

```
--drop database dbsegundo;  
create database dbsegundo;  
use dbsegundo;
```

```
/*=====*/  
/* Table: DEPARTAMENTOS */  
/*=====*/
```

```
create table DEPARTAMENTOS (  
    DEPTNO          int          not null,  
    DNAME           varchar(15)   not null,  
    LOC             varchar(10)   not null,  
    constraint PK_DEPARTAMENTOS primary key nonclustered (DEPTNO)  
);
```

```
/*=====*/  
/* Table: EMPLEADOS */  
/*=====*/
```

```
create table EMPLEADOS (  
    EMPNO           int          not null,  
    DEPTNO          int          null,  
    ENAME           varchar(10)  not null,  
    JOB             varchar(20)  not null,  
    MGR            int          null,  
    HIREDATE        datetime     not null,  
    SAL             int          not null,  
    COMM            int          null,  
    constraint PK_EMPLEADOS primary key nonclustered (EMPNO)  
);
```

```
/*=====*/  
/* Index: TIENE_FK */  
/*=====*/
```

```
create index TIENE_FK on EMPLEADOS (  
    DEPTNO ASC  
);
```

```
create table DEPARTAMENTOS(  
    DEPTNO          int          not null,  
    DNAME           varchar(20)  not null,  
    LOC             varchar(20)  not null,  
    constraint PK_DEPARTAMENTOS primary key nonclustered (DEPTNO));
```

```
alter table EMPLEADOS  
    add constraint FK_EMPLEADO_TIENE_DEPARTAM foreign key (DEPTNO)  
        references DEPARTAMENTOS (DEPTNO);
```

```
insert into departamentos values
(10, 'ACCOUNTING', 'NEW YORK'),
(20, 'RESEARCH', 'DALLAS'),
(30, 'SALES', 'CHICAGO'),
(40, 'OPERATIONS', 'BOSTON');
```

```
insert into empleados (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)
values
```

```
(7369, 'SMITH', 'CLERK', 7902, '1980/12/17', 800, null, 20),
(7499, 'ALLEN', 'SALESMAN', 7698, '1981/02/20', 1600, 300, 30),
(7521, 'WARD', 'SALESMAN', 7698, '1981/02/22', 1250, 500, 30),
(7566, 'JONES', 'MANAGER', 7839, '1981/04/02', 2975, null, 20),
(7654, 'MARTIN', 'SALESMAN', 7698, '1981/09/28', 1250, 1400, 30),
(7698, 'BLAKE', 'MANAGER', 7839, '1981/05/01', 2850, null, 30),
(7782, 'CLARK', 'MANAGER', 7839, '1981/06/09', 2450, null, 10),
(7788, 'SCOTT', 'ANALYST', 7566, '1982/12/09', 3000, null, 20),
(7839, 'KING', 'PRESIDENT', null, '1981/11/17', 5000, null, 10),
(7844, 'TURNER', 'SALESMAN', 7698, '1981/09/08', 1500, 0, 30),
(7876, 'ADAMS', 'CLERK', 7788, '1983/01/12', 1100, null, 20),
(7900, 'JAMES', 'CLERK', 7698, '1981/12/03', 950, null, 30),
(7902, 'FORD', 'ANALYST', 7566, '1981/12/03', 3000, null, 20),
(7934, 'MILLER', 'CLERK', 7782, '1982/01/23', 1300, null, 10);
```

```
ALTER TABLE `m5`.`empleados`
CHANGE COLUMN `HIREDATE` `HIREDATE` DATE NOT NULL ;
```

```
ALTER TABLE `m5`.`empleados`
CHANGE COLUMN `DEPTNO` `DEPTNO` INT NULL DEFAULT NULL AFTER `COMM`;
```

**/\* 1. Obtener todos los datos de todos los empleados.**

```
SELECT empno, ename, job, mgr, DATE_FORMAT(hiredate, '%d-%m-%y') AS hiredate, sal,
comm, deptno FROM empleados;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17/12/80	800	20	
7499	ALLEN	SALESMAN	7698	20/02/81	1600	300	30
7521	WARD	SALESMAN	7698	22/02/81	1250	500	30
7566	JONES	MANAGER	7839	02/04/81	2975	20	
7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400	30
7698	BLAKE	MANAGER	7839	01/05/81	2850	30	

7782 CLARK	MANAGER	7839 09/06/81	2450	10	
7788 SCOTT	ANALYST	7566 09/12/82	3000	20	
7839 KING	PRESIDENT	17/11/81	5000	10	
7844 TURNER	SALESMAN	7698 08/09/81	1500	0	30
7876 ADAMS	CLERK	7788 12/01/83	1100	20	
7900 JAMES	CLERK	7698 03/12/81	950	30	
7902 FORD	ANALYST	7566 03/12/81	3000	20	
7934 MILLER	CLERK	7782 23/01/82	1300	10	

14 filas seleccionadas.\*/

**/\* 2. Obtener todos los datos de todos los departamentos.**

`SELECT * FROM departamentos;`

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

4 filas seleccionadas.\*/

**/\* 3. Obtener todos los datos de los administrativos (su trabajo es, en inglés, 'CLERK').**

`SELECT empno, ename, job, mgr, DATE_FORMAT(hiredate, '%d-%m-%y') AS hiredate, sal, comm, deptno FROM empleados WHERE job = 'clerk';`

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17/12/80	800	20	
7876	ADAMS	CLERK	7788	12/01/83	1100	20	
7900	JAMES	CLERK	7698	03/12/81	950	30	
7934	MILLER	CLERK	7782	23/01/82	1300	10	

4 filas seleccionadas.\*/

**/\* 4. Idem, pero ordenado por el nombre.**

```
SELECT empno, ename, job, mgr, DATE_FORMAT(hiredate, '%d-%m-%y') AS hiredate, sal,
comm, deptno FROM empleados WHERE job = 'clerk' ORDER BY ename;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7876	ADAMS	CLERK	7788	12/01/83	1100	20	
7900	JAMES	CLERK	7698	03/12/81	950	30	
7934	MILLER	CLERK	7782	23/01/82	1300	10	
7369	SMITH	CLERK	7902	17/12/80	800	20	

