**Задание 2 (Верстальщик)**

Сверстать дашборд как на макете.

Есть таблица, в которой есть числа. Числа сравниваются с каким то интервалом в прошлом и если значение хуже то показываем индикатором падение (красная стрелка вниз), если лучше то положительный рост (зеленая стрелка вверх)

Для отображения график можно использовать сторонний компонент, например  
<http://www.highcharts.com/demo/> или любой другой.

Для таблицы можно использовать сторонний компонент, или сверстать статичную таблицу, на усмотрение кандидата.

Нужно в html+js (можно использовать фреймворки) сверстать. Данные для графика нужно хранить в json-файле, или в js-файле или в переменной, как будет удобнее. Бэкенд вообще не нужен.

Ссылка на макет: https://www.figma.com/file/EhTdNykEscl7EP3aaljpWz/Frontend?node-id=0%3A1&t=imzrldqW8Fdv8DNE-0

**Задание 3 (На отчеты)**

В любом онлайн редакторе SQL например (<https://sqliteonline.com/>) выполнить скрипт

--DROP TABLE Department;

--DROP TABLE Employee;

CREATE TABLE Department (

ID NUMBER NOT NULL PRIMARY KEY,

NAME VARCHAR(255)

);

INSERT INTO Department (ID, NAME) VALUES (1, "Dep1");

INSERT INTO Department (ID, NAME) VALUES (2, "Dep2");

INSERT INTO Department (ID, NAME) VALUES (3, "Dep3");

INSERT INTO Department (ID, NAME) VALUES (4, "Dep4");

CREATE TABLE Employee (

ID NUMBER NOT NULL PRIMARY KEY,

DEPARTMENT\_ID NUMBER,

CHIEF\_ID NUMBER,

NAME VARCHAR(255),

SALARY INTEGER,

FOREIGN KEY (DEPARTMENT\_ID) REFERENCES Department(ID)

FOREIGN KEY (CHIEF\_ID) REFERENCES Employee(ID)

);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (1, 1, null, "Bob", 1000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (2, 1, 1, "Bob Jr", 100);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (3, 1, 1, "Bob Jr 1", 3000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (4, 1, 1, "Bob Jr 2", 50);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (10, 2, null, "Robert", 10000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (11, 2, 10, "Sean", 1000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (12, 2, 10, "Olga", 8000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (13, 2, 10, "Nataly", 15000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (14, 2, 10, "Stan", 4000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (22, 3, null, "Maria", 100000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (23, 3, 22, "Bob Jr 2", 20000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (33, 4, null, "Rodrigo", 60000);

INSERT INTO Employee (ID, department\_id, chief\_id, name, salary)

VALUES (34, 4, null, "Dilan", 60000);

И написать запросы

1. Вывести список сотрудников, получающих заработную плату большую чем у непосредственного руководителя
2. Вывести список сотрудников, получающих максимальную заработную плату в своем отделе
3. Вывести список ID отделов, количество сотрудников в которых не превышает 3 человек
4. Вывести список сотрудников, не имеющих назначенного руководителя, работающего в том-же отделе
5. Найти список ID отделов с максимальной суммарной зарплатой сотрудников