**A young but promising company, Pens and Pencils, is based in the USA and operates there as well. Since March 2017, its focus has been on B2B and B2C online sales of office supplies (office equipment, stationery, and furniture).**

**Formalized tasks:**

**1. Evaluate the sales dynamics and revenue distribution by products.**

**2. Create a customer profile by determining which customers generate the highest revenue.**

**3. Monitor the company's logistics (determine if all orders are delivered on time and identify the best state to open an offline store).**

To accomplish these tasks, I utilized the following tools, data, and methods:

1. Data Analysis: PostgreSQL was employed to manage and analyze the company's actual data. Querying the database and performing data manipulations provided insights into various aspects of the business.
2. Statistical Methods: Statistical techniques and methodologies were applied to analyze sales dynamics and revenue distribution. Descriptive statistics, regression analysis, and hypothesis testing were utilized to identify patterns, trends, and relationships within the data.
3. PowerBI: PowerBI, a powerful data visualization tool, was utilized to create interactive dashboards and reports. By connecting to the data sources, visual representations of key metrics and performance indicators were generated, facilitating a comprehensive understanding of the company's sales and revenue patterns.

By employing these tools, leveraging data, and applying statistical methods, meaningful insights were derived, and data-driven decisions were made for the company.

**27.5.1**

**Rounded monthly sales**

-- revenue by order

with sales\_by\_orders as

(

select

sc.order\_id,

sum ((price - (price \* discount)) \* quantity) revenue

from sql.store\_products sp

join sql.store\_carts sc using (product\_id)

group by order\_id

)

-- rounded revenue by months

select

to\_char(sd.order\_date, 'yyyy-mm-01') date,

round (sum (revenue),0) revenue

from sales\_by\_orders sbo

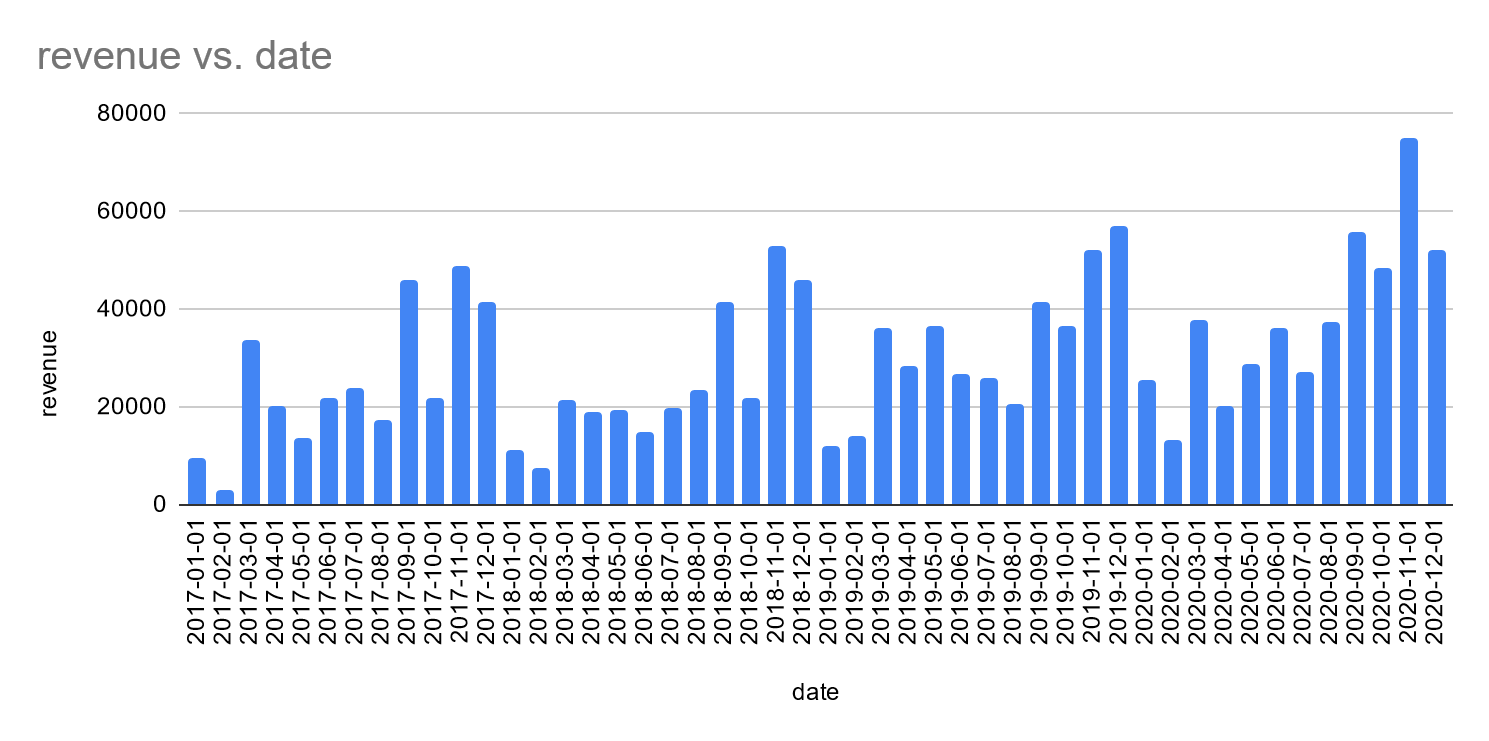
join sql.store\_delivery sd using (order\_id)

