Аналитика таблиц в продукте.

Часто нам придется анализировать происходящий прогресс у наших пользователей, обрабатывать отчеты полученные нашими службами в разных форматах. Самый популярный вариант данных, понятный и нам, и стейк-холдерам - таблицы, базы данных, с использованием языка SQL - он со столбцами и строками, сущностями и связями, ключами и группировками позволить манипулировать данными



Пример задания

Задайте один или несколько запросов к таблице leads , используя групповые функции. Проверьте, правильно ли они работают.

Решение:

SELECT claims.

ROUND(SUM(price*amount), 2) AS Стоимость,

ROUND((SUM(price*amount)0.18)/(1+0.18), 2) AS НДС,

ROUND(SUM((price*amount)/(1.18)),2) AS Стоимость_без_НДС

FROM leads

GROUP BY claims;

Пример задания

Вывести информацию (горячие заявки, ФИО клиента и цену) о горячих заказах, цены которых меньше или равны средней цене горячих заказов . Информацию вывести в отсортированном по убыванию цены виде. Среднее вычислить как среднее по цене горячих заказов

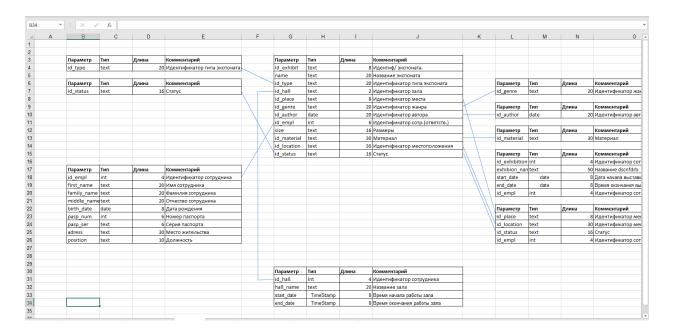
Решение:

SELECT name, claims, price AVG(price) AS "Средняя_цена" FROM leads
WHERE price <= AVG price

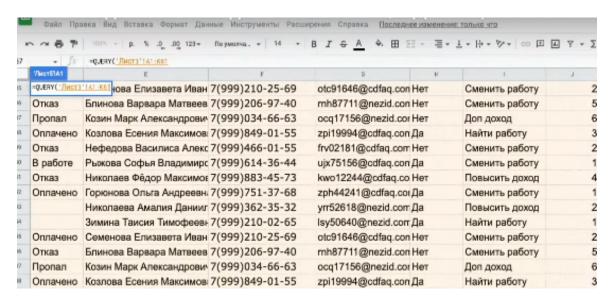
GROUP BY name, claims, price ORDER BY DESC

Порой мы должны выступить в роли архитекторов БД - так мы через таблицы, нормализуя их, сможем упростить задачу не только хранения, но и обработки данных. Мы группой решали задачу - оцифровка небольшого музея в городке N. Получилась реляционная БД, я делала связи и описывала хранилище.

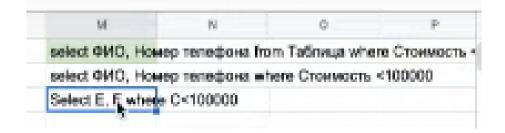
https://docs.google.com/spreadsheets/d/1fFzxspo5N6PPi_durRiCW6IUD0oP4h0f99D9atCOPOo/edit?usp=sharing



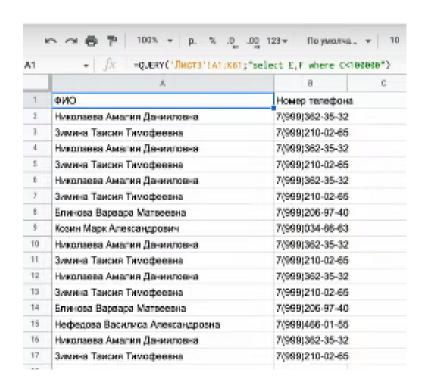
Есть способ, но не без недостатков, решения простых задач на SQL, встроенный в Googleтаблицы. Например, нам надо прозвонить всех пользователей, решивших купить продукт по цене, которая скоро вырастет:





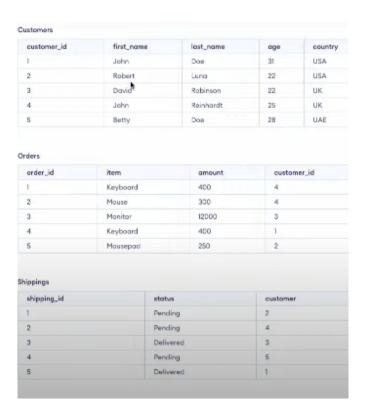


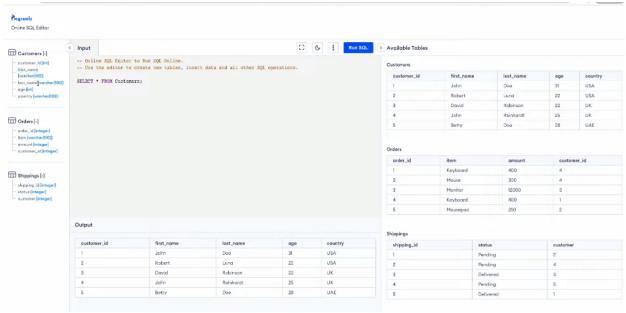
Имея такую таблицу прямо в строке формул формируем SQL-запрос через формулу =QUERY просто говоря командами по-ангшлийски. И вот результат:



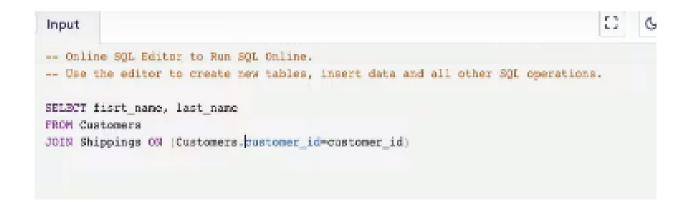


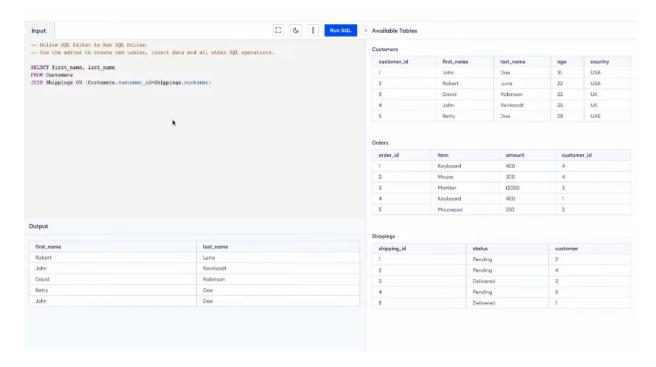
Ho Google-таблицы не позволяет нам склеивать данные из разных таблиц, а нам надо продемонстрировать решение следующей задачки - приходится вернуться к MySQL. У нас есть три таблицы - клиенты, заказы, доставка и структура БД.

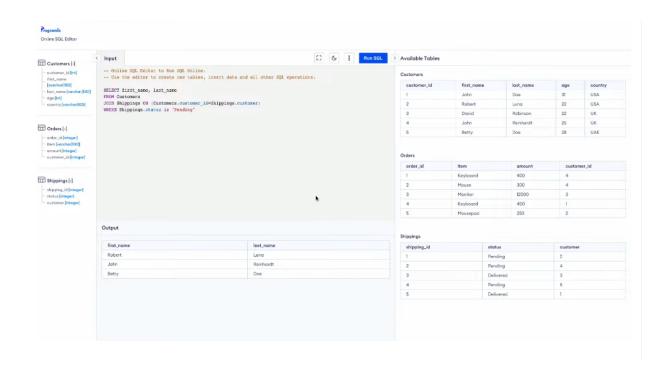




Объединив таблицы, мы решаем новую задачу: выявить клиентов, заказы которых находятся в статусе обработки, по ключу 'Customer_id'







Такова работа аналитика в продукте. Согласитесь, что это приятно делать!