Проект № 2

ЗАДАЧА

Проанализировать работу компании с точки зрения ее эффективности

Дать рекомендации по масштабированию бизнеса, а именно в каком штате лучше открыть офлайн-магазин

КОНКРЕТНЫЕ ШАГИ

1. Оценить динамику продаж и распределение выручки по товарам.
2. Составить портрет клиента, а для этого — выяснить, какие клиенты приносят больше всего выручки.
3. Проконтролировать логистику компании (определить, все ли заказы доставляются в срок и в каком штате лучше открыть офлайн-магазин)

## **Шаг 1: Определяем эффективность продаж**

## 

## Какова динамика продаж?

* Какие категории и подкатегории продаж приносят больше всего выручки?
* Какие товары продаются лучше всего?

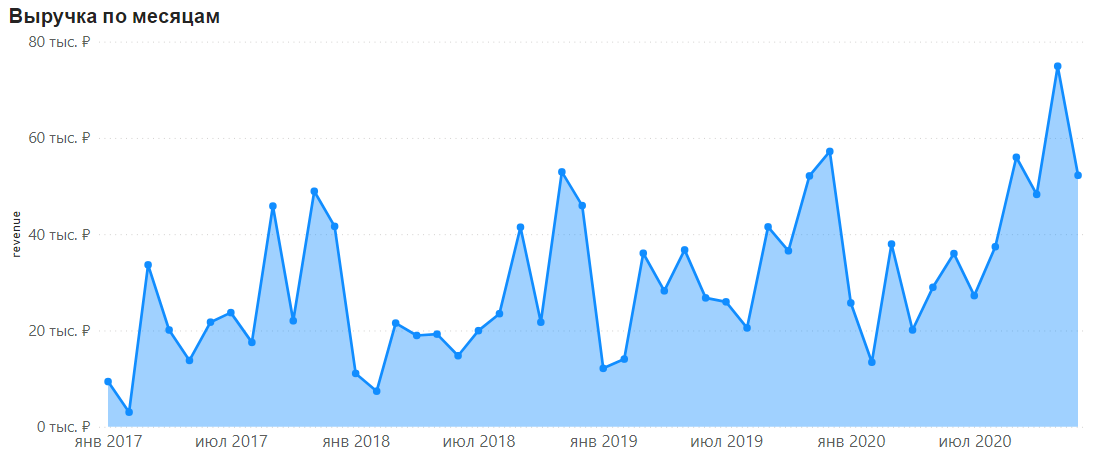
### **Считаем выручку**

* date (месяц заказа) ― тип date;
* revenue (объем выручки) ― округленные значения до целых с помощью round.

Запрос:

| *select*  *date\_trunc('month',sd.order\_date)::date date,*  *round(SUM(sp.price \* sc.quantity \* (1 - sc.discount))) revenue --Выручка = цена \* кол-во \* (1 - доля скидки)*  *from*  *sql.store\_products sp*  *join sql.store\_carts sc on sc.product\_id = sp.product\_id*  *join sql.store\_delivery sd on sd.order\_id = sc.order\_id*  *group by 1*  *order by 1* |
| --- |

**Динамика выручки по месяцам:**



**Вывод:**

По диаграмме отчетливо видно, что наблюдается сезонность

### **Считаем выручку по категориям**

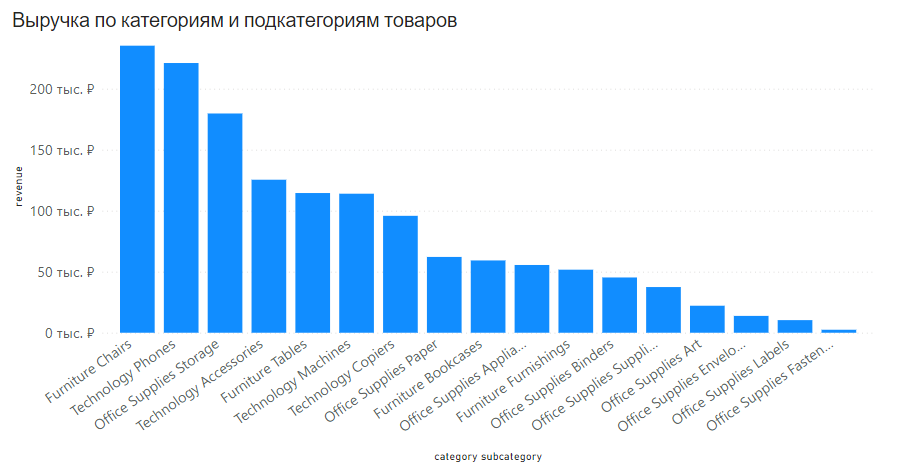
Посмотрим, что клиенты покупают больше всего.

