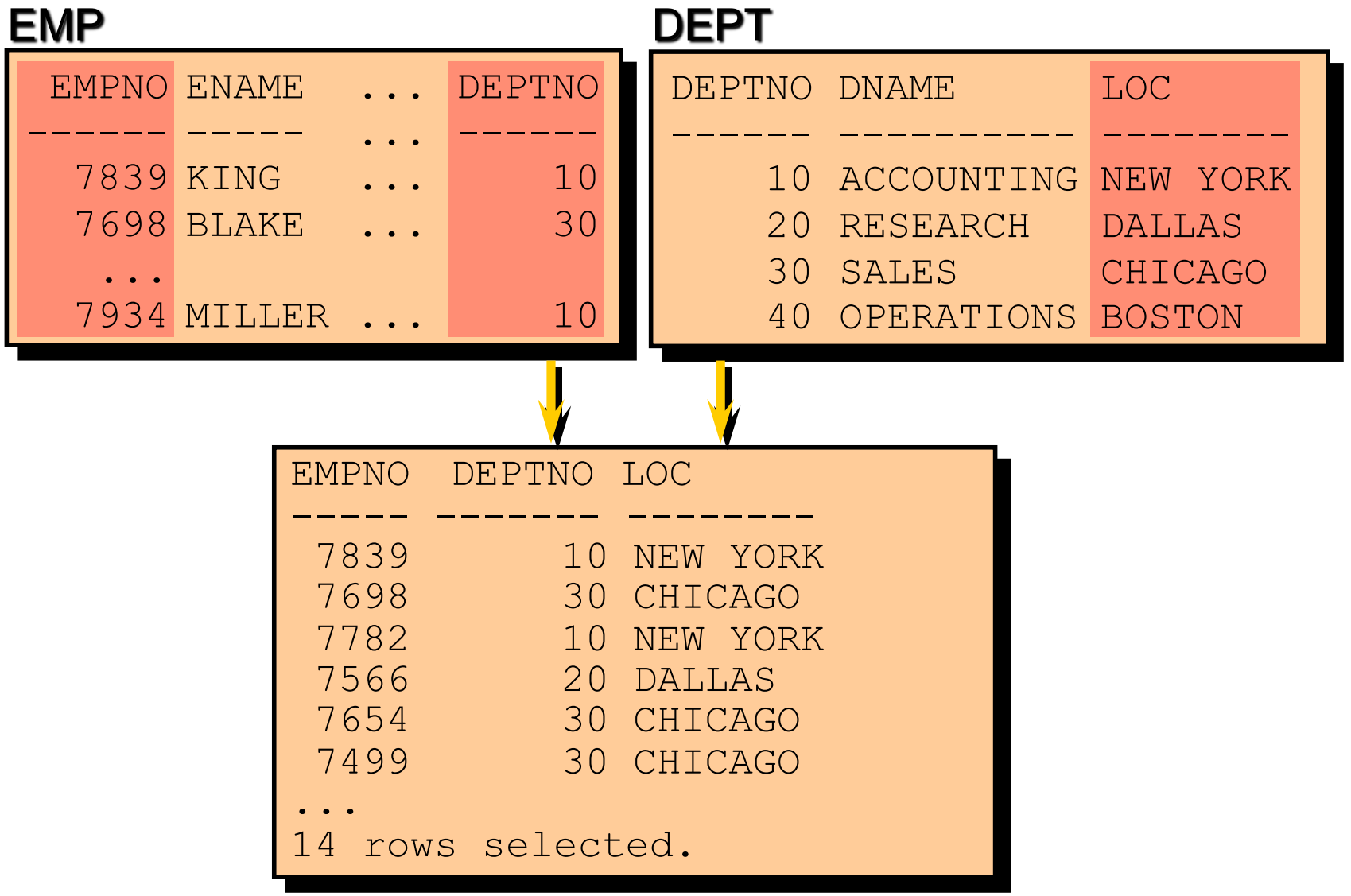
SQL

Obteniendo información de más de una tabla

Lecciones de SQL

Obteniendo información de más de una tabla



JOIN

```
SELECT   table1.column, table2.column
FROM     table1, table2
WHERE    table1.column1 = table2.column2;
```

Indique las columnas involucradas después del WHERE.