4 filas seleccionadas.\*/

**/\* 5. Obten el mismo resultado de la pregunta anterior, pero modificando la sentencia SQL.**

```
SELECT empno, ename, job, mgr, DATE_FORMAT(hiredate, '%d-%m-%y') AS hiredate, sal,
comm, deptno FROM empleados WHERE job = 'clerk' ORDER BY ename ASC;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7876	ADAMS	CLERK	7788	12/01/83	1100	20	
7900	JAMES	CLERK	7698	03/12/81	950	30	
7934	MILLER	CLERK	7782	23/01/82	1300	10	
7369	SMITH	CLERK	7902	17/12/80	800	20	

4 filas seleccionadas.\*/

**/\* 6. Obtén el número de empleado (código), nombre y salario de los empleados.**

```
SELECT empno, ename, sal FROM empleados;
```

EMPNO	ENAME	SAL
7369	SMITH	800
7499	ALLEN	1600
7521	WARD	1250
7566	JONES	2975
7654	MARTIN	1250
7698	BLAKE	2850
7782	CLARK	2450

7788 SCOTT	3000
7839 KING	5000
7844 TURNER	1500
7876 ADAMS	1100
7900 JAMES	950
7902 FORD	3000
7934 MILLER	1300

14 filas seleccionadas.\*/

**/\* 7. Lista los nombres de todos los departamentos.**

`SELECT dname FROM departamentos;`

```
DNAME
-----
ACCOUNTING
RESEARCH
SALES
OPERATIONS
```

4 filas seleccionadas.\*/

**/\* 8. Idem, pero ordenándolos por nombre.**

`SELECT dname FROM departamentos ORDER BY dname;`

```
DNAME
-----
ACCOUNTING
OPERATIONS
RESEARCH
SALES
```

4 filas seleccionadas.\*/

**/\* 9. Idem, pero ordenandolo por la ciudad (no se debe seleccionar la ciudad en el resultado).**

`SELECT dname FROM departamentos ORDER BY loc;`

DNAME

-----

OPERATIONS

SALES

RESEARCH

ACCOUNTING

4 filas seleccionadas.\*/

**/\* 10. Idem, pero el resultado debe mostrarse ordenado por la ciudad en orden inverso.**

SELECT dname FROM departamentos ORDER BY loc DESC;

DNAME

-----

ACCOUNTING

RESEARCH

SALES

OPERATIONS

4 filas seleccionadas.\*/

**/\* 11. Obtener el nombre y empleo de todos los empleados, ordenado por salario.**

SELECT ename, job FROM empleados ORDER BY sal;

ENAME      JOB

-----

SMITH      CLERK

JAMES      CLERK

ADAMS      CLERK

WARD      SALESMAN

MARTIN      SALESMAN

MILLER      CLERK

TURNER      SALESMAN

ALLEN      SALESMAN

CLARK      MANAGER

BLAKE      MANAGER

JONES      MANAGER

SCOTT      ANALYST

FORD      ANALYST

KING      PRESIDENT

14 filas seleccionadas.\*/

**/\* 12. Obtener el nombre y empleo de todos los empleados, ordenado primero por su trabajo y luego por su salario.**

SELECT ename, job FROM empleados ORDER BY job, sal;

ENAME	JOB
SCOTT	ANALYST
FORD	ANALYST
SMITH	CLERK
JAMES	CLERK
ADAMS	CLERK
MILLER	CLERK
CLARK	MANAGER
BLAKE	MANAGER
JONES	MANAGER
KING	PRESIDENT
WARD	SALESMAN
MARTIN	SALESMAN
TURNER	SALESMAN
ALLEN	SALESMAN

14 filas seleccionadas.\*/

**/\* 13. Idem, pero ordenando inversamente por empleo y normalmente por salario.**

SELECT ename, job FROM empleados ORDER BY job DESC, sal;

ENAME	JOB
WARD	SALESMAN
MARTIN	SALESMAN
TURNER	SALESMAN
ALLEN	SALESMAN
KING	PRESIDENT
CLARK	MANAGER
BLAKE	MANAGER
JONES	MANAGER
SMITH	CLERK

JAMES	CLERK
ADAMS	CLERK
MILLER	CLERK
SCOTT	ANALYST
FORD	ANALYST

14 filas seleccionadas.\*/

**/\* 14. Obtén los salarios y las comisiones de los empleados del departamento 30.**

`SELECT sal, comm FROM empleados WHERE deptno = 30;`

SAL	COMM
1600	300
1250	500
1250	1400
2850	
1500	0
950	

6 filas seleccionadas.\*/

**/\* 15. Idem, pero ordenado por comisión.**

`SELECT sal, comm FROM empleados WHERE deptno = 30 ORDER BY comm IS NULL, comm ASC;`

SAL	COMM
1500	0
1600	300
1250	500
1250	1400
2850	
950	

6 filas seleccionadas.\*/



**/\* 16. (a) Obtén las comisiones de todos los empleados.**

`SELECT comm FROM empleados;`

COMM
-----

300
-----

500
-----

1400
------

0
---

14 filas seleccionadas.\*/

**/\* 16. (b) Obtén las comisiones de los empleados de forma que no se repitan.**

`SELECT DISTINCT comm FROM empleados ORDER BY comm IS NULL, comm ASC;`

COMM
-----

0
---

300
-----

500
-----

1400
------

4 filas seleccionadas.\*/

**/\* 17. Obtén el nombre de empleado y su comisión SIN FILAS repetidas.**

`SELECT DISTINCT ename, comm FROM empleados ORDER BY ename;`

ENAME	COMM
-----	-----

ADAMS	
-------	--

ALLEN	300
BLAKE	
CLARK	
FORD	
JAMES	
JONES	
KING	
MARTIN	1400
MILLER	
SCOTT	
SMITH	
TURNER	0
WARD	500

14 filas seleccionadas.\*/

**/\* 18. Obtén los nombres de los empleados y sus salarios, de forma que no se repitan filas.**

`SELECT DISTINCT ename, sal FROM empleados ORDER BY ename;`

ENAME	SAL
-----	-----
ADAMS	1100
ALLEN	1600
BLAKE	2850
CLARK	2450
FORD	3000
JAMES	950
JONES	2975
KING	5000
MARTIN	1250
MILLER	1300
SCOTT	3000
SMITH	800
TURNER	1500
WARD	1250

14 filas seleccionadas.\*/

**/\* 19. Obtén las comisiones de los empleados y sus números de departamento, de forma que no se repitan filas.**

`SELECT DISTINCT comm, deptno FROM empleados ORDER BY comm IS NULL, comm ASC, deptno ASC;`

COMM	DEPTNO
0	30
300	30
500	30
1400	30
	10
	20
	30

7 filas seleccionadas.\*/

**/\* 20. Obten los nuevos salarios de los empleados del departamento 30, que resultarán de sumar a su salario una gratificación de 1000. Muestra también los nombres de los empleados.**

`SELECT ename, sal+1000 FROM empleados WHERE deptno = 30;`

ENAME	SAL+1000
ALLEN	2600
WARD	2250
MARTIN	2250
BLAKE	3850
TURNER	2500
JAMES	1950

6 filas seleccionadas.\*/

**/\* 21. Lo mismo que la anterior, pero mostrando también su salario original, y haz que la columna que almacena el nuevo salario se denomine NUEVO SALARIO.**

`SELECT ename, sal, sal+1000 AS nuevo_salario FROM empleados WHERE deptno = 30;`

ENAME	SAL	NUEVO_SALARIO
-------	-----	---------------

-----	-----	-----
ALLEN	1600	2600
WARD	1250	2250
MARTIN	1250	2250
BLAKE	2850	3850
TURNER	1500	2500
JAMES	950	1950

6 filas seleccionadas.\* /

**/\* 22. Halla los empleados que tienen una comisión superior a la mitad de su salario.**

`SELECT ename FROM empleados WHERE comm > (sal/2);`

ENAME

-----

MARTIN

1 fila seleccionada.\* /

**/\* 23. Halla los empleados que no tienen comisión, o que la tengan menor o igual que el 25% de su salario.**

`SELECT ename FROM empleados WHERE comm IS NULL OR comm <= (sal/4);`

ENAME

-----

SMITH  
ALLEN  
JONES  
BLAKE  
CLARK  
SCOTT  
KING  
TURNER  
ADAMS  
JAMES  
FORD  
MILLER

12 filas seleccionadas.\* /

**/\* 24. Obtén una lista de nombres de empleados y sus salarios, de forma que en la salida aparezca en todas las filas \Nombre:" y \Salario:" antes del respectivo campo. Hazlo de forma que selecciones exactamente tres expresiones.**

