1. Написать запрос, который выводит номера всех проектов и их названия, на которых работает более 5-ти работников;

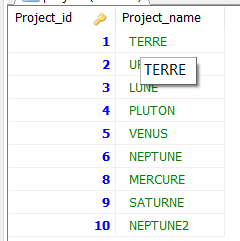
**SELECT** Project\_id, Project\_name **FROM** project

**JOIN** works\_on

**ON** project.Project\_id=works\_on.Project\_id\_FK

**GROUP** **BY** Project\_id

**having** **count**(works\_on.Employee\_id\_FK) >5;

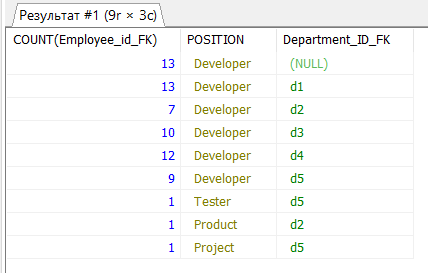


1. Написать запрос, который выводит количество сотрудников для каждой роли в каждом департаменте.

**SELECT** **COUNT**(Employee\_id\_FK),**POSITION**,Department\_ID\_FK **FROM** works\_on

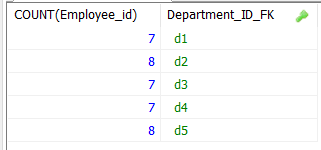
**JOIN** employees **ON** works\_on.Employee\_id\_FK=employees.Employee\_id

**GROUP** **BY** **POSITION**,Department\_ID\_FK



1. Написать запрос, который выводит кол-во работников для каждого из департаментов. Если работник не имеет департамента, то учитывать его не нужно;

**SELECT** **COUNT**(Employee\_id),Department\_ID\_FK **FROM** employees **WHERE** Department\_ID\_FK **is** **NOT** **NULL** **GROUP** **BY** Department\_ID\_FK;

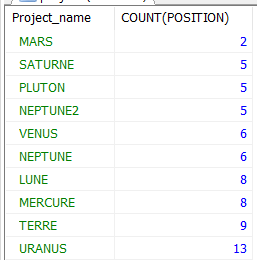


1. Написать запрос, который выводит кол-во различных ролей для каждого из проектов (должны выводиться названия проектов). Результаты должны быть отсортированы по возрастанию количества различных ролей;

**SELECT** Project\_name, **COUNT**(**POSITION**) **FROM** project

**JOIN** works\_on **ON** project.Project\_id=works\_on.Project\_id\_FK

**GROUP** **BY** Project\_name **ORDER** **BY** **COUNT**(**POSITION**);



1. Написать запрос, который выводит работников, которые работают в городах, которые начинаются на букву “М”;

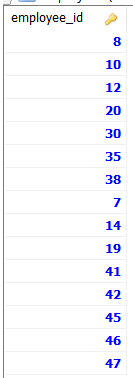
**SELECT** employee\_id **FROM** employees

**JOIN** department **ON** employees.Department\_id\_FK=department.Department\_id

**WHERE** location **LIKE** 'M%'

Или так

**SELECT** employee\_id **FROM** employees **WHERE** Department\_id\_FK **IN** (**SELECT** Department\_id **FROM** department **WHERE** location **LIKE** 'M%')



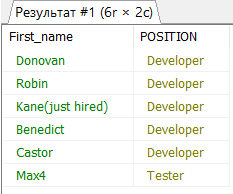
1. Написать запрос, который выводит имена и роли всех сотрудников, которые работают над проектом с максимальным бюджетом;

