**Maximum salary of all bosses that work in the same location (not department!) as their minions:**

SELECT MAX(e1.salary), d.location FROM emps AS e1 INNER JOIN emps AS e2 ON e1.id = e2.manager INNER JOIN deps AS d ON e1.dep\_id = d.id GROUP BY d.location HAVING e2.manager IS NOT NULL;

*SELECT MAX(m.salary) FROM emps AS e JOIN deps AS d ON e.deps\_id = d.id JOIN emps AS m ON m.id = e.manager JOIN deps AS dm ON m.dep\_id = dm.id WHERE d.location = dm.location;*

**Average bonus of each location (not department):**

SELECT AVG(e.bonus), d.location FROM emps AS e JOIN deps AS d ON e.deps\_id = d.id GROUP BY d.location;

*SELECT AVG(e.bonus), d.location FROM emps AS e JOIN (also RIGHT JOIN) deps AS d ON e.dep\_id = d.id GROUP BY d.location;*

**Name and department of the highest paid (bonus included) worker:**

SELECT e.name, d.name FROM emps as e JOIN deps as d ON e.deps\_id = d.id WHERE (SELECT MAX(e.salary+e.bonus));

*SELECT e.name, d.name FROM emps as e JOIN deps as d ON e.deps\_id = d.id WHERE (SELECT MAX(salary + bonus) FROM emps);* (Hay que compararlos y hay que poner de qué tabla son otra vez)

*SELECT e.name, d.name FROM emps as e JOIN deps as d ON e.deps\_id = d.id WHERE (e.salary + e.bonus) = SELECT MAX(salary + bonus) FROM emps;*

*SELECT e.name, d.name FROM emps as e JOIN deps as d ON e.deps\_id = d.id WHERE (e.salary + e.bonus) >= ALL(SELECT salary + bonus FROM emps);*

**Name of the manager with the lowest-paid employee:**

SELECT e1.name FROM emps AS e1 JOIN emps AS e2 ON e2.manager = e1.id WHERE(SELECT MIN(e2.salary));

*SELECT e.name FROM emps AS e JOIN emps AS m ON m.id = e.manager WHERE e.salary = (SELECT MIN(salary) FROM emps);* (Hay que compararlos)

**Shows all departments and the difference between the highest and lowest paid workers of each department:**

SELECT d.name, (MAX(e.salary)-MIN(e.salary)) FROM emps AS e JOIN deps AS d ON e.deps\_id = d.id GROUP BY d;

*SELECT d.name, (MAX(e.salary + e.bonus) – MIN(e.salary + e.bonus)) ‘difference’ FROM emps AS e JOIN deps AS d ON d.id = e.dep\_id GROUP BY d.id;* (Hay que especificar el nombre de la columna siempre que se haga GROUP BY)

**Salaries of %[abc]\_\_[ea][!lp]%:**

SELECT salary FROM emps WHERE name LIKE ‘%[abc]\_\_[ea][!lp]%’;

*SELECT salary FROM emps WHERE name LIKE ‘%[abc]\_\_[ea][!lp]%’;*

**Names and locations of the managers of everybody that works in either Human Resources or R&D:**

SELECT e.name, d.location FROM emps AS e JOIN deps AS d ON e.dep\_id = d.id WHERE d.name LIKE ‘%Human Resources%’ OR ‘%R&D%’;

*SELECT m.name, dm.location FROM emps AS m JOIN emps AS e ON e.manager = m.id JOIN deps AS d ON d.id = e.dep\_id JOIN deps AS dm ON dm.id = m.dep\_id WHERE d.name = ‘R&D’ OR d.name = ‘Human Resources’;* (Muchas cosas mal, repasar a fondo)

*SELECT m.name, dm.location FROM emps AS m JOIN emps AS e ON e.manager = m.id JOIN deps AS d ON m.dep\_id = d.id WHERE e.dep\_id ANY (SELECT id FROM deps WHERE name LIKE ‘Human Resources’ OR ‘R&D’);*

**Reassigns all employees to the department of the lowest-paid employee:**

UPDATE emps SET dep\_id = (SELECT dep\_id FROM emps WHERE (SELECT MIN(salary) FROM emps));

*UPDATE emps SET dep\_id = (SELECT dep\_id FROM emps WHERE salary = (SELECT MIN(salary) FROM emps));* (Hay que compararlos)

**Creates a view that shows the id and average salary of all departments:**

SELECT d.id, AVG(e.salary) ‘average\_salary’ FROM emps AS e JOIN deps AS d GROUP BY d;

*CREATE VIEW id\_averagesalary\_data AS SELECT dep\_id, AVG(salary) FROM emps GROUP BY dep\_id;* (Create the view first)

**Deletes all employees that have an “a” and an “l” somewhere in their name and earn less than the average salary:**

DELETE FROM emps WHERE salary < (SELECT AVG(salary) FROM emps) AND name LIKE ‘%a%l%’;

*DELETE FROM emps WHERE salary < (SELECT AVG(salary) FROM emps) AND (name LIKE ‘%a%l%’ OR name LIKE ‘%l%a%’);* (Acordarse del orden)

*DELETE FROM emps WHERE salary < (SELECT AVG(salary) FROM emps) AND (name LIKE ‘%a%’ AND name LIKE ‘%l%’);*

**Names, locations and number of employees of all departments, including those that have no employees:**

SELECT d.name, d.location, COUNT(e) FROM emps AS e JOIN deps AS d ON e.dep\_id = d.id;

*SELECT d.name, d.location, COUNT(e.id) FROM emps AS e JOIN deps AS d ON e.dep\_id = d.id GROUP BY e.dep\_id;* (COUNT solo acepta columnas y falta agrupar por departamentos)

**Duplicates the bonus of all employees that work the department of their manager:**

UPDATE emps AS e1 SET bonus = 2\*bonus WHERE(SELECT e1.id FROM e1 JOIN emps AS e2 ON e1.manager = e2.id WHERE e1.dep\_id = e2.dep\_id);

*UPDATE emps AS e SET bonus = 2\*bonus WHERE e.dep\_id = (SELECT m.dep\_id FROM emps AS m WHERE m.id = e.manager);*

**Names of all departments with several employees:**

SELECT d.name FROM deps AS d JOIN emps AS e ON d.id = e.dep\_id WHERE (SELECT COUNT(e) > 0);

*SELECT d.name FROM emps AS e JOIN deps AS d ON e.dep\_id = d.id GROUP BY d.id HAVING COUNT(e.id) > 1;*

**In how many locations we have departments:**

SELECT COUNT(location) FROM deps;

*SELECT COUNT(DISTINCT(location)) FROM deps;*

**How many employees have each benefit:**

SELECT COUNT(eb.emp\_id) FROM emps\_benefits AS eb JOIN benefits AS b ON eb.benefit\_id = b.id GROUP BY b;

**How many benefits each employee have:**

SELECT COUNT(eb.benefit\_id) FROM emps\_benefits AS eb JOIN emps AS e ON eb.emp\_id = e.id GROUP BY e;

**The average amount of benefits that employees of each department have (for instance, the employees of the human resources department have, on average, 3.5 benefits):**

SELECT AVG(SELECT COUNT(eb.benefit\_id) FROM emps\_benefits AS eb JOIN emps AS e ON eb.emp\_id = e.id GROUP BY e);