

## Шаг 1: Определяем эффективность продаж

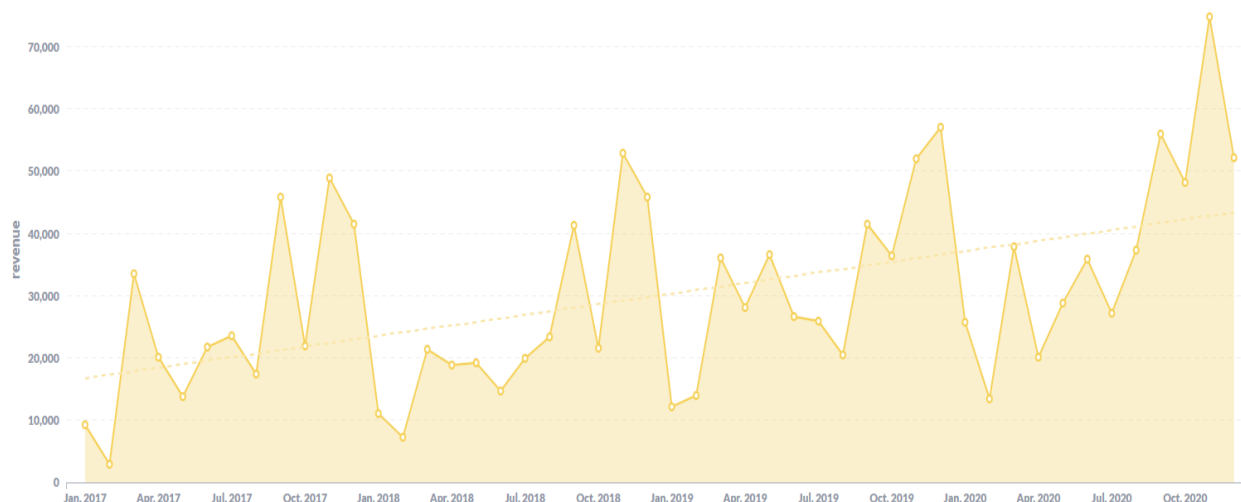
Для этого нужно ответить на следующие вопросы:

- Какова динамика продаж?
- Какие категории и подкатегории продаж приносят больше всего выручки?
- Какие товары продаются лучше всего?

### Считаем выручку

*Сумма выручки по месяцам:*

```
select
    date_trunc('month', sd.order_date)::date as date, --вывел дату доставки, округлил до
    месяца, конвертировал в дату
    round(sum((sp.price-sp.price*sc.discount)*sc.quantity)) revenue --вывел сумму выручки,
    округлите значения до целых
from sql.store_carts sc
    join sql.store_products sp on sc.product_id = sp.product_id --соединил таблицы по id
    товара
    join sql.store_delivery sd on sd.order_id = sc.order_id --соединил таблицы по id заказа
group by date -- сгруппировал по дате заказа
order by date -- отсортировал по дате заказа
```



**Вывод:** в целом динамика выручки растёт, по графику наблюдается сезонность роста и падения выручки. Пики продаж приходятся на сентябрь, ноябрь и декабрь.

## Считаем выручку по категориям

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

```
SELECT
sp.category,
sp.subcategory,
ROUND (SUM (sc.quantity * sp.price * (1-sc.discount))) AS revenue --вывел сумму
выручки с округлением до целых
FROM sql.store_carts as sc
JOIN sql.store_products as sp ON sc.product_id = sp.product_id --соединил таблицы по id
товара
GROUP BY sp.category, sp.subcategory --сгруппировал по категория и подкатегориям
ORDER BY revenue desc --отсортировал по убыванию выручки
```

**Вывод:** на основе результатов в самом топе по продажам оказались стулья.

## Смотрим на товары

*Запрос, который выведет данные топ-25 товаров по объёму выручки:*

```
with total_amount as --использовал подзапрос CTE
(select
sum(c.quantity*p.price*(1-c.discount)) as total_revenue --вывел сумму выручки
from sql.store_products p
join sql.store_carts c on c.product_id=p.product_id)
select
p.product_nm,
round(sum(c.quantity*p.price*(1-c.discount)),2) revenue, -- вывел объём выручки, округлил
до двух знаков после разделителя
sum(c.quantity) quantity, -- вывел количество проданных товаров
round(sum(c.quantity*p.price*(1-c.discount)/ta.total_revenue)*100,2) percent_from_total
--вывел долю от общей выручки в процентах, округлил до двух знаков после
разделителя
from sql.store_products p
join sql.store_carts c on p.product_id=c.product_id --соединил таблицы по id товара
cross join total_amount ta --соединил CTE таблицу
group by 1 --сгруппировал по описанию продукта
order by 2 desc --отсортировал по убыванию объёма выручки товара
limit 25 --оставил топ-25 товаров
```

**Вывод:** наибольший объём выручки у товара Canon imageCLASS 2200 Advanced Copier, она составила 2.56% доли от общей выручки.