**SELECT** First\_name, **POSITION** **FROM** employees

**JOIN** works\_on **ON** employees.Employee\_id=works\_on.Employee\_id\_FK

**JOIN** project **ON** works\_on.Project\_id\_FK=project.Project\_id

**WHERE** Budget=(**SELECT** **MAX**(Budget) **FROM** project);



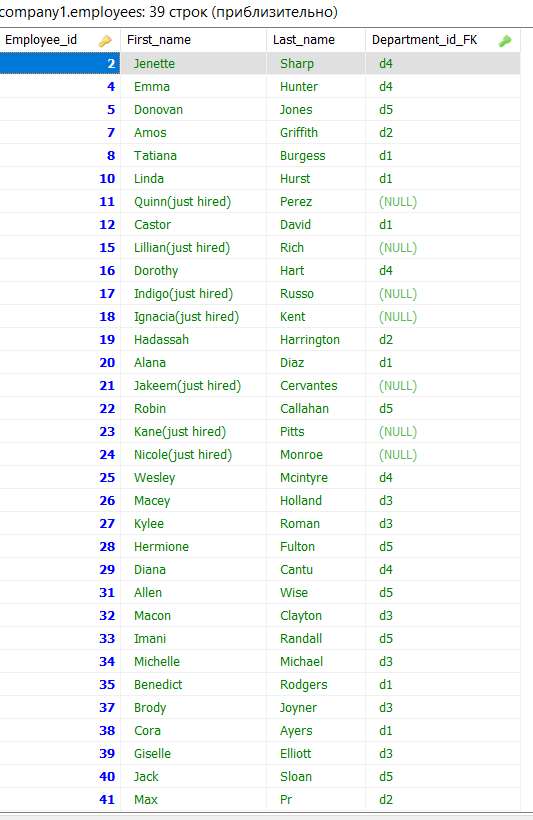
1. Удалить всех работников, которые работают над проектом с минимальным бюджетом;

**DELETE** employees **FROM** employees

**JOIN** works\_on **ON** employees.Employee\_id=works\_on.Employee\_id\_FK

**JOIN** project **ON** works\_on.Project\_id\_FK=project.Project\_id

**WHERE** Budget=(**SELECT** **Min**(Budget) **FROM** project);

Пользователи с d6(минимальная стоимость) - удалились

1. Написать запрос, который выводит информацию о всех работниках и расположении их отделов (тех, кто ещё не определён в какой-либо отдел, тоже выводит) для проекта с максимальным бюджетом;

**SELECT** Employee\_id, First\_name, Department\_Name **FROM** employees

**LEFT** **JOIN** department **ON** employees.Department\_id\_FK=department.Department\_id

**JOIN** works\_on **ON** employees.Employee\_id=works\_on.Employee\_id\_FK

**JOIN** project **ON** works\_on.Project\_id\_FK=project.Project\_id

**WHERE** project.Budget=(**SELECT** **MAX**(Budget) **FROM** project) **OR**(employees.Department\_id\_FK **IS** **NULL** **AND** project.Budget=(**SELECT** **MAX**(Budget) **FROM** project));



1. Создать таблицу employees\_2 (все столбцы такие же, как и в employees). Добавить в таблицу employees\_2 работников из таблицы employees с ID <= 5. Добавить в employees\_2 ещё несколько новых сотрудников.

**USE** company1;

**CREATE** **table** employees\_2

(

Employee\_id **int** **AUTO\_INCREMENT**,

**PRIMARY** **KEY** (Employee\_id),

First\_name **CHAR**(10) **NOT** **NULL**,

Last\_name **CHAR**(10) **NOT** **NULL**,

Department\_id\_FK **CHAR**(10)

);

**INSERT** **INTO** employees\_2 (Employee\_id, First\_name, Last\_name, Department\_id\_FK) **SELECT** Employee\_id, First\_name, Last\_name, Department\_id\_FK **FROM** employees **WHERE** employees.Employee\_id <=5;

**INSERT** **INTO** employees\_2 (First\_name, Last\_name, Department\_id\_FK) **VALUES** ('IVAN', 'Ivanovich', 'd5');



1. Написать запрос, который возвращает таблицу с сотрудниками имеющими ID с 1 по 7 из employees и c 1 по 6 из employees\_2. Дубликаты должны быть показаны.

**SELECT** \* **FROM** employees **WHERE** (employees.Employee\_id **BETWEEN** 1 **AND** 7)

**UNION** **ALL**

**SELECT** \* **FROM** employees\_2 **WHERE** (employees\_2.Employee\_id **BETWEEN** 1 **AND** 6);