Cuándo el nombre de las columnas es el mismo en dos tablas diferentes, la columna debe calificarse.

Producto Cartesiano

- Un producto cartesiano se forma cuando:
  Se omite la clausula WHERE.

- La condición de unión no es válida.

- Todas las filas de la primera tabla se unen a todas las filas de la segunda tabla.

Producto Cartesiano

EMP (14 renglones)

```
 EMPNO ENAME   ...  DEPTNO
------ -----   ...  ------
  7839 KING    ...      10
  7698 BLAKE   ...      30
   ...
  7934 MILLER  ...      10
```

DEPT (4 renglones)

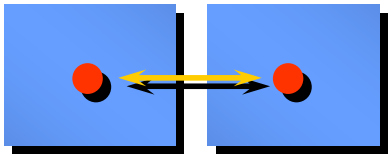
```
DEPTNO DNAME          LOC
------ -------------- --------
    10 ACCOUNTING     NEW YORK
    20 RESEARCH       DALLAS
    30 SALES          CHICAGO
    40 OPERATIONS     BOSTON
```

Producto:
14*4=56
renglones”

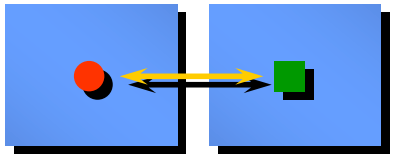
```
ENAME      DNAME
------     ----------
KING       ACCOUNTING
BLAKE      ACCOUNTING
...
KING       RESEARCH
BLAKE      RESEARCH
...
56 rows selected.
```