```
SELECT 'Nombre:' AS 'NOMBRE:', ename AS NOMBRE, CONCAT('Salario: ', sal) AS SALARIO FROM empleados;
```

```
'NOMBRE: NOMBRE    SALARIO
```

```
-----
Nombre: SMITH      Salario: 800
Nombre: ALLEN      Salario: 1600
Nombre: WARD       Salario: 1250
Nombre: JONES      Salario: 2975
Nombre: MARTIN     Salario: 1250
Nombre: BLAKE      Salario: 2850
Nombre: CLARK      Salario: 2450
Nombre: SCOTT      Salario: 3000
Nombre: KING       Salario: 5000
Nombre: TURNER     Salario: 1500
Nombre: ADAMS      Salario: 1100
Nombre: JAMES      Salario: 950
Nombre: FORD       Salario: 3000
Nombre: MILLER     Salario: 1300
```

14 filas seleccionadas.\* /

**/\* 25. Hallar el código, salario y comisión de los empleados cuyo código sea mayor que 7500.**

```
SELECT empno, sal, comm FROM empleados WHERE empno > 7500;
```

```
EMPNO    SAL    COMM
-----
7521     1250    500
7566     2975
7654     1250    1400
7698     2850
7782     2450
7788     3000
7839     5000
7844     1500    0
7876     1100
```

7900	950
7902	3000
7934	1300

12 filas seleccionadas.\*/

**/\* 26. Obtén todos los datos de los empleados que estén (considerando una ordenación ASCII por nombre) a partir de la J, inclusive.**

`SELECT * FROM empleados WHERE ename REGEXP '^[J-Z]';`

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17/12/80	800	20	
7521	WARD	SALESMAN	7698	22/02/81	1250	500	30
7566	JONES	MANAGER	7839	02/04/81	2975	20	
7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400	30
7788	SCOTT	ANALYST	7566	09/12/82	3000	20	
7839	KING	PRESIDENT		17/11/81	5000	10	
7844	TURNER	SALESMAN	7698	08/09/81	1500	0	30
7900	JAMES	CLERK	7698	03/12/81	950	30	
7934	MILLER	CLERK	7782	23/01/82	1300	10	

9 filas seleccionadas.\*/

**/\* 27. Obtén el salario, comisión y salario total (salario+comisión) de los empleados con comisión, ordenando el resultado por número de empleado.**

`SELECT sal, comm, (sal + comm) AS salario_total FROM empleados WHERE comm IS NOT NULL ORDER BY empno;`

SAL	COMM	SALARIO_TOTAL
1600	300	1900
1250	500	1750
1250	1400	2650
1500	0	1500

5 filas seleccionadas.\*/

**/\* 28. Lista la misma información, pero para los empleados que no tienen comisión.**

