

## ПРОЕКТ-2. Решение бизнес-задач с помощью SQL

### ЗАДАЧА

Необходимо проанализировать работу компании с точки зрения её эффективности и дать рекомендации по масштабированию бизнеса, а именно в каком штате лучше открыть офлайн-магазин.

### ФОРМАЛИЗОВАННАЯ ЗАДАЧА

1. Оценить динамику продаж и распределение выручки по товарам.
2. Составить портрет клиента: 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).

### 1. ОПРЕДЕЛЯЕМ ЭФФЕКТИВНОСТЬ ПРОДАЖ

- **Определяем сумму выручки по месяцам.**

#### SQL-запрос:

```
SELECT date_trunc('month',d.order_date)::date date,--отсекаем до месяца и выводим дату, переводим в тип "дата"
round(sum(quantity*price*(1-discount))) revenue --суммарное значение выручки с учетом скидки на товар, округляем до целого значения
FROM sql.store_carts c
JOIN sql.store_products p
ON p.product_id=c.product_id --соединяем с таблицей, чтобы использовать столбец "price"
JOIN sql.store_delivery d
ON d.order_id=c.order_id --соединяем с таблицей, чтобы использовать столбец даты заказа "order_date"
GROUP BY date_trunc('month',d.order_date) -- группируем по месяцу заказа
ORDER BY 1 --сортируем по возрастанию по дате заказа
```

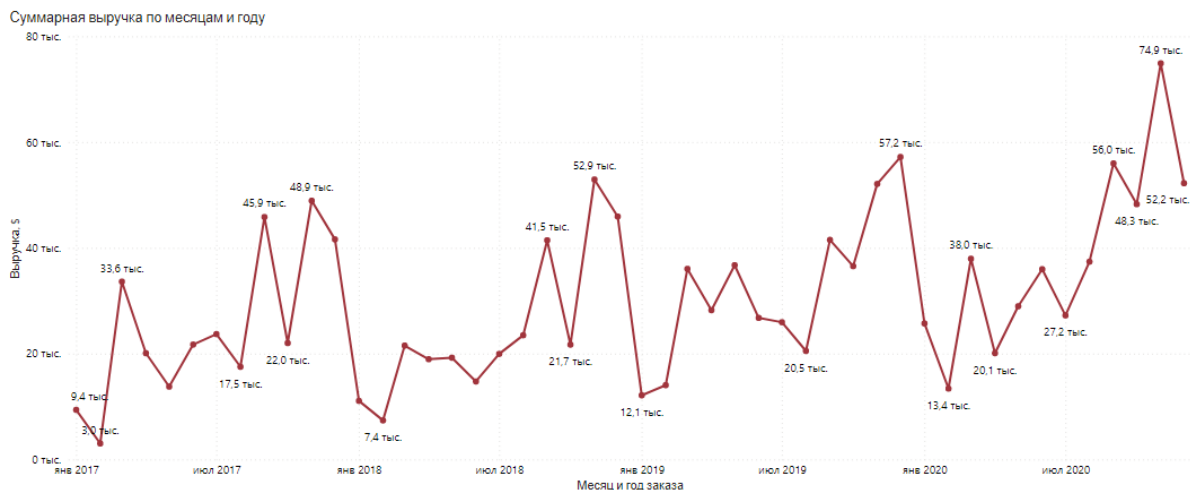


График 1. Динамика продаж по месяцам и году

По графику 1 можем отметить, что выручка выросла с 2017 по 2020 год в полтора раза, если рассматривать период ноябрь 2017 и 2020 годов. Отмечается

сезонность выручки: из года в год осенью объем выручки значительно возрастает, а летом и в первые месяцы года наоборот значительно падает.

- **Определяем сумму выручки по различным категориям и подкатегориям товаров.**

**SQL-запрос:**

```
SELECT p.category, -- выводим категорию товара
       p.subcategory, -- выводим подкатеорию товара
       round(sum(quantity*price*(1-discount))) revenue -- суммарное
значение выручки с учетом скидки на товар, округляем до целого значения
FROM sql.store_carts c
JOIN sql.store_products p
      ON p.product_id=c.product_id --соединяем с таблицей, чтобы
использовать столбец "price", "category", "subcategory"
JOIN sql.store_delivery d
      ON d.order_id=c.order_id --соединяем с таблицей, чтобы использовать
столбец даты заказа "order_date"
GROUP BY 1,2 --группируем по категории и подкатегории товара
ORDER BY 3 DESC --сортируем по убыванию выручки
```

По результатам запроса топ-3 категорий и подкатегорий товаров по объему выручки:

- Мебель - Стулья (235 318 \$);
- Технологии - Телефоны (221 110 \$);
- Офисные принадлежности - Места для хранения (179 736 \$).

