘/’s lined up.**

**1. SELECT concat(ENAME,'/', JOB) AS 'TITLE'**

**FROM `emp`**

**2. SELECT concat\_ws ( '/ ', ENAME, JOB) AS 'TITLE'**

**FROM `emp`**

****

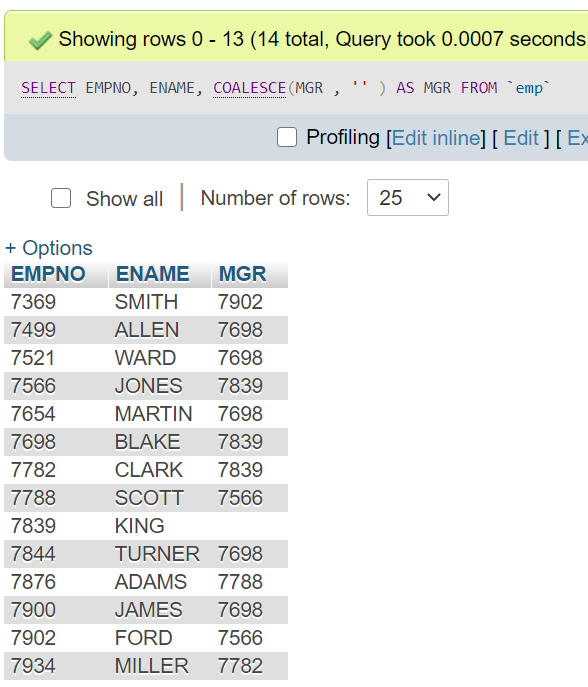
1. **List all employees and their manager’s number.**

**SELECT EMPNO, ENAME, COALESCE(MGR , ' ' ) AS MGR**

**FROM `emp`**

**Here, the null value of KING is not printed as *NULL*….it is printed a blank.**

**COALESCE(column,’ ’ ) is the function which is used for that**



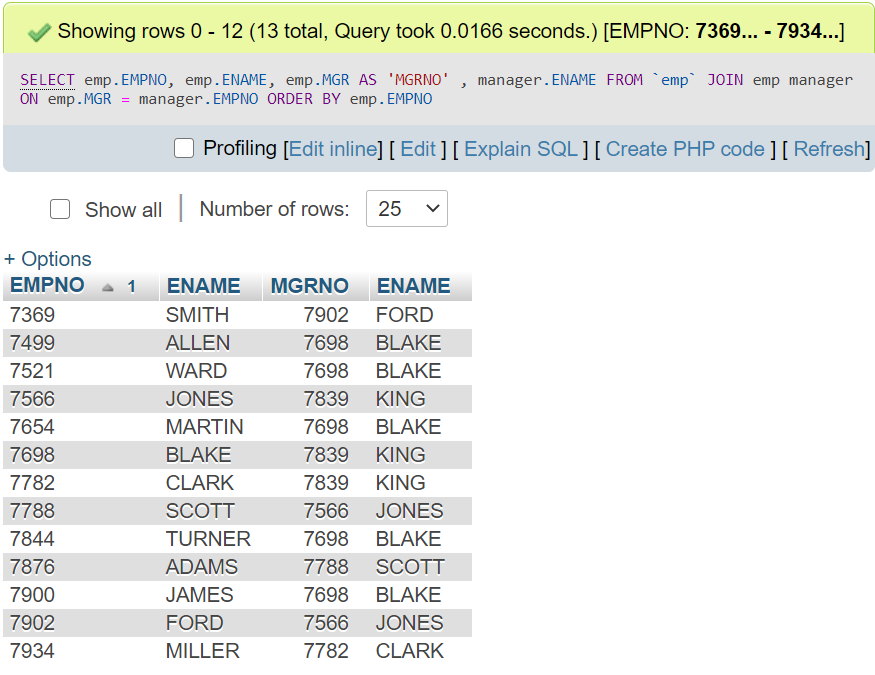
1. **As 8 above, but list their manager’s name as well as his number**

