Course: Database management

Unit: Introduction to SQL II Material: Queries MariaDB 2

Check the following database:

DEPARTMENTS:

ı	num	name	town_code
ï	10	ACCOUNTING	SVQ
i	20	RESEARCH	MAD
i	30	SALES	BCN
i	48	PRODUCTION	BIO

EMPLOYEES:

num	surname	name	manager	İ	start_date	i	salary	commission	dept_num	occu_code
808	BANDERAS	ANTONIO	7839	i	1991-01-09	ï	2885	NULL	20	MAN
7369	SÁNCHEZ	SERGIO	7902	İ	1990-12-17	Ĺ	1040	NULL	20	EMP
7499	ARROYO	MARTA	7698	ĺ	1990-02-20	Ĺ	1500	398	38	SAL
7521	SALA	RAUL	7698	ĺ	1991-02-22	ĺ	1625	658	38	SAL
7566	JIMÉNEZ	JUDIT	7839	ĺ	1991-84-82	Ĺ	2900	NULL	20	MAN
7654	MARTÍN	MONICA	7698	ĺ	1991-09-29	ĺ	1600	1020	38	SAL
7698	NEGRO	BARTOLOME	7839	ĺ	1991-05-01	Ĺ	3005	NULL	38	MAN
7782	CEREZO	ENRIQUE	7839	Ī	1991-06-09	ĺ	2885	NULL	10	MAN
7788	GIL	JESUS	7566	ĺ	1991-11-09	ĺ	3666	NULL	20	ANA
7844	TOVAR	LUIS	7698	ĺ	1991-09-08	ĺ	1350	9	38	SAL
7876	ALONSO	FERNANDO	7788	İ	1991-89-23	Ĺ	1430	NULL	28	EMP
7900	JIMENO	XAVIER	7698	İ	1991-12-03	ĺ	1335	NULL	38	EMP
7902	FERNÁNDEZ	ANA	7566	İ	1991-12-03	Ĺ	3666	NULL	28	ANA
7934	MUÑOZ	ANTONIA	7782	1	1992-01-23	l	1690	NULL	10	EMP
8801	RUIZ	FERNANDA	7839	İ	1992-86-18	Ĺ	2885	NULL	28	MAN

OCCUPATIONS:

code	name
ANA	ANALYST
EMP	EMPLOYEE
MAN	MANAGER
PRE	PRESIDENT
SAL	SALESMAN

TOWNS:

code	name
BCN	BARCELONA
BIO	BILBAO
MAD	MADRID
SVQ	SEVILLA

Import the next database:

```
CREATE DATABASE IF NOT EXISTS `EMPLOYEESDBNORMAL`;

USE `EMPLOYEESDBNORMAL`;

CREATE TABLE IF NOT EXISTS `DEPARTMENTS` (
   `num` int(11) NOT NULL,
   `name` varchar(30) NOT NULL,
```

```
`town code` varchar(3) DEFAULT NULL,
 PRIMARY KEY (`num`),
 KEY `town code` (`town code`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
INSERT INTO `DEPARTMENTS` (`num`, `name`, `town code`) VALUES
(10, 'ACCOUNTING', 'SVQ'),
(20, 'RESEARCH', 'MAD'),
(30, 'SALES', 'BCN'),
(40, 'PRODUCTION', 'BIO');
CREATE TABLE IF NOT EXISTS `EMPLOYEES` (
  `num` int(11) NOT NULL,
  `surname` varchar(50) NOT NULL,
  `name` varchar(50) NOT NULL,
  `manager` int(11) DEFAULT NULL,
  `start date` date DEFAULT NULL,
  `salary` int(11) DEFAULT NULL,
  `commission` int(11) DEFAULT NULL,
  `dept num` int(11) DEFAULT NULL,
  `occu code` varchar(3) DEFAULT NULL,
 PRIMARY KEY (`num`),
 KEY `dept num` (`dept_num`),
 KEY `occu code` (`occu_code`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
INSERT INTO `EMPLOYEES` (`num`, `surname`, `name`, `manager`, `start date`, `salary`,
`commission`, `dept num`, `occu code`) VALUES
(800, 'BANDERAS', 'ANTONIO', 7839, '1991-01-09', 2885, NULL, 20,
'MAN'), (7369, 'SÁNCHEZ', 'SERGIO', 7902, '1990-12-17', 1040, NULL, 20,
'EMP'), (7499, 'ARROYO', 'MARTA', 7698, '1990-02-20', 1500, 390, 30,
'SAL'), (7521, 'SALA', 'RAUL', 7698, '1991-02-22', 1625, 650, 30,
'SAL'), (7566, 'JIMÉNEZ', 'JUDIT', 7839, '1991-04-02', 2900, NULL, 20,
'MAN'), (7654, 'MARTÍN', 'MONICA', 7698, '1991-09-29', 1600, 1020, 30,
'SAL'), (7698, 'NEGRO', 'BARTOLOME', 7839, '1991-05-01', 3005, NULL,
30, 'MAN'), (7782, 'CEREZO', 'ENRIQUE', 7839, '1991-06-09', 2885, NULL,
10, 'MAN'), (7788, 'GIL', 'JESUS', 7566, '1991-11-09', 3000, NULL, 20,
'ANA'), (7844, 'TOVAR', 'LUIS', 7698, '1991-09-08', 1350, 0, 30,
'SAL'),
(7876, 'ALONSO', 'FERNANDO', 7788, '1991-09-23', 1430, NULL, 20,
'EMP'), (7900, 'JIMENO', 'XAVIER', 7698, '1991-12-03', 1335, NULL, 30,
'EMP'), (7902, 'FERNÁNDEZ', 'ANA', 7566, '1991-12-03', 3000, NULL, 20,
'ANA'), (7934, 'MUÑOZ', 'ANTONIA', 7782, '1992-01-23', 1690, NULL, 10,
'EMP'), (8001, 'RUIZ', 'FERNANDA', 7839, '1992-06-10', 2885, NULL, 20,
'MAN');
CREATE TABLE IF NOT EXISTS `OCCUPATIONS` (
  `code` varchar(3) NOT NULL,
  `name` varchar(30) NOT NULL,
 PRIMARY KEY (`code`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
INSERT INTO `OCCUPATIONS` (`code`, `name`) VALUES
('ANA', 'ANALYST'),
('EMP', 'EMPLOYEE'),
('MAN', 'MANAGER'),
('PRE', 'PRESIDENT'),
('SAL', 'SALESMAN');
CREATE TABLE IF NOT EXISTS 'TOWNS' (
  `code` varchar(3) NOT NULL,
  `name` varchar(30) NOT NULL,
```

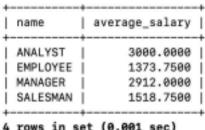
```
PRIMARY KEY (`code`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
INSERT INTO `TOWNS` (`code`, `name`) VALUES
('BCN', 'BARCELONA'),
('BIO', 'BILBAO'),
('MAD', 'MADRID'), ('SVQ', 'SEVILLA');
ALTER TABLE `DEPARTMENTS`
 ADD CONSTRAINT `DEPARTMENTS_ibfk_1` FOREIGN KEY (`town_code`) REFERENCES `TOWNS`
(`code`);
ALTER TABLE `EMPLOYEES`
 ADD CONSTRAINT `EMPLOYEES ibfk 1` FOREIGN KEY ('dept num') REFERENCES 'DEPARTMENTS'
 ADD CONSTRAINT `EMPLOYEES ibfk 2` FOREIGN KEY (`occu code`) REFERENCES `OCCUPATIONS`
(`code`);
```

Do the following queries with that database:

1. Display the number of employees in each department. Use GROUP BY to group by department.

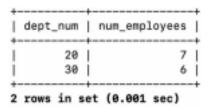
dept_num	N_employees
10	2
20	7
30	6
+	
3 rows in se	et (0.001 sec)

2. For each occupations obtain the average of salary.



4 rows in set (0.001 sec)

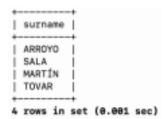
3. Display the departments with more than 5 employees. Use GROUP BY to group by department and HAVING to establish the condition on the groups.