```
SELECT sal, comm, (sal + comm) AS salario_total FROM empleados WHERE comm IS NULL  
ORDER BY empno;
```

SAL	COMM	SALARIO_TOTAL
800		
2975		
2850		
2450		
3000		
5000		
1100		
950		
3000		
1300		

10 filas seleccionadas.\*/

**/\* 29. Muestra el nombre de los empleados que, teniendo un salario superior a 1000, tengan como jefe al empleado cuyo código es 7698.**

```
SELECT ename FROM empleados WHERE sal > 1000 AND mgr = 7698;
```

ENAME
ALLEN
WARD
MARTIN
TURNER

4 filas seleccionadas.\*/

**/\* 30. Halla el conjunto complementario del resultado del ejercicio anterior.**

```
SELECT ename FROM empleados WHERE sal < 1000 AND mgr != 7698;
```

ENAME
SMITH

1 fila seleccionada.\*/

**/\* 31. Indica para cada empleado el porcentaje que supone su comisión sobre su salario, ordenando el resultado por el nombre del mismo.**

```
SELECT (comm * 100) / (sal + comm) AS 'PORCENTAJE' FROM empleados ORDER BY  
ename;
```

PORCENTAJE

-----

15,7894737

52,8301887

0  
28,5714286

14 filas seleccionadas.\*/

**/\* 32. Hallar los empleados del departamento 10 cuyo nombre no contiene la cadena LA.**

```
SELECT ename FROM empleados WHERE deptno = 10 AND ename NOT REGEXP 'la';
```

ENAME

-----

KING  
MILLER

2 filas seleccionadas.\*/



**/\* 33. Obten los empleados que no son supervisados por ningún otro.**

```
SELECT ename FROM empleados WHERE mgr IS NULL;
```

ENAME
-------

-----
-------

KING
------

1 fila seleccionada.\*/

**/\* 34. Obtén los nombres de los departamentos que no sean Ventas (SALES) ni investigación (RESEARCH). Ordena el resultado por la localidad del departamento.**

```
SELECT dname FROM departamentos WHERE dname NOT IN ('SALES', 'RESEARCH')  
ORDER BY loc;
```

DNAME
-------

-----
-------

OPERATIONS
------------

ACCOUNTING
------------

2 filas seleccionadas.\*/

**/\* 35. Deseamos conocer el nombre de los empleados y el código del departamento de los administrativos(CLERK) que no trabajan en el departamento 10, y cuyo salario es superior a 800, ordenado por fecha de contratación.**

```
SELECT ename, deptno FROM empleados WHERE job = 'CLERK' AND deptno != 10 AND sal  
> 800 ORDER BY hiredate;
```

ENAME	DEPTNO
-------	--------

-----	-----
-------	-------

JAMES	30
-------	----

ADAMS	20
-------	----

2 filas seleccionadas.\*/

**/\* 36. Para los empleados que tengan comisión, obtén sus nombres y el cociente entre su salario y su comisión (excepto cuando la comisión sea cero), ordenando el resultado por nombre.**

SELECT ename, (sal/comm) AS cociente FROM empleados WHERE comm > 0 AND comm IS NOT NULL ORDER BY ename;

ENAME	COCIENTE
ALLEN	5,33333333
MARTIN	,892857143
WARD	2,5

3 filas seleccionadas.\*/

**/\* 37. Lista toda la información sobre los empleados cuyo nombre completo tenga exactamente 5 caracteres.**

SELECT \* FROM empleados WHERE ename LIKE '\_\_\_\_\_';  
 -- '\_\_\_\_\_' tiene 5 guiones bajos

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17/12/80	800	20	
7499	ALLEN	SALESMAN	7698	20/02/81	1600	300	30
7566	JONES	MANAGER	7839	02/04/81	2975	20	
7698	BLAKE	MANAGER	7839	01/05/81	2850	30	
7782	CLARK	MANAGER	7839	09/06/81	2450	10	
7788	SCOTT	ANALYST	7566	09/12/82	3000	20	
7876	ADAMS	CLERK	7788	12/01/83	1100	20	
7900	JAMES	CLERK	7698	03/12/81	950	30	

8 filas seleccionadas.\*/

**/\* 38. Lo mismo, pero para los empleados cuyo nombre tenga al menos cinco letras.**

SELECT \* FROM empleados WHERE ename LIKE '\_\_\_\_\_%';  
 -- '\_\_\_\_\_%' tiene 5 guiones bajos y un %

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
-------	-------	-----	-----	----------	-----	------	--------

7369	SMITH	CLERK	7902	17/12/80	800	20
7499	ALLEN	SALESMAN	7698	20/02/81	1600	300 30
7566	JONES	MANAGER	7839	02/04/81	2975	20
7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400 30
7698	BLAKE	MANAGER	7839	01/05/81	2850	30
7782	CLARK	MANAGER	7839	09/06/81	2450	10
7788	SCOTT	ANALYST	7566	09/12/82	3000	20
7844	TURNER	SALESMAN	7698	08/09/81	1500	0 30
7876	ADAMS	CLERK	7788	12/01/83	1100	20
7900	JAMES	CLERK	7698	03/12/81	950	30
7934	MILLER	CLERK	7782	23/01/82	1300	10

11 filas seleccionadas.\* /

**/\* 39. Halla los datos de los empleados que, o bien su nombre empieza por A y su salario es superior a 1000, o bien reciben comisión y trabajan en el departamento 30.**

SELECT \* FROM empleados WHERE (ename REGEXP '^a' AND sal > 1000) OR (comm > 0 AND deptno = 30);

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
-----							
7499	ALLEN	SALESMAN	7698	20/02/81	1600	300	30
7521	WARD	SALESMAN	7698	22/02/81	1250	500	30
7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400	30
7876	ADAMS	CLERK	7788	12/01/83	1100	20	

4 filas seleccionadas.\* /

**/\* 40. Halla el nombre, el salario y el sueldo total de todos los empleados, ordenando el resultado primero por salario y luego por el sueldo total. En el caso de que no tenga comisión, el sueldo total debe reflejar sólo el salario.**

SELECT ename, sal, (sal + IFNULL(comm,0)) AS salario\_total FROM empleados ORDER BY sal, salario\_total;

ENAME	SAL	SALARIO_TOTAL
-----		
SMITH	800	800
JAMES	950	950
ADAMS	1100	1100

WARD	1250	1750
MARTIN	1250	2650
MILLER	1300	1300
TURNER	1500	1500
ALLEN	1600	1900
CLARK	2450	2450
BLAKE	2850	2850
JONES	2975	2975
SCOTT	3000	3000
FORD	3000	3000
KING	5000	5000

14 filas seleccionadas.\*/

**/\* 41. Obtén el nombre, salario y la comisión de los empleados que perciben un salario que está entre la mitad de la comisión y la propia comisión.**

SELECT ename, sal, comm FROM empleados WHERE sal BETWEEN (comm / 2) AND comm;

ENAME	SAL	COMM
MARTIN	1250	1400

1 fila seleccionada.\*/

**/\* 42. Obtén el complementario del anterior.**

SELECT ename, sal, comm FROM empleados WHERE sal NOT BETWEEN (comm / 2) AND comm;

ENAME	SAL	COMM
ALLEN	1600	300
WARD	1250	500
TURNER	1500	0

3 filas seleccionadas.\*/

**/\* 43. Lista los nombres y empleos de aquellos empleados cuyo empleo acaba en MAN y cuyo nombre empieza por A.**