- **Определим топ-25 товаров по объему выручки.**

**SQL-запрос:**

```
-- создаем CTE для определения выручки и количества проданного товара
with sm AS
  (SELECT p.product_id,
         p.product_nm,
         round(sum(quantity*price*(1-discount))),
         2) revenue ,--объем выручки по товару
         sum(quantity) quantity -- определяем суммарное количество
проданных товаров
  FROM sql.store_carts c
 JOIN sql.store_products p
        ON p.product_id=c.product_id --соединяем с таблицей, чтобы
использовать столбец "price", "product_id", "product_nm"
  GROUP BY 1,2)
SELECT product_nm, -- выводим название товара
       revenue , -- выводим выручку по товару
       sm.quantity, -- выводим количество проданных товаров
       round(revenue*100/sum(revenue)--доля от общей выручки в
процентах: делим объем выручки по товару на общий объем выручки, переводим в %,
округляем до 2х знаков после запятой
       OVER (),2) percent_from_total
FROM sql.store_carts c
JOIN sm
      ON sm.product_id=c.product_id--соединяем с созданным ранее CTE по id
GROUP BY 1,2,3
```

ORDER BY 2 DESC -- сортируем по убыванию объемов выручки  
limit 25 --выводим 25 первых товаров

Самую большую долю выручки 2,56% имеет товар “Canon imageCLASS 2200 Advanced Copier”, при этом он не имеет самый большой показатель по количеству продаж. Товар имеет значительную стоимость (является вторым товаром по самой наибольшей цене), поэтому его вклад в общую выручку достаточно высок по сравнению с другими.

Второй и третий товары имеют долю от общей выручки 1,06% и 0,99% соответственно. Вклад остальных товаров менее 0,94%.

### **Выводы:**

На основании выполненного анализа можно отметить хороший рост выручки с 2017 по 2020 год. При этом наблюдается сезонность объемов выручки: наиболее успешным периодом продаж является 4 квартал.

Определили топ-3 категорий и подкатегорий товаров, и топ-25 наименований товаров по объемам выручки.

## **2. Составляем портрет клиента**

- **Определим количество клиентов и выручку по категориям клиента.**

### **SQL-запрос:**

```
SELECT ct.category, --выводим категорию клиента
       count(distinct cust_nm) cust_cnt, -- выводим количество клиентов
       round(sum(c.quantity*p.price*(1-c.discount))) revenue -- суммарное
значение выручки с учетом скидки на товар, округляем до целого значения
FROM sql.store_customers ct
JOIN sql.store_delivery d --соединяем с таблицей, чтобы получить ключ
"order_id" к sql.store_carts
  ON d.cust_id=ct.cust_id
JOIN sql.store_carts c
  ON c.order_id=d.order_id--соединяем с таблицей, чтобы использовать
столбец "quantity", "discount" и получить ключ "product_id" к sql.store_products
JOIN sql.store_products p
  ON p.product_id=c.product_id --соединяем с таблицей, чтобы
использовать столбец "price"
GROUP BY 1 --группируем по категории клиента
ORDER BY 3 DESC --сортируем по убыванию выручки
```

Из результата запроса получаем, что компании (B2B-клиенты) приносят в разы больше выручки, чем розничные покупатели (B2C-клиенты). Также отметим, что корпоративных клиентов в 4 раза больше по количеству.

- **Определим количество новых корпоративных клиентов по месяцам.**

### **SQL-запрос:**

```
--создаем CTE для определения даты первого заказа
with new AS
  (SELECT min(date_trunc('month',order_date)::date) "month", ct.cust_id --
отсекаем до месяца и выводим первую дату заказа, переводим в тип "дата"
FROM sql.store_customers ct
JOIN sql.store_delivery d
```

```

ON d.cust_id=ct.cust_id --соединяем с таблицей, чтобы
использовать столбец "order_date"
WHERE ct.category='Corporate'--создаем условие учитывать только
корпоративных клиентов
GROUP BY 2) --группируем по id клиента
SELECT new.month, --выводим месяц первого заказа, т.е. когда клиент пришел
count(distinct cust_id) new_custs -- выводим количество новых
клиентов
FROM new--используем CTE
GROUP BY 1
ORDER BY 1 --сортируем по возрастанию даты заказа

```

Также можно определить количество новых клиентов при помощи оконной функции.