## Шаг 2: Составляем портрет клиента

Вопросы, которые помогут получить больше информации о клиентах компании:

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

## Считаем выручку клиентов по категориям

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

```
SELECT
    scu.category,
    count(distinct sd.cust_id) as cust_cnt, --вывел количество клиентов
    round(SUM(quantity*price*(1-discount))) AS revenue --вывел объём выручки с
округлением до целых
FROM sql.store_delivery sd
join sql.store_customers scu on scu.cust_id = sd.cust_id
join sql.store_carts sca on sca.order_id = sd.order_id
join sql.store_products sp on sp.product_id = sca.product_id
group by 1 --сгруппировал по типу клиентов
order by 3 desc --отсортировал по убыванию выручки
```

**Вывод:** корпоративных клиентов больше их количество составило 645, что составляет 81,3% от общего количества, объём их выручки также больше, он составил 1172009 долларов и 81% от общего объёма.

## Составляем характеристику B2B-клиентов

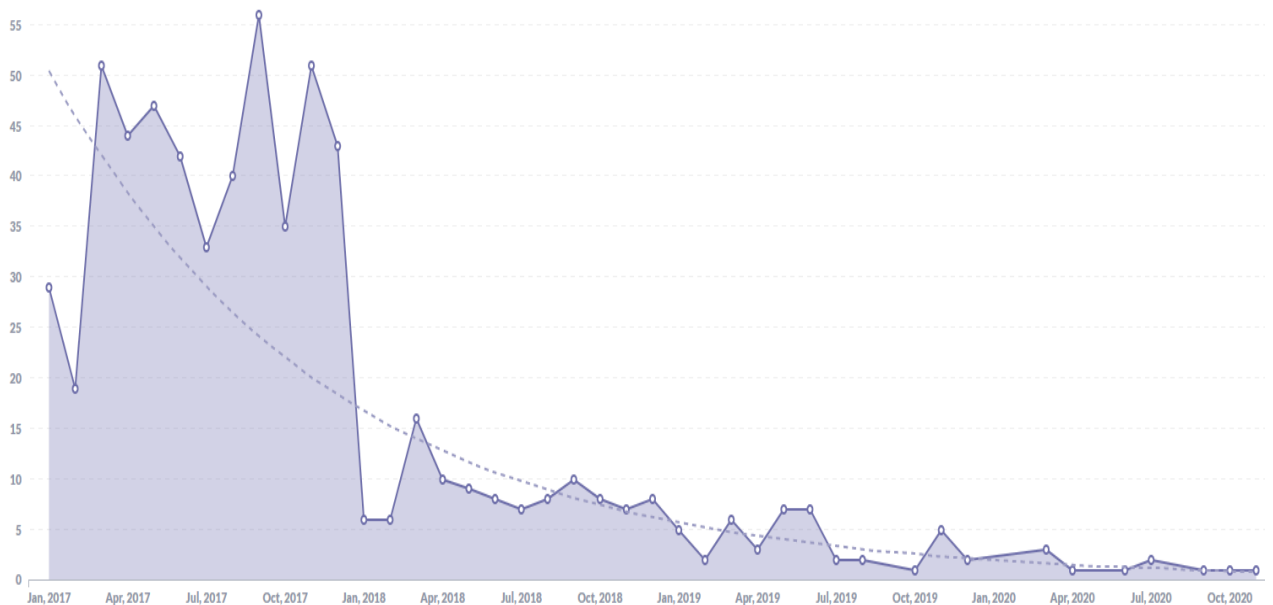
Запрос, который выведет количество новых корпоративных клиентов по месяцам:

```
with table_1 as --использовал подзапрос CTE
(select
    min (date_trunc ('month', sd.order_date))::date as one_month, -- округлил до минимальных
дат месяца, конвертировал в дату
    cust.cust_id
from sql.store_customers cust
```

```

join sql.store_delivery sd on cust.cust_id = sd.cust_id --соединил таблицы по id клиента
where cust.category in ('Corporate') -- отфильтровал, оставив только корпоративных
клиентов
group by 2) --сгруппировал по месяцам
select
one_month,
count(distinct cust_id) as new_custs --вывел количество новых клиентов
from table_1
group by 1 --сгруппировал по месяцам
order by 1 asc --отсортировал по в порядке возрастания

```



**Вывод:** количество новых клиентов падает, по графику видно, в самом начале было хорошее привлечение клиентов, а после 2018 года новые клиенты почти не привлекались, нужно поработать на маркетинговой составляющей.