Tipos de Joins

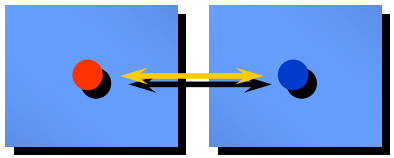
Equijoin



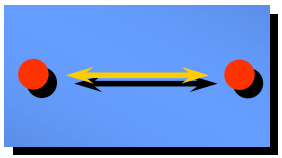
Non-equijoin



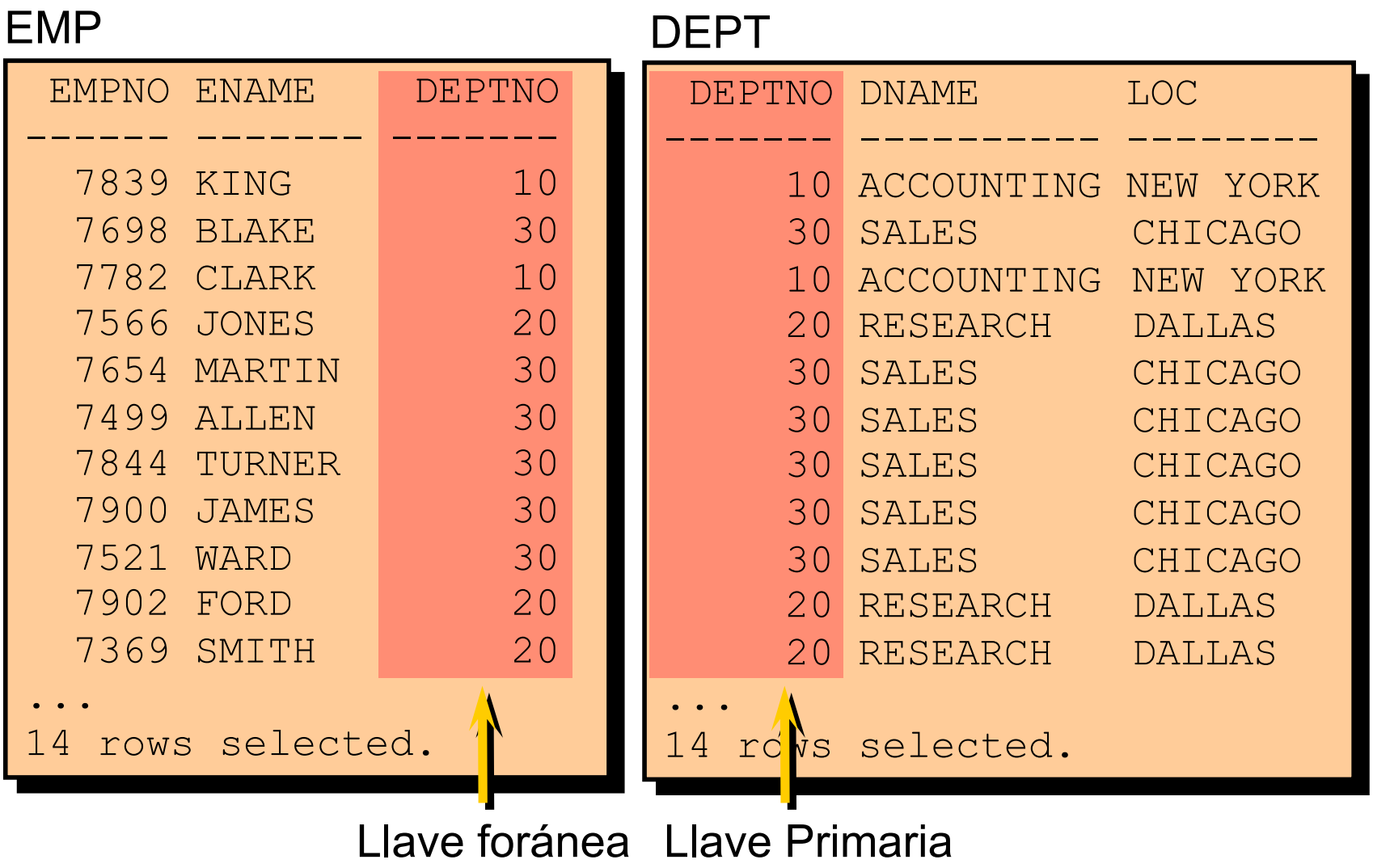
Outer join



Self join



EQUIJOIN


EMP
EMPNO ENAME DEPTNO
7839 KING 10
7698 BLAKE 30
7782 CLARK 10
7566 JONES 20
7654 MARTIN 30
7499 ALLEN 30
7844 TURNER 30
7900 JAMES 30
7521 WARD 30
7902 FORD 20
7369 SMITH 20
...
14 rows selected.
DEPT
DEPTNO DNAME LOC
10 ACCOUNTING NEW YORK
30 SALES CHICAGO
10 ACCOUNTING NEW YORK
20 RESEARCH DALLAS
30 SALES CHICAGO
30 SALES CHICAGO
30 SALES CHICAGO
30 SALES CHICAGO
30 SALES CHICAGO
20 RESEARCH DALLAS
20 RESEARCH DALLAS
...
14 rows selected.
Llave foránea
Llave Primaria

Operador de igualdad

```
SQL> SELECT   emp.empno, emp.ename, emp.deptno,
  2           dept.deptno, dept.loc
  3   FROM    emp, dept
  4   WHERE   emp.deptno=dept.deptno;
```

```
EMPNO ENAME   DEPTNO DEPTNO LOC
----- ------ ------ ------ ----------
 7839 KING       10     10 NEW YORK
 7698 BLAKE      30     30 CHICAGO
 7782 CLARK      10     10 NEW YORK
 7566 JONES      20     20 DALLAS
...
14 rows selected.
```