### **SQL-запрос:**

```

--создаем CTE для определения имени клиента, совершившего предыдущий
заказ, выбираем по id клиента и сортируем по дате заказа
with new AS
(SELECT *,
LAG(ct.cust_nm) over(partition by ct.cust_id
ORDER BY order_date) pr_date --определяем имя клиента, совершившего
предыдущий заказ, по id клиента с сортировкой по дате заказа
FROM sql.store_customers ct
JOIN sql.store_delivery d
ON d.cust_id=ct.cust_id --соединяем с таблицей, чтобы
использовать столбец "order_date"
WHERE ct.category='Corporate') --создаем условие выводить только
корпоративных клиентов
SELECT date_trunc('month',order_date)::date "month", -- отсекаем до месяца и
выводим первую дату заказа , переводим в тип "дата"
count(cust_nm) new_custs -- выводим количество новых клиентов
FROM new --используем CTE
WHERE pr_date is null -- создаем условие учитывать только тех клиентов, у
которых отсутствует предыдущее имя, т.е. отсутствует предыдущий заказ.
GROUP BY 1
ORDER BY 1 --сортируем по возрастанию даты заказа

```

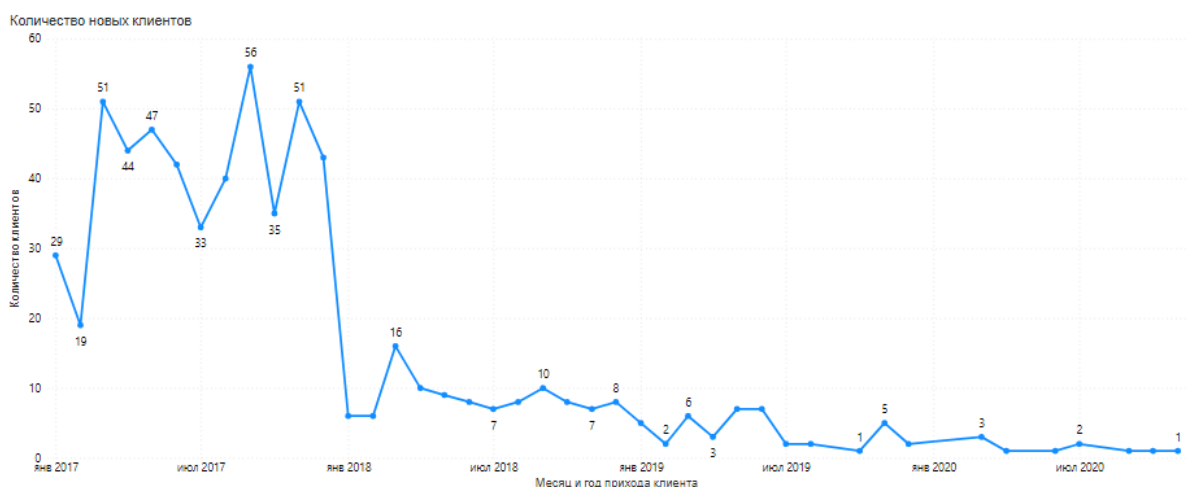


График 2. Количество привлечения новых клиентов по месяцам и году

Из графика 2 очевидно, что начиная с 2018 года новые клиенты почти не привлекались. Т.е. можно сделать вывод, что выручка растёт за счёт увеличения продаж старым клиентам.