Отвечу на следующие вопросы:

1. Сколько в среднем различных товаров в заказах у корпоративных клиентов?
2. Какая в среднем сумма заказов у корпоративных клиентов?
3. Сколько в среднем различных офисов у корпоративных клиентов?

```

select
round(avg(prods_by_order),1) avg_products, --в среднем различных товаров в заказах у
корпоративных клиентов, с округлением до одного знаков после разделителя
round(avg(rev_by_order),1) avg_revenue, -- в среднем сумма заказов у корпоративных
клиентов, с округлением до одного знаков после разделителя
round(avg(offices_by_cust),1) avg_offices --в среднем различных офисов у
корпоративных клиентов, с округлением до одного знаков после разделителя
from
(select --использовал подзапрос
sc.order_id,
count (distinct sc.product_id) prods_by_order, -- количество различных товаров в заказах
у корпоративных клиентов
sum(price*quantity*(1-discount)) as rev_by_order --сумма заказов у корпоративных
клиентов
from sql.store_carts sc
join sql.store_delivery sd ON sc.order_id = sd.order_id
join sql.store_customers cu ON cu.cust_id = sd.cust_id
join sql.store_products sp on sp.product_id = sc.product_id
where cu.category = 'Corporate' --отфильтровал, оставив только корпоративных
клиентов
group by sc.order_id) products_by_order --сгруппировал, задал алиас
cross join --соединил таблицы
(select
cu.cust_id,
count(distinct zip_code) as offices_by_cust --количество различных офисов у
корпоративных клиентов
from sql.store_delivery sd
join sql.store_customers cu on sd.cust_id = cu.cust_id
where cu.category = 'Corporate' --отфильтровал, оставив только корпоративных
клиентов
group by cu.cust_id) by_cust --сгруппировал, задал алиас

```

**Вывод:** в среднем 2 различных товара в заказах у корпоративных клиентов, средняя сумма заказов у корпоративных клиентов 285.9 долларов, в среднем 6 различных офисов у корпоративных клиентов. Стоит поработать над маркетингом для увеличения количества товаров в заказе, например с помощью дополнительных скидок, либо акций.

## Шаг 3: Анализируем логистику компании

Вопросы, на которые мне предстоит найти ответы в рамках задачи:

- Насколько эффективно выполняются текущие доставки?
- Как распределяются доставки и выручка по штатам и городам?
- Какой штат является наиболее перспективным для открытия офлайн-магазина? Почему?

Напишу запрос, который выведет:

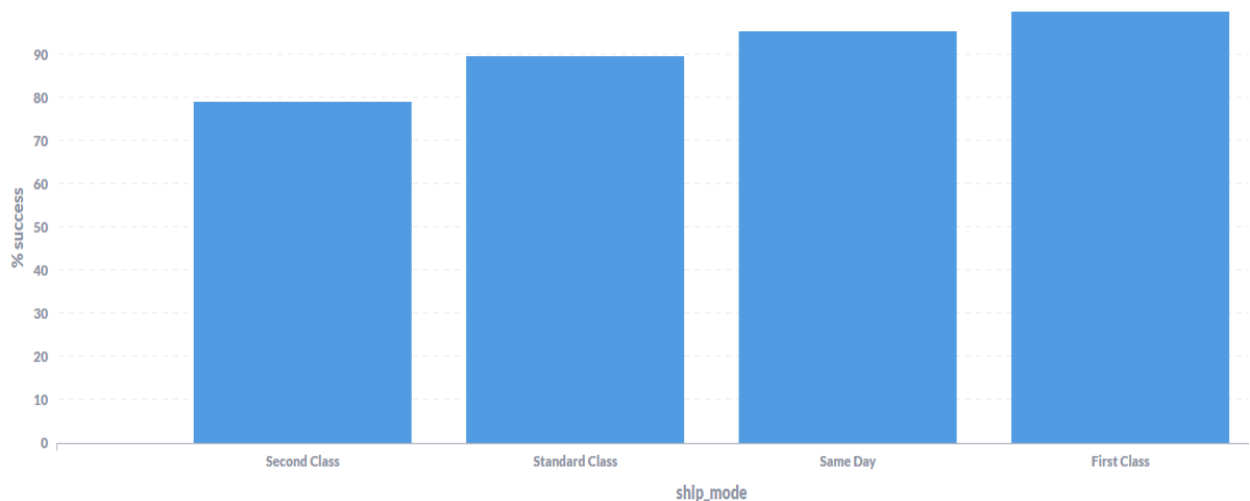
- тип доставки;
- общее количество заказов (**orders\_cnt**);
- количество заказов, которые не были доставлены вовремя (**late\_orders\_cnt**);
- долю выполненных вовремя заказов, в процентах (**% success**), округлённую до двух знаков после запятой.

```
WITH table1 as --использовал подзапрос CTE
(SELECT
ship_mode,
COUNT (order_id) as orders_cnt, --вывел общее количество заказов
COUNT (CASE WHEN ship_mode = 'Standard Class' AND (ship_date - order_date) > 6
THEN order_id
WHEN ship_mode = 'Second Class' AND (ship_date - order_date) > 4 THEN order_id
WHEN ship_mode = 'First Class' AND (ship_date - order_date) > 3 THEN order_id
WHEN ship_mode = 'Same Day' AND (ship_date - order_date) > 0 THEN order_id
END) as late_orders_cnt --с помощью оператор CASE вывел количество заказов,
которые не были доставлены вовремя
FROM sql.store_delivery
GROUP BY ship_mode)
SELECT
ship_mode,
orders_cnt,
late_orders_cnt,
```

```

round(sum(orders_cnt - late_orders_cnt)/orders_cnt*100,2) "% success" --вывел долю
выполненных вовремя заказов в процентах и округлил до двух знаков после запятой
FROM table1
GROUP BY table1.ship_mode, table1.orders_cnt, table1.late_orders_cnt
ORDER BY "% success" asc --отсортировал запрос по доле выполненных вовремя
заказов в порядке возрастания

```



**Вывод:** чаще всего с опозданием доходят заказы отправленные вторым классом, их доля выполненных вовремя заказов составляет 79%.

Запрос, чтобы вывести долю заказов, отправленных вторым классом, которые были доставлены с опозданием, по кварталам:

```

WITH table1 as --использовал подзапрос CTE
(SELECT
ship_mode,
date_trunc('quarter', ship_date)::date date_quarter, --вывел дату доставки, округлённую до
квартала, конвертировал в дату
COUNT (order_id) as orders_cnt, --вывел общее количество заказов
COUNT (CASE WHEN ship_mode = 'Standard Class' AND (ship_date - order_date) > 6
THEN order_id
WHEN ship_mode = 'Second Class' AND (ship_date - order_date) > 4 THEN order_id
WHEN ship_mode = 'First Class' AND (ship_date - order_date) > 3 THEN order_id
WHEN ship_mode = 'Same Day' AND (ship_date - order_date) > 0 THEN order_id
END) as late_orders_cnt --с помощью оператор CASE вывел количество заказов,
которые не были доставлены вовремя
FROM sql.store_delivery
GROUP BY date_quarter, ship_mode)
SELECT
date_quarter,
round(sum(late_orders_cnt-orders_cnt)/orders_cnt*100+100,2) "% success" --вывел долю
невыполненных вовремя заказов в процентах и округлил до двух знаков после запятой

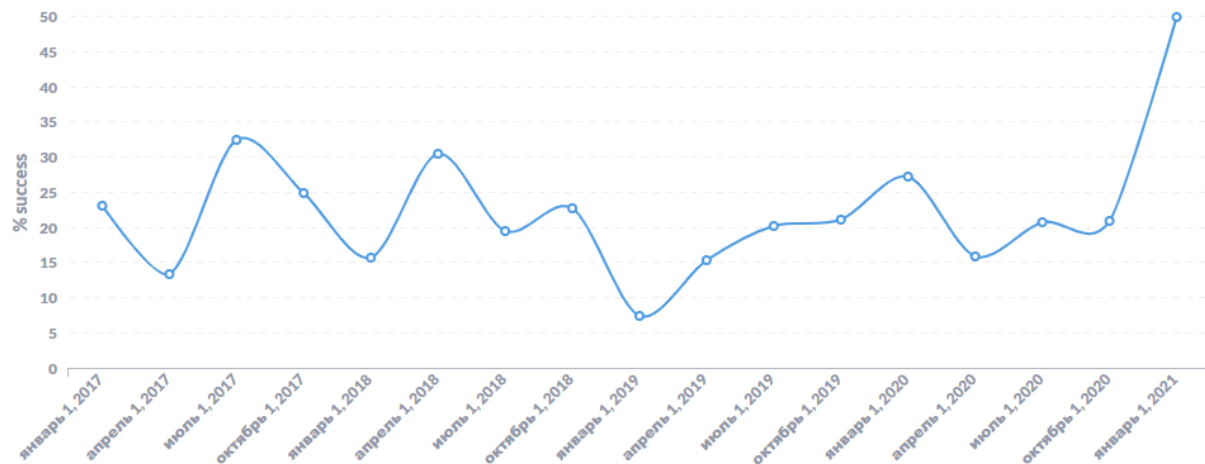
```

