4.1) Вывести список сотрудников, которые получают заработную плату ниже, чем у непосредственного руководителя.

```
SELECT a.name FROM employee a, employee b
WHERE a.salary < b.salary AND b.id = a.chief_id;
```

4.2) Вывести список сотрудников, которые получают в отделе минимальную заработную плату в своем отделе.

```
SELECT name FROM employee
WHERE (department_id, salary) IN
(SELECT department_id, MIN (salary) FROM employee
GROUP BY department_id)
ORDER BY department_id;
```

4.3) Вывести список ID отделов, количество сотрудников в которых не превышает трех человек.

```
SELECT department_id, COUNT (id) AS quantity FROM employee
GROUP BY department_id
HAVING COUNT (id) <= 3;
```

4.4) Вывести список сотрудников, не имеющих назначенного руководителя, который работал бы в том же отделе.

```
SELECT DISTINCT a.id, a.name FROM employee a, employee b
WHERE (a.chief_id = b.id AND a.department_id != b.department_id) OR a.chief_id IS NULL
ORDER BY a.id;
```

4.5) Найти список ID отделов с максимальной суммарной заработной платой сотрудников.

```
CREATE TABLE Result_45 AS
SELECT department_id, SUM (salary) FROM employee
GROUP BY department_id
HAVING SUM (salary) = (
        SELECT MAX (sum_salary) FROM (
                SELECT SUM (salary) AS sum_salary FROM employee GROUP BY department_id)
        AS rich_bastards
);
```

4.6) Составить SQL-запрос, вычисляющий сумму всех значений всех ЗП в конкретном столбце таблицы.

```
SELECT SUM (salary) AS staff_salaries FROM employee
--Надеюсь, я правильно понял это задание.
```

5) Реализовать хранимую процедуру (update_salary_for_department):

```
CREATE OR REPLACE FUNCTION update_salary_for_department (department int, percent real)
--CREATE PROCEDURE не работает в версии 9.6
RETURNS SETOF employee AS $$
BEGIN
 ALTER TABLE employee
 ADD COLUMN IF NOT EXISTS old_salary int;

 UPDATE employee
 SET old_salary = salary;

 UPDATE employee
 SET salary = salary + salary * percent / 100
 WHERE department_id = department
 AND id != ALL (
          SELECT DISTINCT a.id FROM employee a, employee b
          WHERE (a.chief_id = b.id AND a.department_id != b.department_id) OR a.chief_id IS NULL);

 IF
   (SELECT salary FROM employee
   WHERE department_id = department
   AND id = ANY (
            SELECT DISTINCT a.id FROM employee a, employee b
            WHERE (a.chief_id = b.id AND a.department_id != b.department_id) OR a.chief_id IS NULL))
   < (SELECT MAX (salary) FROM employee WHERE department_id = department)
 THEN
   UPDATE employee
   SET salary = (SELECT MAX (salary) FROM employee WHERE department_id = department)
   WHERE department_id = department
   AND id = ANY (
            SELECT DISTINCT a.id FROM employee a, employee b
            WHERE (a.chief_id = b.id AND a.department_id != b.department_id) OR a.chief_id IS NULL);
 ELSE
   UPDATE employee
   SET salary = salary;
 END IF;

END;
$$ LANGUAGE plpgsql;
```