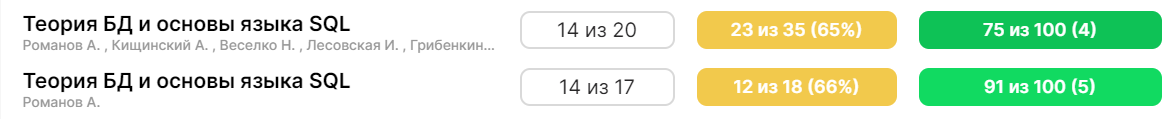
**Дополнительное задание на 15 баллов.**

*Баллы могут добавить в МОЮ ОЦЕНКУ ПО SQL за прошлый семестр!*



*Это позволит исправить мою оценку на 5!*

**ВОПРОСЫ:**

В этой задаче присутствуют месяцы. Как с ними быть?

Их нужно учитывать. Нужно сохранять информацию по месяцам.

Поле месяца сделаю в формате datetime. Это позволит сохранять информацию на протяжении многих лет. Назову его dt.

Возможно в части полей нет необходимости и их можно сделать расчётными?

Да. Аркадий сказал, что при проектировании БД часто встаёт вопрос о хранении расчётной информации.

Можно поле – счётчик, которое будет автоматически обновляться. Когда будет интересно узнать значение будет просто делаться select поля счётчика.

Можно не делать поле, а составить сложный select для расчёта.

Если значение счётчика будет запрашиваться регулярно, то лучше сделать его в качестве отдельного поля.

Если будет увольняться/добавляться сотрудник в магазин, то план магазина должен пересчитываться в + или -?

Нет. Это не учитываем. Т.к. это сильно усложняет задачу из-за того, что не понятно, как быть с продажами, которые сделал сотрудник.

План продаж для магазина будет определяться в начале месяца по имеющимся данным.

В каждом *магазине* работают *продавцы*. У каждого из них есть план продаж на месяц. (Он у всех одинаковый).

У каждого магазина также есть план продаж на месяц.

Осуществление возрастов - затруднительно.

Т.к., возможно, ситуация, когда план продаж был выполнен сотрудником / магазином, а потом происходит возврат и нужно пересчитать все ЗП снова.

Чтобы таким не заниматься, нужно подводить итоги по достижению целей в конце месяца!

С другой стороны, может быть директор, который хочет знать, как идут дела в любой момент времени.

-- Поэтому у меня возвраты товаров запрещены.

**Разбор ТЗ:**

За каждую продажу продавец получает 200 баллов.

План продавца = 100 продаж в месяц. (Необходимо отслеживать, как каждый отдельно взятый продавец её выполняет! Здесь будет логичным реализовать счётчик продаж).

План магазина = количество продаж каждого продавца \* на количество продавцов, работающих в этом магазине (в месяц). (Необходимо отслеживать, как каждый отдельно взятый магазин справляется с этим)

Выходит, что у всех продавцов в 1 магазине одинаковый план продаж.

Таким образом, баллы складываются в премиальную часть сотрудника *(score)* и в конце месяца премиальная часть складывается с его окладом (salary) и формируется его итоговая выплата (income).

Как именно формируется income читай ниже:

Если продавец достигает плана, то ему положена премия-продавца в размере 10% от количества его баллов.

Получается, что у каких-то продавцов может быть эта премия, а у каких-то – нет.

Income = fixed\_salary + score \* 0.1

Если весь магазин выполняет план, то каждому продавцу положена премия-магазина в размере 20% (20% от score этого сотрудника?) Да. Верно.

Income = fixed\_salary + score \* 0.2

(Суммируется ли премия от магазина с премией для отдельного продавца?)

Нет не суммируются.

Таким образом, у вас должны быть таблицы, которые показывают то,

* Сколько баллов получает продавец за конкретный месяц (score)
* Каков его оклад за конкретный месяц (salary)
* Какова премиальная часть с учетом дополнительной премии за выполнение плана (income) за конкретный месяц.

**Всю работу нужно сделать на триггерах. Перед этим проработайте структуру таблиц и алгоритм.**

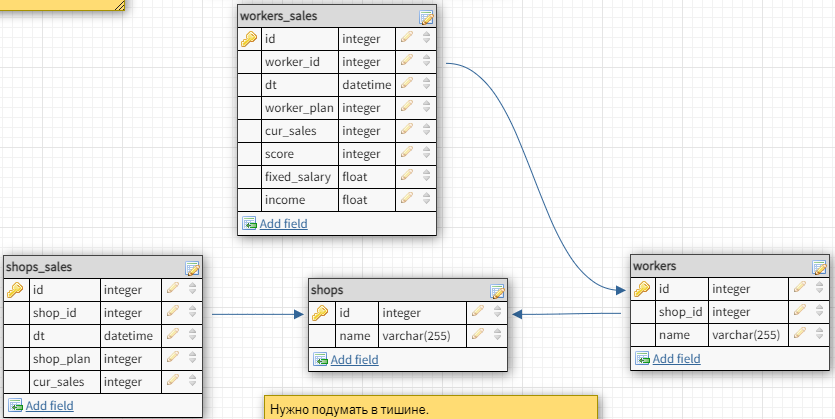
​это дополнительное задание на 15 баллов

дедлайн по допзаданию - до следующего понедельника.

Триггер должен работать на каждую продажу.

Каждый INSERT = работа триггера

**База данных, что я спроектировал**



**Дальнейшие шаги:**

1. Теперь я создам её в Open Server
2. Реализую триггеры

Где я их буду писать?

Триггеры вешаются на конкретные события в конкретных таблицах и выполняют заданные действия.

Я хочу повторить тему триггеров по презентации.

1. Заполню таблицы шаблонными данными
2. Протестирую триггеры. Т.к. триггеры должны срабатывать на доб. данные
3. Сдам работу