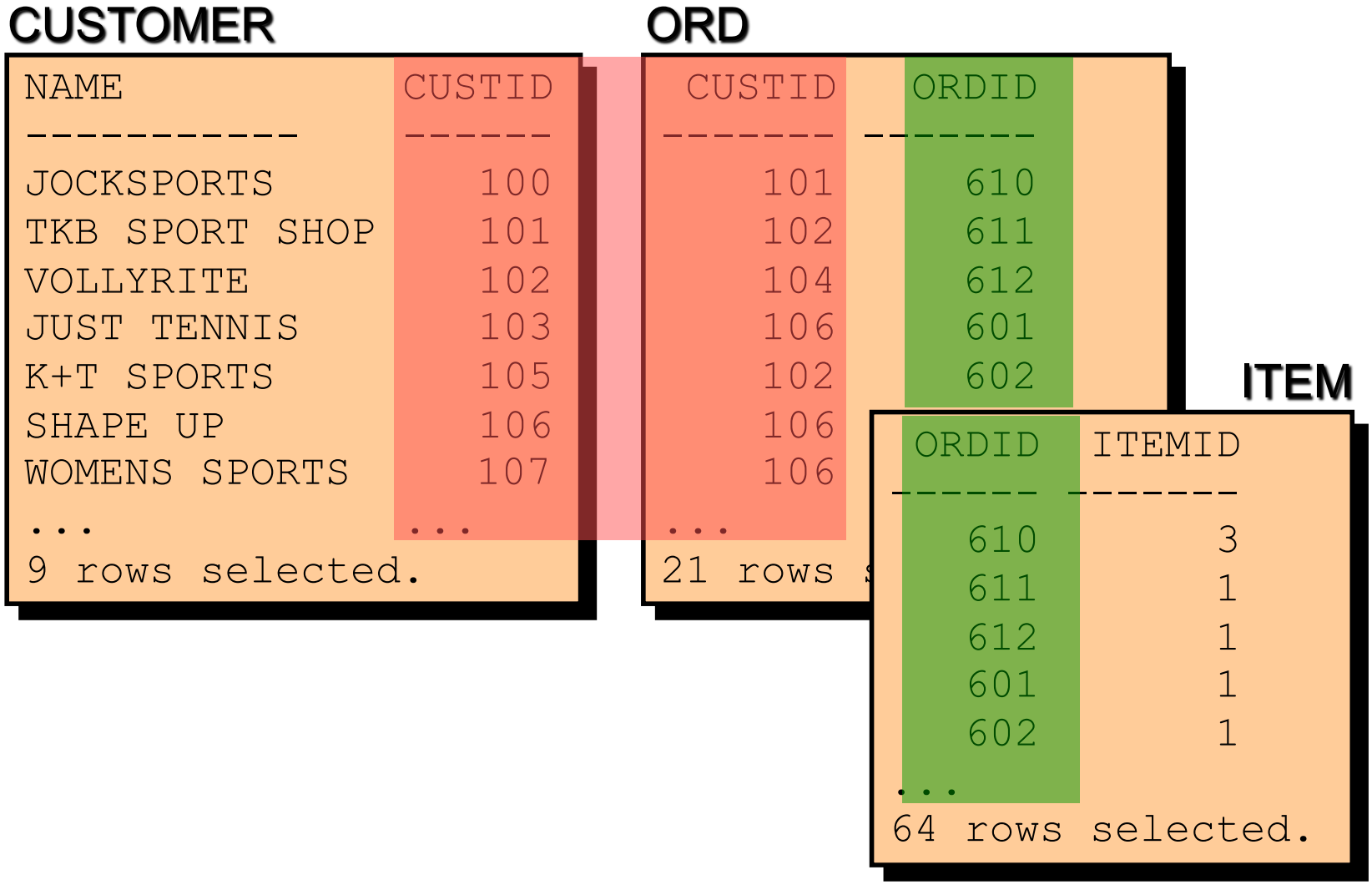
Usando Alias

Utilice los alias para simplificar la consulta.

```
SQL> SELECT emp.empno, emp.ename, emp.deptno,
  2         dept.deptno, dept.loc
  3  FROM   emp, dept
  4  WHERE  emp.deptno=dept.deptno;
```

```
SQL> SELECT e.empno, e.ename, e.deptno,
  2         d.deptno, d.loc
  3  FROM   emp e, dept d
  4  WHERE  e.deptno=d.deptno;
```

JOINS QUE INVOLUCRAN MAS DE DOS TABLAS



NON-EQUIJOINS

EMP

EMPNO	ENAME	SAL
7839	KING	5000
7698	BLAKE	2850
7782	CLARK	2450
7566	JONES	2975
7654	MARTIN	1250
7499	ALLEN	1600
7844	TURNER	1500
7900	JAMES	950

...
14 rows selected.