```
SELECT ename, job FROM empleados WHERE job REGEXP 'man$' AND ename REGEXP '^a';
```

ENAME	JOB
ALLEN	SALESMAN

1 fila seleccionada.\*/

**/\* 44. Intenta resolver la pregunta anterior con un predicado simple, es decir, de forma que en la cláusula WHERE no haya conectores lógicos como AND, OR, etc. Si ayuda a resolver la pregunta, se puede suponer que el nombre del empleado tiene al menos cinco letras.**

```
SELECT ename, job FROM empleados WHERE job REGEXP 'man$' HAVING LENGTH(ename) = 5;
```

ENAME	JOB
ALLEN	SALESMAN

1 fila seleccionada.\*/

**/\* 45. Halla los nombres de los empleados cuyo nombre tiene como máximo cinco caracteres.**

```
SELECT ename FROM empleados WHERE LENGTH(ename) <=5;
```

ENAME
SMITH
ALLEN
WARD
JONES
BLAKE
CLARK
SCOTT

KING  
ADAMS  
JAMES  
FORD

11 filas seleccionadas.\*/

**/\* 46. Suponiendo que el año próximo la subida del sueldo total de cada empleado será del 6 %, y el siguiente del 7 %, halla los nombres y el salario total actual, del año próximo y del siguiente, de cada empleado. Indique además con SI o NO, si el empleado tiene comisión. Como en la pregunta**

```
SELECT ename, (IF(comm IS NULL, 'NO', 'SI')) AS comm,  
(sal + IFNULL(comm, 0)) AS salario_total,  
TRUNCATE(((sal * 1.6) + IFNULL(comm, 0)),0) AS 'salario_proximo_año',  
TRUNCATE((((sal * 1.6) + IFNULL(comm, 0)) * 1.7), 0) AS 'salario_año_siguiente'  
FROM empleados ORDER BY ename;
```

ENAME	COMM	SALARIO_TOTAL	SALARIO_DEL_PROXIMO_AÑO	SALARIO_DEL_SIGUIENTE_AÑO
-------	------	---------------	-------------------------	---------------------------

ADAMS	NOSESABE	1100	1760	2992
ALLEN	SI	1900	3040	5168
BLAKE	NOSESABE	2850	4560	7752
CLARK	NOSESABE	2450	3920	6664
FORD	NOSESABE	3000	4800	8160
JAMES	NOSESABE	950	1520	2584
JONES	NOSESABE	2975	4760	8092
KING	NOSESABE	5000	8000	13600
MARTIN	SI	2650	4240	7208
MILLER	NOSESABE	1300	2080	3536
SCOTT	NOSESABE	3000	4800	8160
SMITH	NOSESABE	800	1280	2176
TURNER	NO	1500	2400	4080
WARD	SI	1750	2800	4760

14 filas seleccionadas.\*/

**/\* 47. Lista los nombres y fecha de contratación de aquellos empleados que no son vendedores (SALESMAN).**

```
SELECT ename, DATE_FORMAT(hiredate, '%d-%m-%y') AS hiredate FROM empleados  
WHERE job NOT IN ('salesman');
```

ENAME	HIREDATE
SMITH	17/12/80
JONES	02/04/81
BLAKE	01/05/81
CLARK	09/06/81
SCOTT	09/12/82
KING	17/11/81
ADAMS	12/01/83
JAMES	03/12/81
FORD	03/12/81
MILLER	23/01/82

10 filas seleccionadas.\*/

**/\* 48. Obtén la información disponible de los empleados cuyo número es uno de los siguientes: 7844, 7900, 7521, 7521, 7782, 7934, 7678 y 7369, pero que no sea uno de los siguientes: 7902, 7839, 7499 ni 7878. La sentencia no debe complicarse innecesariamente, y debe dar el resultado correcto independientemente de lo empleados almacenados en la base de datos.**

```
SELECT * FROM empleados
WHERE empno IN (7844, 7900, 7521, 7521, 7782, 7934, 7678, 7369)
AND empno NOT IN (7902, 7839, 7499, 7878);
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17/12/80	800	20	
7521	WARD	SALESMAN	7698	22/02/81	1250	500	30
7782	CLARK	MANAGER	7839	09/06/81	2450	10	
7844	TURNER	SALESMAN	7698	08/09/81	1500	0	30
7900	JAMES	CLERK	7698	03/12/81	950	30	
7934	MILLER	CLERK	7782	23/01/82	1300	10	