**SELECT emp.EMPNO, emp.ENAME, emp.MGR AS 'MGRNO' , manager.ENAME**

**FROM `emp`**

**JOIN emp manager**

**ON emp.MGR = manager.EMPNO**

 **ORDER BY emp.EMPNO**

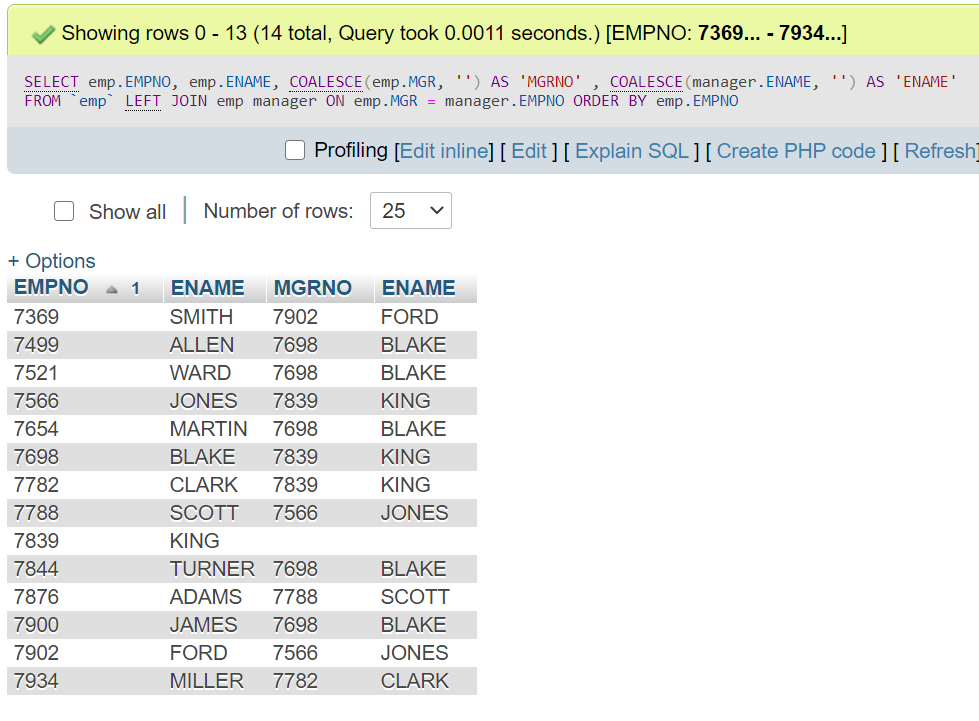
1. **Same as 9. Above, but include those employees that don’t have a manager.**

**SELECT emp.EMPNO, emp.ENAME, COALESCE(emp.MGR, '') AS 'MGRNO' , COALESCE(manager.ENAME, '') AS 'ENAME'**

**FROM `emp`**

**LEFT JOIN emp manager**

**ON emp.MGR = manager.EMPNO**

**ORDER BY emp.EMPNO**

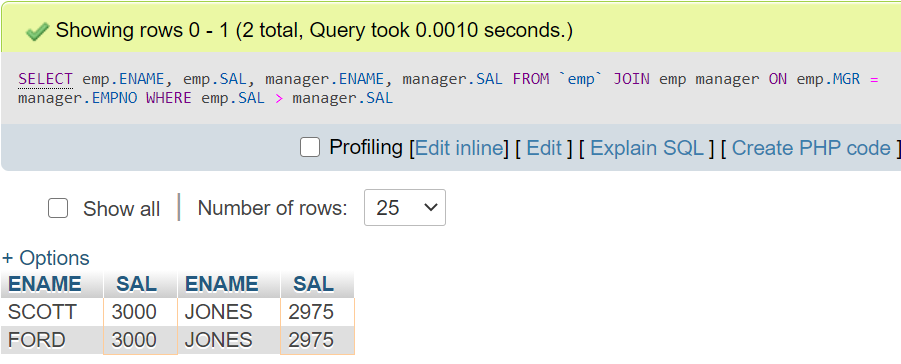
1. **List the employee’s name and salary and manager’s name and salary for all employees who earn more than their manager.**

**SELECT emp.ENAME, emp.SAL, manager.ENAME, manager.SAL**

**FROM `emp`**

**JOIN emp manager**

**ON emp.MGR = manager.EMPNO**

 **WHERE emp.SAL > manager.SAL**