- **Определяем сколько в среднем различных товаров в заказах у корпоративных клиентов.**

**SQL-запрос:**

```
--создаем CTE для определения количества товаров в каждом заказе
корпоративных клиентов
with avg_product AS
    (SELECT d.order_id,
        count(distinct c.product_id) a_order --количество различных
товаров заказе
        FROM sql.store_customers ct
        JOIN sql.store_delivery d
            ON d.cust_id=ct.cust_id --соединяем с таблицей, чтобы получить
ключ "order_id" к sql.store_carts
        JOIN sql.store_carts c
            ON c.order_id=d.order_id --соединяем с таблицей, чтобы
использовать столбец "product_id"
        WHERE ct.category='Corporate' --создаем условие учитывать
только корпоративных клиентов
        GROUP BY 1) -- группируем по id заказа
    SELECT round(AVG(a_order),1) --выводим среднее количество товаров в заказе,
округляем до 1 знака после запятой
    FROM avg_product --используем CTE
```

**В среднем у корпоративных клиентов в заказе 2 типа различных товаров.**

- **Определяем какая в среднем сумма заказов у корпоративных клиентов**

**SQL-запрос:**

```
--создаем CTE для определения суммы каждого заказа для корпоративных
клиентов
with avg_order AS
    (SELECT d.order_id,
        sum(quantity*price*(1-discount)) a_sum --сумма заказа с учетом
скидки на товар
        FROM sql.store_customers ct
        JOIN sql.store_delivery d
            ON d.cust_id=ct.cust_id --соединяем с таблицей, чтобы получить
ключ "order_id" к sql.store_carts и использовать столбец "order_id"
        JOIN sql.store_carts c --соединяем с таблицей, чтобы использовать
столбец "quantity", "discount" и получить ключ "product_id" к sql.store_products
            ON c.order_id=d.order_id
        JOIN sql.store_products p
            ON p.product_id=c.product_id --соединяем с таблицей, чтобы
использовать столбец "price"
        WHERE ct.category='Corporate' --создаем условие учитывать только
корпоративных клиентов
        GROUP BY 1) --группируем по id заказа
```

```
SELECT round(AVG(a_sum),1) --выводим среднюю сумму заказа, округляя до 1
знака после запятой
FROM avg_order --используем CTE
```

**В среднем сумма заказов у корпоративных клиентов составляет 285.9 \$.**

- **Определяем сколько в среднем различных офисов у корпоративных клиентов.**

#### **SQL-запрос:**

```
--создаем CTE для определения количества адресов доставок (офисов) у
корпоративных клиентов
with avg_address AS
    (SELECT ct.cust_id,
        count(distinct zip_code) a_zip -- количество различных адресов
        доставок
        FROM sql.store_customers ct
        JOIN sql.store_delivery d
        ON d.cust_id=ct.cust_id --соединяем с таблицей, чтобы
        использовать столбец "zip_code"
        WHERE ct.category='Corporate' --создаем условие учитывать только
        корпоративных клиентов
        GROUP BY 1)--группируем по id клиента
SELECT round(AVG(a_zip),1) --выводим среднее количество адресов доставки,
округляя до 1 знака после запятой
FROM avg_address --используем CTE
```

**В среднем различных офисов у корпоративных клиентов - 6.2.**

#### **Выводы:**

Значительную часть прибыли приносят клиенты-компании (B2B-клиенты).  
Выручка растет за счет старых клиентов, новые клиенты практически не привлекаются с 2018 года.  
В среднем корпоративные клиенты заказывают два различных товара в заказе на сумму 285,9 \$ и имеют 6 офисов (различных адресов доставок).  
Необходимо разработать и организовать маркетинговую кампанию по привлечению новых корпоративных клиентов.

### **3. АНАЛИЗ ЛОГИСТИКИ КОМПАНИИ**

- **Определяем какая доля заказов выполняется в срок по каждому типу доставки (*Standard Class, Second Class, First Class, Same Day*).**