Запрос, который выведет сумму выручки по различным категориям и подкатегориям:

* category (категория товара);
* subcategory (подкатегория товара);
* revenue (объем выручки) ― округлите до целых с помощью round.

Запрос:

| select sp.category, sp.subcategory, round(SUM(sp.price \* sc.quantity \* (1 - sc.discount))) revenue  from  sql.store\_products sp  join sql.store\_carts sc on sc.product\_id = sp.product\_id  group by 1,2  order by 3 desc |
| --- |



### **Смотрим на товары**

Важно не только понять, какие из товаров приносят больше всего выручки, но и определить, какую долю от общей выручки составляют их продажи.

Запрос, который выведет данные топ-25 товаров по объёму выручки в следующем формате:

* product\_nm — наименование товара;
* revenue — объем выручки;
* quantity — количество проданных товаров;
* percent\_from\_total — доля от общей выручки в процентах.

Запрос:

| select  **sp.product\_nm**,  round(SUM(sp.price \* sc.quantity \* (1 - sc.discount)),2) **revenue**,  SUM(quantity) as **quantity**,  round(SUM(sp.price \* sc.quantity \* (1 - sc.discount)) /  (  select SUM(sp.price \* sc.quantity \* (1 - sc.discount)) total\_sum  from  sql.store\_products sp join sql.store\_carts sc on sc.product\_id = sp.product\_id  )\*100,2  ) as **percent\_from\_total**  from  sql.store\_products sp join sql.store\_carts sc on sc.product\_id = sp.product\_id  group by 1  order by 2 desc  limit 25 |
| --- |

Комментарий к запросу:

(

select SUM(sp.price \* sc.quantity \* (1 - sc.discount)) total\_sum

from

sql.store\_products sp join sql.store\_carts sc on sc.product\_id = sp.product\_id

)

*Этот запрос вычисляет общую сумму выручки. Он используется при вычислении доли в списке столбцов*

## 

## **Шаг 2: Составляем портрет клиента**

Ответим на следующие вопросы:

* Как соотносится количество клиентов по категориям B2B и B2C?
* Как соотносится выручка по этим категориям?
* Каких клиентов больше, новых или старых?

Рассчитаем основные показатели по целевым клиентам:

* количество заказов
* средняя сумма заказов
* количество различных адресов доставки

### 

### **Считаем выручку клиентов по категориям**

Посмотрим, сколько у компании B2B и B2C-клиентов и какую долю от общей выручки они приносят.

Запрос, который выведет количество клиентов и выручку по категориям клиента:

* category (категория клиента);
* cust\_cnt (количество клиентов);
* revenue (объём выручки) ― округлите до целых с помощью round.

Запрос:

| select  scu.category,  count(distinct scu.cust\_id) cust\_cnt,  round(SUM(sp.price \* sc.quantity \* (1 - sc.discount))) revenue  from  sql.store\_customers scu  join sql.store\_delivery sd on scu.cust\_id = sd.cust\_id  join sql.store\_carts sc on sc.order\_id = sd.order\_id  join sql.store\_products sp on sp.product\_id = sc.product\_id  group by scu.category  order by 3 desc |
| --- |

Комментарий:  
*Необходимо указать ключевое слово distinct при подсчете количества клиентов*

**Вывод:**



Можно заметить, что B2B-клиентов намного больше, и выручки они приносят тоже в разы больше!

### **Составляем характеристику B2B-клиентов**

Ответим на следующие вопросы:

* Какова динамика новых B2B-клиентов по месяцам?
* С чем связан рост выручки: за счет увеличения продаж по старым клиентам или же за счёт привлечения новых клиентов?

Запрос, который выведет количество новых **корпоративных** клиентов по месяцам:

* month (месяц) ― тип date;
* new\_custs (количество новых клиентов).