6 filas seleccionadas.\*/

**/\* 49. Ordena los empleados por su código de departamento, y luego de manera descendente por su número de empleado.**

**SELECT \* FROM empleados ORDER BY deptno, empno DESC;**

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7934	MILLER	CLERK	7782	23/01/82	1300	10	
7839	KING	PRESIDENT		17/11/81	5000	10	
7782	CLARK	MANAGER	7839	09/06/81	2450	10	
7902	FORD	ANALYST	7566	03/12/81	3000	20	
7876	ADAMS	CLERK	7788	12/01/83	1100	20	
7788	SCOTT	ANALYST	7566	09/12/82	3000	20	
7566	JONES	MANAGER	7839	02/04/81	2975	20	
7369	SMITH	CLERK	7902	17/12/80	800	20	
7900	JAMES	CLERK	7698	03/12/81	950	30	
7844	TURNER	SALESMAN	7698	08/09/81	1500	0	30
7698	BLAKE	MANAGER	7839	01/05/81	2850	30	
7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400	30
7521	WARD	SALESMAN	7698	22/02/81	1250	500	30
7499	ALLEN	SALESMAN	7698	20/02/81	1600	300	30

14 filas seleccionadas.\*/

**/\* 50. Para los empleados que tengan como jefe a un empleado con código mayor que el suyo, obtén los que reciben de salario más de 1000 y menos de 2000, o que están en el departamento 30.**

**SELECT ename FROM empleados WHERE mgr > empno AND ((sal BETWEEN 1000 AND 2000) OR deptno = 30);**

ENAME  
-----  
ALLEN  
WARD  
MARTIN  
BLAKE

4 filas seleccionadas.\*/



**/\* 51. Obtén el salario más alto de la empresa, el total destinado a comisiones y el número de empleados.**

`SELECT MAX(sal), SUM(comm), COUNT(*) FROM empleados;`

MAX(SAL)	SUM(COMM)	COUNT(*)
5000	2200	14

1 fila seleccionada.\*/

**/\* 52. Halla los datos de los empleados cuyo salario es mayor que el del empleado de código 7934, ordenando por el salario.**

`SELECT * FROM empleados WHERE sal > (SELECT sal FROM empleados WHERE empno = 7934) ORDER BY sal;`

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7844	TURNER	SALESMAN	7698	08/09/81	1500	0	30
7499	ALLEN	SALESMAN	7698	20/02/81	1600	300	30
7782	CLARK	MANAGER	7839	09/06/81	2450	10	
7698	BLAKE	MANAGER	7839	01/05/81	2850	30	
7566	JONES	MANAGER	7839	02/04/81	2975	20	
7788	SCOTT	ANALYST	7566	09/12/82	3000	20	
7902	FORD	ANALYST	7566	03/12/81	3000	20	
7839	KING	PRESIDENT		17/11/81	5000	10	

8 filas seleccionadas.\*/

**/\* 53. Obtén información en la que se reflejen los nombres, empleos y salarios tanto de los empleados que superan en salario a Allen como del propio Allen.**

`SELECT ename, job, sal FROM empleados WHERE sal >= (SELECT sal FROM empleados WHERE ename = 'ALLEN');`

ENAME	JOB	SAL
ALLEN	SALESMAN	1600
JONES	MANAGER	2975
BLAKE	MANAGER	2850

CLARK	MANAGER	2450
SCOTT	ANALYST	3000
KING	PRESIDENT	5000
FORD	ANALYST	3000

7 filas seleccionadas.\*/

**/\* 54. Halla el nombre del último empleado por orden alfabético.**

```
SELECT MAX(ename) FROM empleados;
```

```
MAX(ENAME)
```

```
-----
```

```
WARD
```

1 fila seleccionada.\*/

**/\* 55. Halla el salario más alto, el más bajo, y la diferencia entre ellos.**

```
SELECT MAX(sal) AS sal_mas_alto, MIN(sal) AS sal_mas_bajo, (MAX(sal) - MIN(sal)) AS
diferencia FROM empleados;
```

```
SAL_MAS_ALTO SAL_MAS_BAJA DIFERENCIA
```

```
-----
```

```
5000      800      4200
```

1 fila seleccionada.\*/

**/\* 56. Sin conocer los resultados del ejercicio anterior, ¿quienes reciben el salario más alto y el más bajo, y a cuánto ascienden estos salarios?**

```
SELECT ename, sal FROM empleados
WHERE (sal = (SELECT MAX(sal) FROM empleados))
OR (sal = (SELECT MIN(sal) FROM empleados)) ORDER BY sal;
```

```
ENAME      SAL
```

```
-----
```

```
KING      5000
```

```
SMITH      800
```

2 filas seleccionadas.\*/

**/\* 57. Considerando empleados con salario menor de 5000, halla la media de los salarios de los departamentos cuyo salario mínimo supera a 900. Muestra también el código y el nombre de los departamentos.**

```
SELECT d.dname, AVG(e.sal)
FROM empleados e
JOIN departamentos d
USING (deptno)
WHERE e.sal < 5000
GROUP BY e.deptno
HAVING MIN(e.sal) > 900
ORDER BY d.dname;
```

DNAME	AVG(A.SAL)
-----	-----
ACCOUNTING	1875
SALES	1566,66667

2 filas seleccionadas.\*/

**/\* 58. ¿Que empleados trabajan en ciudades de más de cinco letras? Ordena el resultado inversamente por ciudades y normalmente por los nombres de los empleados.**

```
SELECT ename FROM empleados e
JOIN departamentos d
USING (deptno)
WHERE LENGTH(loc) > 5
ORDER BY d.loc DESC, ename;
```

ENAME
-----
CLARK
KING
MILLER
ADAMS
FORD
JONES

SCOTT  
SMITH  
ALLEN  
BLAKE  
JAMES  
MARTIN  
TURNER  
WARD

14 filas seleccionadas.\*/

**/\* 59. Halla los empleados cuyo salario supera o coincide con la media del salario de la empresa.**

```
SELECT ename FROM empleados e
JOIN departamentos d
USING (deptno)
WHERE sal >= (SELECT AVG(sal) FROM empleados);
```