#### **SQL-запрос:**

```
--создаем CTE, где находим общее количество заказов для каждого типа
доставки
WITH sum_orders AS
    (SELECT ship_mode, --тип доставки
        count(distinct order_id) orders_cnt --общее количество заказов
        FROM sql.store_delivery
        GROUP BY 1), --группируем по типу доставки
--создаем CTE, где находим количество заказов, которые были доставлены не
вовремя
last_orders AS
```

```

(SELECT ship_mode, --тип доставки
count(distinct order_id) late_orders_cnt --количество опоздавших
товаров
FROM sql.store_delivery
WHERE ship_mode='Standard Class'
AND (ship_date-order_date)>6
OR ship_mode='Second Class'
AND (ship_date-order_date)>4
OR ship_mode='First Class'
AND (ship_date-order_date)>3
OR ship_mode='Same Day'
AND (ship_date-order_date)>=1 --создаем условия
учитывать тип доставки и количество дней между доставкой и оформлением заказа
(когда заказ опаздывает)
GROUP BY 1) -- группируем по типу доставки
SELECT sum_orders.ship_mode, --выводим тип доставки
orders_cnt, --выводим общее количество заказов
late_orders_cnt, -- количество заказов, которые не были
доставлены вовремя
round(((orders_cnt-late_orders_cnt)*100/ orders_cnt::numeric,
2) "%_success" --доля выполненных вовремя заказов, в процентах,
округляем до двух знаков после запятой, переводим в тип "numeric" для получения
корректного ответа с округлением
FROM sum_orders --используем первое CTE
JOIN last_orders
ON last_orders.ship_mode=sum_orders.ship_mode --соединяем со вторым
CTE по типу доставки
ORDER BY 4 --сортируем по доле выполненных товаров в порядке возрастания

```

Из результата запроса видим, что тип доставки “First Class” имеет долю выполненных вовремя заказов 99,87%, т.е. почти все заказы доставляются вовремя.

В то время как тип доставки “Second Class” имеет долю 79,05%, т.е. каждый пятый заказ доставляется с опозданием.

- **Определяем долю заказов, отправленных вторым классом и которые были доставлены с опозданием, по кварталам.**

### **SQL-запрос:**

```

--создаем CTE, где находим общее количество заказов для типа доставки
"Second Class" по кварталам
WITH sum_orders AS
(SELECT date_trunc('quarter',order_date)::date quarter_orders, --отсекаем до
квартала и выводим дату заказа, переводим в тип "дата"
count(distinct order_id) orders_cnt --определяем общее количество заказов
FROM sql.store_delivery
WHERE ship_mode='Second Class' -- создаем условие учитывать только
тип доставки "Second Class"
GROUP BY 1), -- группируем по кварталам
--создаем CTE, где определяем количество заказов, доставленных с
опозданием по кварталам
last_orders AS
(SELECT date_trunc('quarter',order_date)::date quarter_last, --отсекаем до
квартала и выводим дату заказа, переводим в тип "дата"

```



```

count(distinct order_id) late_orders_cnt --определяем количество заказов,
которые опаздывают
FROM sql.store_delivery
WHERE ship_mode='Second Class'
AND (ship_date-order_date)>4 -- создаем условия учитывать
только тип доставки "Second Class" и количество дней между доставкой и
оформлением заказа (когда заказ опаздывает)
GROUP BY 1) -- группируем по кварталам
SELECT quarter_orders, -- квартал, в котором был оформлен заказ
round(late_orders_cnt*100/ orders_cnt::numeric,
2) "%_success" --доля заказов, которые были доставлены с
опозданием в %, округляем до двух знаков после запятой, переводим в тип "numeric"
для получения корректного ответа с округлением
FROM sum_orders --используем первое CTE
JOIN last_orders
ON last_orders.quarter_last=sum_orders.quarter_orders ----соединяем со
вторым CTE по кварталу заказа
ORDER BY 1 --сортируем по убыванию даты

```



График 3. Доля заказов, доставленных с опозданием, по кварталам

Квартал с самой низкой долей заказов, доставленных с опозданием - это первый квартал 2019 года. До 2019 года доля опоздавших заказов составляла 13-35% от общего количества заказов, со второго квартала 2019 доля составляет 15-22%. Средний показатель доли заказов, доставленных не вовремя, за все года составляет 20,7%. По графику можно сделать вывод, что нет закономерности в нарушении сроков доставки вторым классом в зависимости от времени года, стабильно около 20% заказов не доставляются вовремя. Из улучшений можно отметить, что после 3 квартала 2018 года отсутствует доля опоздавших заказов более 26%.