SALGRADE

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

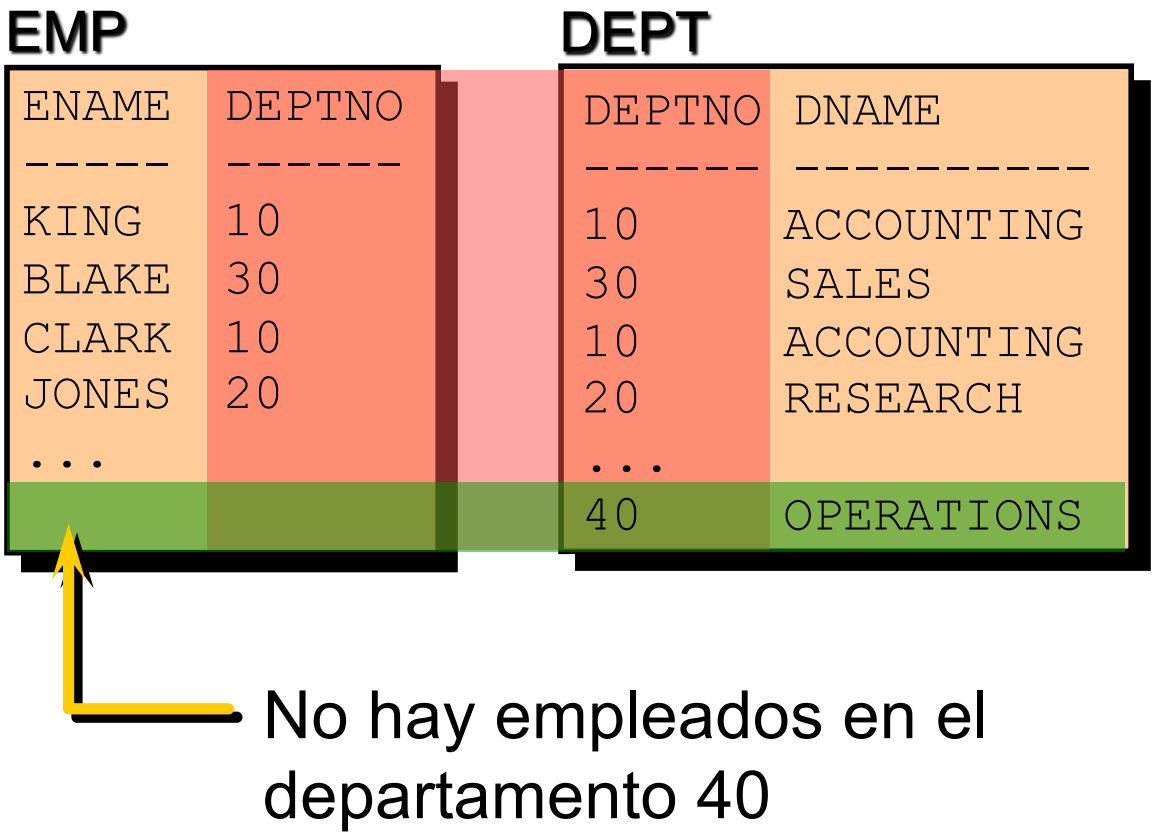
"Nivel del salario de un empleado"

NON-EQUIJOINS

```
SQL>   SELECT   e.ename, e.sal, s.grade
   2   FROM     emp e, salgrade s
   3   WHERE    e.sal
   4   BETWEEN  s.losal AND s.hisal;
```

```
ENAME             SAL      GRADE
---------- ---------- ----------
JAMES             950          1
SMITH             800          1
ADAMS            1100          1
...
14 rows selected.
```

OUTER JOINS


EMP
ENAME DEPTNO
KING 10
BLAKE 30
CLARK 10
JONES 20
...
DEPT
DEPTNO DNAME
10 ACCOUNTING
30 SALES
10 ACCOUNTING
20 RESEARCH
...
40 OPERATIONS
No hay empleados en el departamento 40

OUTER JOINS

- Muestra las filas que no cumplen con la condición de unión (complemento).
- El signo del operador Outer join es (+).

```
SELECT table.column, table.column
FROM   table1, table2
WHERE  table1.column(+) = table2.column;
```