group by date

order by date

**27.5.2**

Sales exhibit seasonality. January and February are the months with the lowest sales. But overall, sales are growing.



**27.5.5**

**Sales revenue split by category and subcategory**

-- revenue by category and subcategory

select

sp.category,

sp.subcategory,

round (sum (sc.quantity\*(1-sc.discount)\*sp.price)) revenue

from sql.store\_carts sc

join sql.store\_products sp using (product\_id)

group by 1,2

order by 3 desc

**27.5.6**

-- **Top 25 products by sales revenue**

with top25\_sales as

(

select

sp.product\_nm,

sum ((price - (price \* discount)) \* quantity) revenue,

sum (sc.quantity) quantity

from sql.store\_products sp

join sql.store\_carts sc using (product\_id)

group by 1

order by 2 desc

limit 25

),

-- Total revenue

total\_revenue as

(

select

sum ((price - (price \* discount)) \* quantity) total

from sql.store\_products sp

join sql.store\_carts sc using (product\_id)

)

-- top 25 products revenue, quantity and percentage share

select

product\_nm,

round (revenue,2) revenue,

quantity,

(

select

round (revenue/total\*100,2)

from total\_revenue t

) percent\_from\_total

from top25\_sales ts

**27.6.1**

-- **Sales revenue by customer**

with sales\_by\_customers as

(

select

sd.cust\_id,

sum ((price - (price \* discount)) \* quantity) revenue

from sql.store\_products sp

join sql.store\_carts sc using (product\_id)

join sql.store\_delivery sd using (order\_id)

group by cust\_id

)

select

category,

count (distinct (cust\_id)) cust\_cnt,

round (sum (revenue)) revenue

from sql.store\_customers sc

join sales\_by\_customers using (cust\_id)

group by category

order by revenue desc

**27.6.2**

-- **Creating corporate customer cohorts by the 1st order**

with frst\_order as

(

select

min (to\_char(sd.order\_date, 'yyyy-mm-01')) date,

cust\_id

from sql.store\_delivery sd

group by cust\_id

)