ENAME  
-----  
JONES  
BLAKE  
CLARK  
SCOTT  
KING  
FORD

6 filas seleccionadas.\*/

**/\* 60. Obten los empleados cuyo salario supera al de sus compañeros de departamento.**

```
SELECT ename FROM empleados e
JOIN departamentos d
USING (deptno)
WHERE sal IN (SELECT MAX(sal) FROM empleados GROUP BY DEPTNO)
ORDER BY empno;
```

ENAME  
-----  
BLAKE  
SCOTT  
KING

FORD

4 filas seleccionadas.\*/

**/\* 61. ¿Cuántos empleos diferentes, cuántos empleados, y cuántos salarios diferentes encontramos en el departamento 30, y ¿ a cuánto asciende la suma de salarios de dicho departamento?**

```
SELECT COUNT(DISTINCT job) AS num_job,  
COUNT(DISTINCT empno) AS num_emp,  
COUNT(DISTINCT sal) AS num_sal,  
SUM(sal) FROM empleados  
WHERE deptno = 30;
```

NUM_JOB	NUM_EMP	NUM_SAL	SUM(SAL)
3	6	5	9400

1 fila seleccionada.\*/

**/\* 62. ¿Cuántos empleados tienen comisión?**

```
SELECT COUNT(*) FROM empleados WHERE comm IS NOT NULL AND comm > 0;
```

COUNT(*)
3

1 fila seleccionada.\*/

**/\* 63. ¿Cuántos empleados tiene el departamento 20?**

```
SELECT COUNT(*) FROM empleados WHERE deptno = 20;
```

NUM_EMP
5

1 fila seleccionada.\*/

**/\* 64. Halla los departamentos que tienen más de tres empleados, y el número de empleados de los mismos.**

```
SELECT deptno, COUNT(*) AS 'num_emp' FROM empleados GROUP BY deptno HAVING  
COUNT(*) > 3;
```

DEPTNO	NUM_EMP
20	5
30	6

2 filas seleccionadas.\*/

**/\* 65. Obten los empleados del departamento 10 que tienen el mismo empleo que alguien del departamento de Ventas. Desconocemos el código de dicho departamento.**

```
SELECT DISTINCT ename FROM empleados e  
JOIN departamentos d  
USING (deptno)  
WHERE deptno = 10 AND job IN  
      (SELECT DISTINCT job FROM empleados e  
      JOIN departamentos d  
      USING (deptno)  
      WHERE dname = 'SALES');
```

ENAME
CLARK
MILLER

2 filas seleccionadas.\*/

**/\* 66. Halla los empleados que tienen por lo menos un empleado a su mando, ordenados inversamente por nombre.**

```
SELECT ename FROM empleados WHERE empno IN (SELECT mgr FROM empleados)  
ORDER BY ename DESC;
```

ENAME

-----  
SCOTT  
KING  
JONES  
FORD  
CLARK  
BLAKE

6 filas seleccionadas.\*/

**/\* 67. Obtén información sobre los empleados que tienen el mismo trabajo que algún empleado que trabaje en Chicago.**

FROM empleados e  
JOIN departamentos d  
USING (deptno)  
WHERE job IN  
    (SELECT DISTINCT job FROM empleados e  
      JOIN departamentos d  
      USING (deptno)  
      WHERE loc = 'CHICAGO')  
ORDER BY job;

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17/12/80	800	20	
7876	ADAMS	CLERK	7788	12/01/83	1100	20	
7934	MILLER	CLERK	7782	23/01/82	1300	10	
7900	JAMES	CLERK	7698	03/12/81	950	30	
7566	JONES	MANAGER	7839	02/04/81	2975	20	
7782	CLARK	MANAGER	7839	09/06/81	2450	10	
7698	BLAKE	MANAGER	7839	01/05/81	2850	30	
7499	ALLEN	SALESMAN	7698	20/02/81	1600	300	30
7654	MARTIN	SALESMAN	7698	28/09/81	1250	1400	30
7844	TURNER	SALESMAN	7698	08/09/81	1500	0	30
7521	WARD	SALESMAN	7698	22/02/81	1250	500	30

11 filas seleccionadas.\*/

**/\* 68. ¿Qué empleos distintos encontramos en la empresa, y cuántos empleados**

**desempeñan cada uno de ellos?**

```
SELECT job, COUNT(*)  
FROM empleados e  
JOIN departamentos d  
USING (deptno)  
GROUP BY job  
ORDER BY job;
```

JOB	COUNT(*)
ANALYST	2
CLERK	4
MANAGER	3
PRESIDENT	1
SALESMAN	4

5 filas seleccionadas.\*/

**/\* 69. Halla la suma de salarios de cada departamento.**

```
SELECT dname, SUM(sal)  
FROM empleados e  
JOIN departamentos d  
USING (deptno)  
GROUP BY dname  
ORDER BY dname;
```

DNAME	SUM(SAL)
ACCOUNTING	8750
RESEARCH	10875
SALES	9400

3 filas seleccionadas.\*/

**/\* 70. Obtén todos los departamentos sin empleados.**

```
SELECT DISTINCT dname
```



```
FROM departamentos
WHERE dname NOT IN
      (SELECT DISTINCT dname FROM empleados e
       JOIN departamentos d
       USING (deptno));
```

DNAME

-----

OPERATIONS

1 fila seleccionada.\*/

**/\* 71. Halla los empleados que no tienen a otro empleado a sus ordenes.**

```
SELECT ename FROM empleados WHERE empno NOT IN (SELECT DISTINCT mgr FROM
empleados WHERE mgr IS NOT NULL);
```