Запрос:

| with cust\_mindate as --CTE для подсчёта дату первой покупки каждого клиента  (  select scu.cust\_id, MIN(date\_trunc('month',sd.order\_date)::date) min\_date  from  sql.store\_customers scu join sql.store\_delivery sd on scu.cust\_id = sd.cust\_id  where scu.category = 'Corporate'  group by 1  )  select min\_date as month, count(cust\_id) new\_custs -- Запрос подсчёта количества новых клиентов в каждый месяц  from cust\_mindate  group by min\_date  order by 1 |
| --- |

По полученным данным построим график динамики новых корпоративных клиентов по месяцам



**Вывод:**

Как видно из графика, в самом начале было привлечено много клиентов, а после 2018 года новые клиенты почти не привлекались.

### **Составляем характеристику B2B-клиентов**

Изучим основные показатели по корпоративным клиентам:

Ответим на следующие вопросы:

* Какова сумма заказа?
* Сколько товаров в заказе и сколько различных офисов?
* Куда они доставлялись?

Найдём ответы на следующие вопросы.

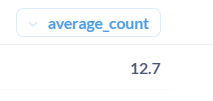
*Все ответы округлены до одного знака после разделителя*

**Сколько в среднем различных товаров в заказах у корпоративных клиентов?**

| with cust\_countproduct as  (  select scu.cust\_id, COUNT(sp.product\_id) as countproduct  from  sql.store\_customers scu join sql.store\_delivery sd on scu.cust\_id = sd.cust\_id  join sql.store\_carts sc on sd.order\_id = sc.order\_id  join sql.store\_products sp on sp.product\_id = sc.product\_id  where scu.category = 'Corporate'  group by 1  )  select round(avg(countproduct),1) as average\_count  from  cust\_countproduct |
| --- |

*В комментариях не нуждается*

*Результат:*

**

**Какая в среднем сумма заказов у корпоративных клиентов?**

Запрос:

| with cust\_revenue as  (  select scu.cust\_id, SUM(sc.quantity \* sp.price) as sum\_revenue  from  sql.store\_customers scu join sql.store\_delivery sd on scu.cust\_id = sd.cust\_id  join sql.store\_carts sc on sd.order\_id = sc.order\_id  join sql.store\_products sp on sp.product\_id = sc.product\_id  where scu.category = 'Corporate'  group by 1  )  select round(avg(sum\_revenue), 1) as average\_sum  from cust\_revenue |
| --- |

*В комментариях не нуждается*

*Результат:*

**

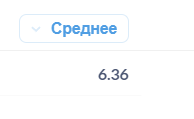
**Сколько в среднем различных офисов у корпоративных клиентов?**

Запрос:

| with cust\_zipcode as  (  select scu.cust\_id, COUNT(sd.zip\_code) as sum\_zipcode  from  sql.store\_customers scu join sql.store\_delivery sd on scu.cust\_id = sd.cust\_id  where scu.category = 'Corporate'  group by 1  )  select avg(sum\_zipcode)  from  cust\_zipcode |
| --- |

*В комментариях не нуждается*

*Результат:*

**

## **Шаг 3: Анализируем логистику компании**

Оценим текущую картину по логистике поставок и найдём штат, в котором лучше всего открыть офлайн-магазин.

Сформулируем вопросы, на которые нам предстоит найти ответы в рамках задачи:

* Насколько эффективно выполняются текущие доставки?
* Как распределяются доставки и выручка по штатам и городам? (Ответ на этот вопрос поможет определить наиболее удачное место для офлайн-магазина.)

### **Определяем эффективность доставки**

Определим, какая доля заказов выполняется в срок по каждой категории.

Запрос, который выведет:

* тип доставки;
* общее количество заказов (orders\_cnt);
* количество заказов, которые не были доставлены вовремя (late\_orders\_cnt);
* долю выполненных вовремя заказов, в процентах (% success), округленную до двух знаков после запятой.

Запрос:

| with is\_ontime as --CTE показывает, вовремя или нет был доставлен каждый заказ каждого типа  (  select sd.ship\_mode,  case  when sd.ship\_mode = 'Standard Class' and (sd.ship\_date - sd.order\_date <= 6) then 'yes'  when sd.ship\_mode = 'Second Class' and (sd.ship\_date - sd.order\_date <= 4) then 'yes'  when sd.ship\_mode = 'First Class' and (sd.ship\_date - sd.order\_date <= 3) then 'yes'  when sd.ship\_mode = 'Same Day' and (sd.ship\_date - sd.order\_date <= 0) then 'yes'  else 'no'  end as status  from sql.store\_delivery sd  ),  table\_success as --CTE показывает количество вовремя доставленных заказов каждого типа  (  select ship\_mode, COUNT(status) count\_success  from is\_ontime  where status = 'yes'  group by 1  ),  table\_notsuccess as --CTE показывает количество заказов каждого типа, доставленных с опозданием  (  select ship\_mode, COUNT(status) count\_notsuccess  from is\_ontime  where status = 'no'  group by 1  ),  count\_orders as --CTE показывает количество заказов каждого типа  (  select sd.ship\_mode, COUNT(sd.order\_id) orders\_cnt  from sql.store\_delivery sd  group by 1  )  select co.ship\_mode, orders\_cnt, count\_notsuccess late\_orders\_cnt, round(count\_success::numeric/orders\_cnt \* 100,2) as "% success"  from count\_orders co  join table\_success ts on ts.ship\_mode = co.ship\_mode  join table\_notsuccess tn on tn.ship\_mode = ts.ship\_mode  order by 4 |
| --- |

Определим долю заказов, отправленных вторым классом, которые были доставлены с опозданием, по кварталам и построим график

Запрос:

| with is\_ontime\_2ndclass as -- CTE выводит статус, вовремя или нет была осуществлена доставка вторым классом по кварталам  (  select sd.ship\_mode, date\_trunc('quarter', sd.ship\_date)::date as "quarter\_date",  case  when sd.ship\_mode = 'Second Class' and (sd.ship\_date - sd.order\_date <= 4) then 'yes'  else 'no'  end as status  from sql.store\_delivery sd  where sd.ship\_mode = 'Second Class'  ),  table\_notsuccess\_2ndclass as -- CTE: количество не вовремя доставленных заказов второго класса по кварталам  (  select ship\_mode, quarter\_date, COUNT(status) count\_notsuccess\_2ndclass  from is\_ontime\_2ndclass  where status = 'no'  group by 1,2  order by 2  ),  count\_orders\_2ndclass as -- Количество заказов второго класса  (  select sd.ship\_mode, COUNT(sd.order\_id) orders\_cnt\_2ndclass  from sql.store\_delivery sd  where sd.ship\_mode = 'Second Class'  group by 1  )  select quarter\_date, round(count\_notsuccess\_2ndclass::numeric/ orders\_cnt\_2ndclass \* 100, 2) as "% not\_success"  from table\_notsuccess\_2ndclass tn join count\_orders\_2ndclass co on tn.ship\_mode = co.ship\_mode |
| --- |



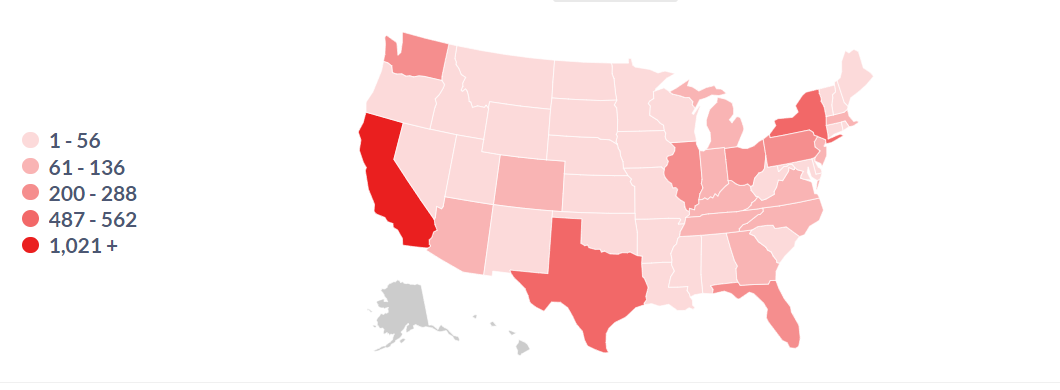
Вывод:  
  
Заказы доставляются с опозданием в течение ограниченного периода времени.

**Выбираем оффлайн точку продаж**

С помощью оффлайн-магазина можно привлечь больше клиентов и снизить стоимость доставки, нужно только выбрать, где его открыть.  
  
Напишем запрос, который выведет количество доставок по штатам.

Запрос:

| select state, COUNT(sd.order\_id)  from  sql.store\_delivery sd  group by 1  order by 2 desc |
| --- |



Вывод:

Исходя из данной карты распределения количества доставок по штатам можно сделать вывод, что штат Калифорния является наиболее подходящим для размещения