- **Определяем количество доставок по штатам.**

**SQL-запрос:**

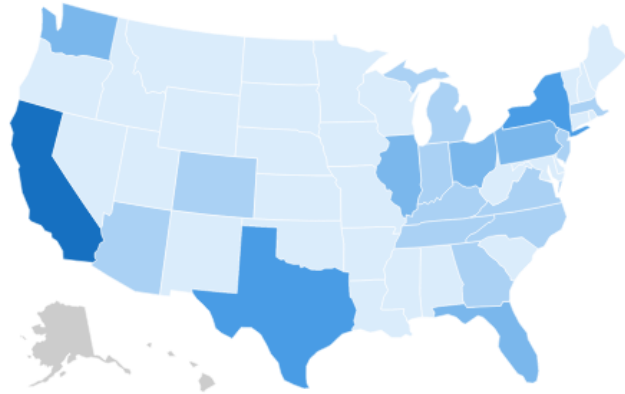
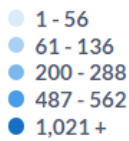
```

SELECT state, -- выводим штат доставки
count(distinct order_id) -- выводим общее количество заказов
FROM sql.store_delivery
GROUP BY 1 -- группируем по штатам
ORDER BY 2 desc --группируем по количеству заказов в порядке убывания

```

Наиболее популярный штат по количеству доставок - штат California, туда доставлялся 1021 заказ, затем идет штат New York - 562 заказа, и штат Texas - 487 заказов.





Карта распределения количества заказов по штатам в США.

### **Выводы:**

Необходимо улучшить логистику компании для типа доставки “Second Class”, т.к. для этого типа стабильно каждый пятый заказ доставляется с опозданием вне зависимости от времени года.

Наиболее перспективным для открытия офлайн-магазина является штат California. Он имеет большую площадь территории, соответственно можно привлечь больше новых клиентов. Штат расположен недалеко от штата, где находится склад магазина, т.е. логистические расходы будут оптимальными. Также рядом расположен штат Arizona, в которые тоже осуществлялись доставки и они могут стать потенциальными клиентами офлайн магазина в California.

Также можно задуматься на перспективу открыться в штате New York, т.к. рядом с этим штатом расположены еще два штата Pennsylvania и Ohio, куда было осуществлено более 230 доставок в каждый и они могут также быть заинтересованы в открытии офлайн магазина недалеко от их штата. И город New York City является лидером среди городов с наибольшим количеством доставок (450 заказов).

## ОБЩИЕ ВЫВОДЫ

1. Отмечается хороший рост выручки с 2017 по 2020 год. При этом наблюдается сезонность в объемах выручки: наиболее успешным периодом продаж является 4 квартал каждого года.

Топ-3 категорий и подкатегорий товаров по объему выручки:

- Мебель - Стулья (235 318 \$);
- Технологии - Телефоны (221 110 \$);
- Офисные принадлежности - Места для хранения (179 736 \$).

Самую большую долю от общей выручки 2,56% имеет товар “Canon imageCLASS 2200 Advanced Copier”, за счет своей высокой стоимости (второй по цене товар) он вносит значительный вклад в общий объем прибыли.

Рекомендуется запускать рекламные кампании в наиболее удачные периоды года, для привлечения еще большего количества клиентов. Также можно продумать систему лояльности/скидок на повторные заказы для поддержания уровня продаж в другие периоды.

2. Значительную часть прибыли приносят клиенты-компании (B2B-клиенты). При этом выручка растет за счет старых клиентов, новые клиенты практически не привлекаются с 2018 года.

В среднем корпоративные клиенты заказывают два различных товара в одном заказе на сумму 285,9 \$ и имеют 6 офисов (различных адресов доставок).

Необходимо разработать стратегию по привлечению новых корпоративных клиентов.

3. Необходимо улучшить логистику типа доставки “Second Class”, т.к. каждый пятый заказ доставляется с опозданием вне зависимости от времени года.

Наиболее перспективным для открытия офлайн-магазина является штат California. Он имеет большую площадь территории, соответственно можно привлечь больше новых клиентов. Штат расположен недалеко от штата, где находится склад магазина, т.е. логистические расходы будут оптимальными.

В перспективе можно открыть офлайн магазин в штате New York, т.к. рядом с этим штатом расположены еще два штата Pennsylvania и Ohio, куда было осуществлено более 230 доставок в каждый и они могут также быть заинтересованы в открытии офлайн магазина недалеко от их штата. И отметим, что город New York City является лидером среди городов с наибольшим количеством доставок (450 заказов).

**В целом у компании хорошая тенденция развития, наблюдается рост прибыли, старые корпоративные клиенты являются основным источником продаж. Рекомендуется разработать грамотные эффективные маркетинговые кампании, устранить выявленные в процессе анализа недостатки и масштабировать бизнес в других штатах.**