-- Count of new corporate customers by month

select

fo.date,

count (distinct (cust\_id)) new\_custs

from sql.store\_customers sc

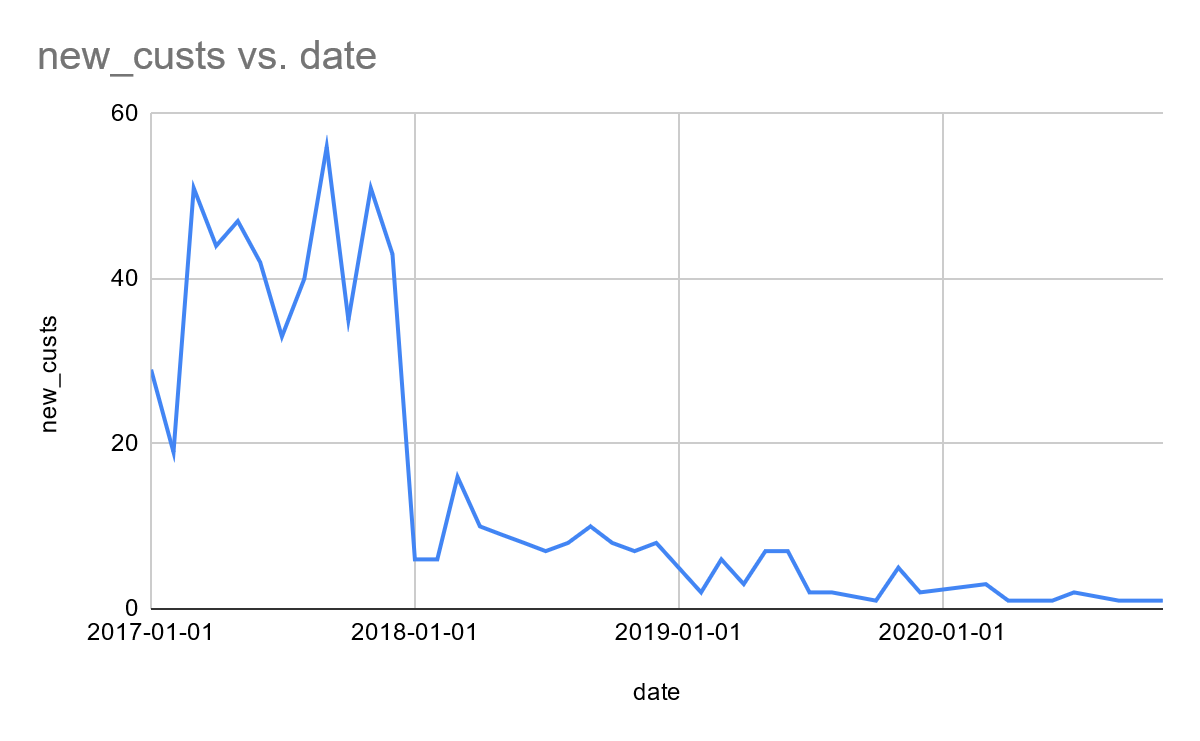
join frst\_order fo using (cust\_id)

where category like 'Corporate'

group by date

order by date

Since the beginning of 2018, there have been noticeably fewer new customers, perhaps indicating that campaigns with the highest lead traffic were scaled back.



**27.6.4**

-- **1) Average number of items per order for corporate clients (2)**

with customer\_orders as

(

select

sd.order\_id,

count (distinct(scc.product\_id)) items

from sql.store\_delivery sd

join sql.store\_customers sc on sd.cust\_id = sc.cust\_id

join sql.store\_carts scc on scc.order\_id = sd.order\_id

where sc.category like 'Corporate'

group by 1

)

select

round (avg(items),1)

from customer\_orders

-- **2) The average buy for corporate customers (285,9)**

with order\_sum as

(

select

sc.order\_id,

sum ((price - (price \* discount)) \* quantity) revenue

from sql.store\_products sp

join sql.store\_carts sc using (product\_id)

group by sc.order\_id

),

corp\_cust as

(

select

sd.order\_id

from sql.store\_customers sc

join sql.store\_delivery sd using (cust\_id)

where category like 'Corporate'

group by sd.order\_id

)

select

round ( avg (revenue),1)

from corp\_cust cc

join order\_sum os using (order\_id)

**The average number of offices for corporate clients (6.2)**

-- **3) The total number of branches per corporate customer**

with corp\_cust as

(

select

cust\_id,

count(distinct (zip\_code)) branch\_num

from sql.store\_customers sc

join sql.store\_delivery sd using (cust\_id)

where category like 'Corporate'

group by cust\_id

)

-- average no of offices per corporate customer

select

round (avg(branch\_num),1)

from corp\_cust cc

**27.7.1**

-- **% of successful deliveries by class**

with delivery\_intervals as

(

select

ship\_mode,

count (order\_id) as orders\_cnt,

count (case when ship\_mode = 'Same Day' and (ship\_date - order\_date) <= 0 then order\_id

when ship\_mode = 'First Class' and (ship\_date - order\_date) <= 3 then order\_id

when ship\_mode = 'Second Class' and (ship\_date - order\_date) <= 4 then order\_id

when ship\_mode = 'Standard Class' and (ship\_date - order\_date) <= 6 then order\_id

else null end) as delievered\_ontime

from sql.store\_delivery

group by 1

)