FROM table1

where ship\_mode like 'Second Class' --отфильтровал, оставив только доставку второго класса

GROUP BY table1.date\_quarter, table1.orders\_cnt, table1.late\_orders\_cnt

ORDER BY date\_quarter

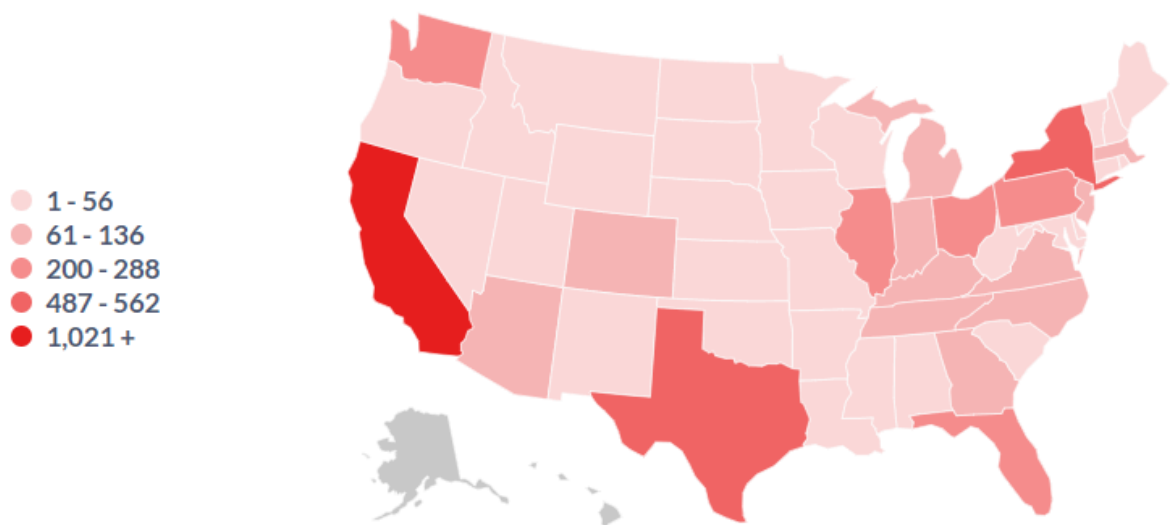


**Вывод:** по графику видно, что довольно часто заказы второго класса доставляются с опозданием, в среднем он составляет 22,49%, особенно большой всплеск составил в первом квартале 2021 года, его процент 50%.

Запрос, который выведет количество доставок по штатам:

```
SELECT
state,
COUNT (order_id) AS orders_cnt --вывел количество доставок
FROM sql.store_delivery
GROUP BY state
ORDER BY orders_cnt desc
```





**Вывод:** штат Калифорния наиболее популярен по количеству доставок, на втором месте штат Нью Йорк, но нас больше интересует город. В штате Нью Йорк находится самый популярный город Нью Йорк, до которого кратчайшее расстояние по воздуху от Хьюстона составляет 2281 км, на автомобиле 2622 км. Сравним расстояние со вторым город по популярности, которым является Лос-Анджелес, он находится в самом популярном штате Калифорния. Кратчайшее расстояние по воздуху Хьюстон - Лос-Анджелес составляет 2206 км., на автомобиле 2491 км. Лос-Анджелес находится ближе, следовательно стоимость доставки товара в будет ниже и так как этот город находится в наиболее популярном штате по количеству онлайн заказов. Ещё одним преимуществом является близкое расстояние до штата Вашингтон, в котором наша продукция пользуется хорошим спросом, что в перспективе из-за сокращения расстояния до офлайн-магазина увеличит количество заказов и привлечёт больше новых клиентов, поэтому новый офлайн-магазин следует открыть городе Лос-Анджелес.