ENAME

-----

SMITH  
ALLEN  
WARD  
MARTIN  
TURNER  
ADAMS  
JAMES  
MILLER

8 filas seleccionadas.\*/

**/\* 72. ¿Cuántos empleados hay en cada departamento, y cual es la media anual del salario de cada uno (el salario almacenado es mensual)? Indique el nombre del departamento para clarificar el resultado.**

```
SELECT dname, AVG(sal), COUNT(*)
FROM empleados e
JOIN departamentos d
USING (deptno)
GROUP BY dname
ORDER BY dname;
```

DNAME	AVG(SAL)	COUNT(*)
ACCOUNTING	2916,66667	3
RESEARCH	2175	5
SALES	1566,66667	6

3 filas seleccionadas.\*/

**/\* 73. Halla los empleados del departamento 30, por orden descendente de comisión**

```
SELECT ename, comm FROM empleados WHERE deptno = 30 ORDER BY comm IS NULL
DESC, comm DESC;
```

ENAME
BLAKE
JAMES
MARTIN
WARD
ALLEN
TURNER

6 filas seleccionadas.\*/

**/\* 74. Obtén los empleados que trabajan en Dallas o New York.**

```
SELECT * FROM empleados e
JOIN departamentos d
USING (deptno)
WHERE loc IN ('DALLAS', 'NEW YORK');
```

ENAME
CLARK
KING
MILLER
SMITH
ADAMS
FORD
SCOTT
JONES

8 filas seleccionadas.\* /

**/\* 75. Obtén un listado en el que se reflejen los empleados y los nombres de sus jefes. En el listado deben aparecer todos los empleados, aunque no tengan jefe, poniendo un nulo el nombre de este.**

```
SELECT e.ename AS ename_empleado, m.ename AS ename_mgr
FROM empleados e
LEFT JOIN empleados m
ON e.mgr = m.empno
ORDER BY ename_mgr DESC;
```

ENAME	ENAME
-----	-----
FORD	JONES
SCOTT	JONES
JAMES	BLAKE
TURNER	BLAKE
MARTIN	BLAKE
WARD	BLAKE
ALLEN	BLAKE
MILLER	CLARK
ADAMS	SCOTT
CLARK	KING
BLAKE	KING
JONES	KING
SMITH	FORD
KING	

14 filas seleccionadas.\* /

**/\* 76. Lista los empleados que tengan el mayor salario de su departamento, mostrando el nombre del empleado, su salario y el nombre del departamento.**

```
SELECT ename, sal, dname FROM empleados
JOIN departamentos
USING (deptno)
WHERE sal IN (
    SELECT MAX(sal) FROM empleados
    JOIN departamentos
```

```

        USING (deptno)
        GROUP BY dname)
ORDER BY sal;

```

ENAME	SAL	DNAME
BLAKE	2850	SALES
FORD	3000	RESEARCH
SCOTT	3000	RESEARCH
KING	5000	ACCOUNTING

4 filas seleccionadas.\*/

**/\* 77. Deseamos saber cuántos empleados supervisa cada jefe. Para ello, obtén un listado en el que se reflejen el código y el nombre de cada jefe, junto al número de empleados que supervisa directamente. Como puede haber empleados sin jefe, para estos se indicará solo el número de ellos, y los valores restantes (código y nombre del jefe) se dejarán como nulos.**

```

SELECT m.ename AS mgr_name, e.mgr AS empno, COUNT(*) AS num_empleados
FROM empleados e
LEFT JOIN empleados m
ON e.mgr = m.empno
GROUP BY e.mgr
ORDER BY m.empno IS NULL DESC, e.mgr DESC;

```

ENAME	EMPNO	NUM_EMPLEADOS
		1
FORD	7902	1
KING	7839	3
BLAKE	7698	5
CLARK	7782	1
JONES	7566	2
SCOTT	7788	1

7 filas seleccionadas.\*/

**/\* 78. Hallar el departamento cuya suma de salarios sea la más alta, mostrando esta suma de salarios y el nombre del departamento**

```
SELECT dname, SUM(sal)
FROM empleados e
JOIN departamentos d
USING (deptno)
GROUP BY dname
ORDER BY SUM(sal) DESC LIMIT 1;
```

DNAME	SUM(SAL)
RESEARCH	10875

1 fila seleccionada.\*/

**/\* 79. Obtén los datos de los empleados que cobren los dos mayores salarios de la empresa. (Nota: Procure hacer la consulta de forma que sea fácil obtener los empleados de los N mayores salarios)**

```
SELECT empno, ename, job, mgr, DATE_FORMAT(hiredate, '%d-%m-%y') AS hiredate, sal,
comm, deptno
FROM empleados
JOIN departamentos
USING (deptno)
WHERE sal = (SELECT MAX(sal) FROM empleados)
      OR sal = (SELECT MAX(sal) FROM empleados WHERE sal NOT IN (SELECT DISTINCT
MAX(sal) FROM empleados))
ORDER BY sal DESC;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7839	KING	PRESIDENT		17/11/81	5000	10	
7788	SCOTT	ANALYST	7566	09/12/82	3000	20	
7902	FORD	ANALYST	7566	03/12/81	3000	20	

3 filas seleccionadas.\*/

**/\* 80. Obtén las localidades que no tienen departamentos sin empleados y en las que trabajen al menos cuatro empleados. Indica también el número de empleados que trabajan en esas localidades. (Nota: Por ejemplo, puede que en A Coruña existan dos departamentos, uno con más de cuatro empleados y otro sin empleados, en tal caso, A Coruña no debe aparecer en el resultado, puesto que tiene un departamento SIN EMPLEADOS, a pesar de tener otro con empleados y tener más de cuatro empleados EN**

**TOTAL. ATENCIÓN, la restricción de que tienen que ser cuatro empleados se refiere a la totalidad de los departamentos de la localidad.)**

```
SELECT d.loc, COUNT(*) AS numero_empleados
FROM departamentos d
LEFT JOIN empleados e
USING (deptno)
GROUP BY d.deptno
HAVING numero_empleados >= 4;
```

LOC	NUMERO_EMPLEADOS
DALLAS	5
CHICAGO	6

2 filas seleccionadas.\*/