select ship\_mode,

orders\_cnt,

(orders\_cnt - delievered\_ontime) late\_orders\_cnt,

round (delievered\_ontime \* 100.00 / orders\_cnt, 2) as "% success"

from delivery\_intervals

order by 4

**27.7.2**

**-- The quantity of failed 'Second Class' deliveries by quarter**

with delivery\_intervals as

(

select

DATE\_TRUNC('quarter', order\_date)::date quarter,

ship\_mode,

count (order\_id) as orders\_cnt,

count (case when ship\_mode = 'Same Day' and (ship\_date - order\_date) <= 0 then order\_id

when ship\_mode = 'First Class' and (ship\_date - order\_date) <= 3 then order\_id

when ship\_mode = 'Second Class' and (ship\_date - order\_date) <= 4 then order\_id

when ship\_mode = 'Standard Class' and (ship\_date - order\_date) <= 6 then order\_id

else null end) as delievered\_ontime

from sql.store\_delivery

where ship\_mode = 'Second Class'

group by 1,2

)

select

quarter,

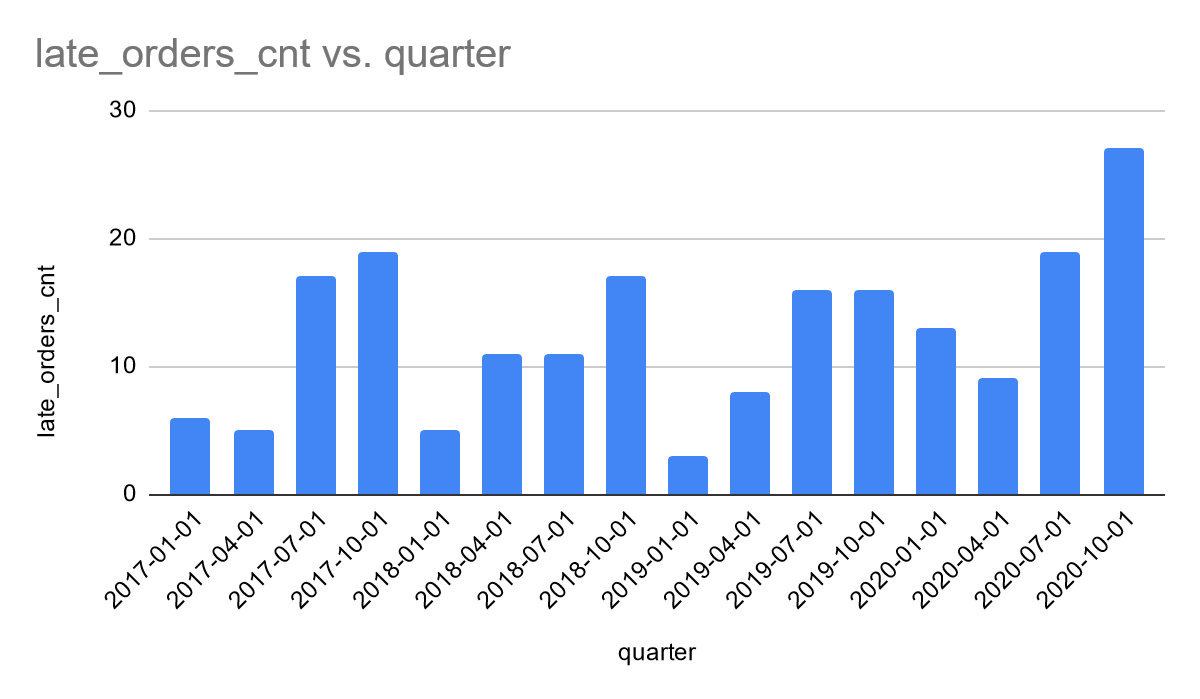
(orders\_cnt - delievered\_ontime) late\_orders\_cnt

from delivery\_intervals

order by 1

**27.7.3**

Delivery deadlines have been missed throughout the observed period. This is most likely due to fluctuations in sales and logistical issues.

****

**27.7.6**

**-- Sales revenue by state**

select

state,

count (distinct (order\_id))

from sql.store\_delivery sd

group by state

order by count desc

Based on the sales map below, California appears to be the state with the highest sales volume. The next state in terms of sales on the West Coast is Washington. It would be recommended to consider establishing an offline sales point in Los Angeles, as it is the largest city with the highest purchasing power in California.