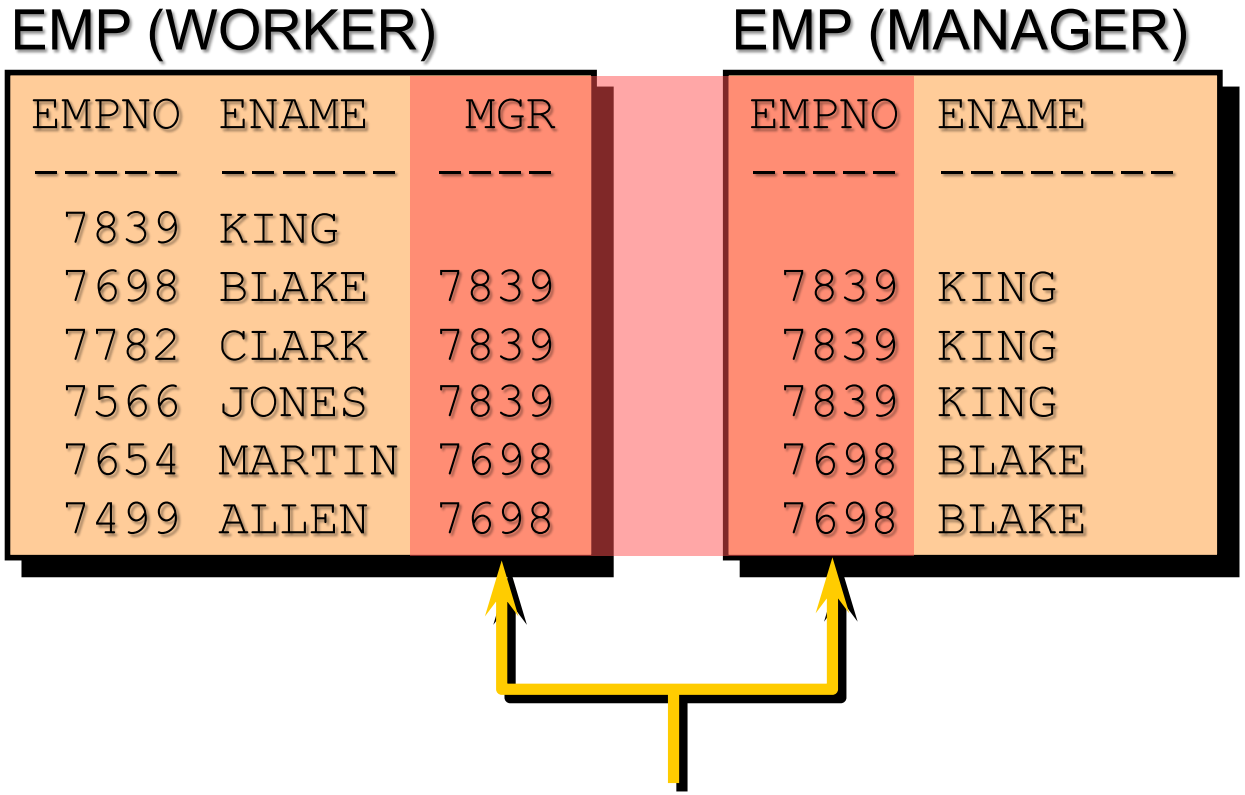
```
SELECT table.column, table.column
FROM   table1, table2
WHERE  table1.column = table2.column(+);
```

Utilizando Outer Joins

```
SQL> SELECT   e.ename, d.deptno, d.dname
  2  FROM     emp e, dept d
  3  WHERE    e.deptno(+) = d.deptno
  4  ORDER BY e.deptno;
```

```
ENAME          DEPTNO DNAME
---------- ---------- --------------
KING               10 ACCOUNTING
CLARK              10 ACCOUNTING
...
                   40 OPERATIONS
15 rows selected.
```

SELF JOINS



"MGR en la tabla WORKER es el EMPNO en la tabla MANAGER"

Es un JOIN sobre la misma tabla

```
SQL> SELECT worker.ename||' works for '||manager.ename
  2  FROM   emp worker, emp manager
  3  WHERE  worker.mgr = manager.empno;
```

```
WORKER.ENAME||'WORKSFOR'||MANAG
------------------------------
BLAKE works for KING
CLARK works for KING
JONES works for KING
MARTIN works for BLAKE
...
13 rows selected.
```