**Задание на семинар**

**Даны таблицы:**

**departments:**

* id serial primary key;
* name varchar(50) – название отдела;
* total\_salary integer – общая сумма зарплат всех сотрудников отдела.

**employees:**

* id serial primary key;
* name varchar(50) – имя сотрудника;
* department\_id integer – id отдела, в котором работает сотрудник;
* grade varchar(50) – грейд сотрудника;
* salary integer – зарплата;
* email varchar(50) – email-адрес сотрудника;
* boss\_id integer – id начальника.

**Задания:**

1. Сделать так, чтобы при добавлении нового сотрудника или обновлении зарплаты сотрудника или удалении сотрудника, автоматически изменялась сумма зарплат всех сотрудников отдела.
2. Запретить добавление нового сотрудника в отдел, если общая сумма зарплат всех сотрудников отдела превышает 500.  
   Подсказка: нужно использовать исключение.
3. Написать функцию для перевода сотрудника из одного отдела в другой и залогировать это действие.  
   Сигнатура функции: transfer\_employee( employee\_id INT, new\_department\_id INT)  
   Таблица для логов:  
   CREATE TABLE department\_transfer\_log (

id SERIAL PRIMARY KEY,

employee\_id INT,

old\_department\_id INT,

new\_department\_id INT,

transfer\_date TIMESTAMP DEFAULT NOW() );

1. Оптимизировать запрос для поиска сотрудника по имени и отделу.  
   Пример запроса:  
   SELECT \* FROM employees

WHERE name LIKE 'A%' AND department\_id = 5;

1. Написать функцию, которая меняет домен для почты сотрудника.  
   Сигнатура функции: change\_domain( employee\_id INT, new\_domain varchar(50))  
   Пример вызова функции: change\_domain( 2, ‘@mirea.ru’)