4. Find the average wages (="media de los salarios") of each department (use the function avg and GROUP BY).

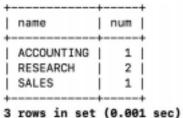
dept_num	average_wages
1 10	2287.5000
26	2448.5714
36	1735.8333
+	-+
3 rows in	set (0.002 sec)

5. Display the surname of the salesmen of the 'SALES' department.



6. Display the sum of salaries of the 'SALES' department.

7. Display the count of employees with occupation "EMPLOYEE" in every department (show the name of the department).



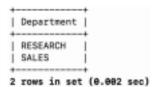
3 10W5 III SEC (0.001 SEC)

8. Show the number of different occupations in each department.

Department	Occupation	Number_of_employees
ACCOUNTING	EMPLOYEE	1
ACCOUNTING	MANAGER	j 1
RESEARCH	ANALYST	j 2
RESEARCH	EMPLOYEE	j 2
RESEARCH	MANAGER	į a
SALES	EMPLOYEE	j 1
SALES	MANAGER	j 1
SALES	SALESMAN	j 4

8 rows in set (0.004 sec)

9. Show departments that have more than two people working in the same occupation.

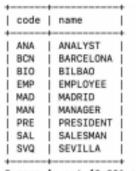


10. Displays a query that is the union between the table OCCUPATIONS and TOWNS.



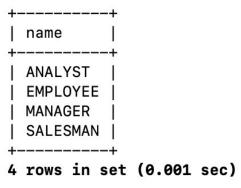
9 rows in set (0.001 sec)

11. Do the same query than in exercise 10 but order the results by



name. 9 rows in set (0.001 sec)

12. Select the occupation names of all the employees of the department with name 'RESEARCH' and do the union of this query with the selection of the occupation names of the employees of the department with name 'SALES'. Use union operator.



13. Repeat the last query showing the repeated results (union all).

```
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
name |
na
```

13 rows in set (0.001 sec)

14. Display the number of sellers in the 'SALES' department.

15. Display the surnames and occupations of the employees of the 'SALES'

 +-	surname		name
l	JIMENO	Ī	EMPLOYEE
ĺ	NEGRO	ĺ	MANAGER
ĺ	ARROYO	ĺ	SALESMAN
ĺ	SALA	ĺ	SALESMAN
İ	MARTÍN	İ	SALESMAN
İ	TOVAR	Ì	SALESMAN

department. 6 rows in set (0.001 sec)

16. Display the number of employees and occupations of the employees of the 'SALES' department.

name	number_of_employees
EMPLOYEE	1
MANAGER	1
SALESMAN	4

3 rows in set (0.001 sec)

17. Display the number of employees of each department whose profession is "EMPLOYEE".

+	++ number_of_employees +
ACCOUNTING RESEARCH SALES	1 2 1

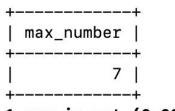
3 rows in set (0.001 sec)

18. Display de department names and the count of employees working into

name	number_of_employees
ACCOUNTING	2
RESEARCH	7
SALES	1 6

them. 3 rows in set (0.001 sec)

19. Display the maximum number of employees of all the departments (clue: you need exercise 18 as a subquery and you should use MAX function).



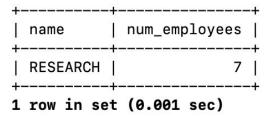
1 row in set (0.001 sec)

20. Show the departments whose average salary is greater than the average of salaries of all employees.

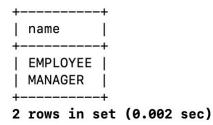
+	+ dept_num	average_salary	-+ -+
i	10	2287.5000	i
1	20	2448.5714	1
+	+		-+

2 rows in set (0.001 sec)

21. DANGER, this is for PROS: Display the name of the department with more employees and their number of employees (clue you must use HAVING with a subselect inside).



22. Repeat 12 changing "union" for "intersect".



23. Repeat 22 but do not use intersect operator to query the same data (clue: IN