Analysis Sales Report (Visualization)

INSIGHT DATA & TOOLS

* This analysis uses the detail_transaksi dataset, and it has the same condition as MySQL analysis report

* This analysis uses Google Colab, Matplotlib-Seaborn Python and Tableau

* The Tableau analysis file was saved in the twb format, in GITHUB and Google drive

MATPLOTLIB & SEABORN

```
[38] import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import seaborn as sns
     from matplotlib.ticker import FuncFormatter
```

```
[2]  df = pd.read_excel('detail_transaksi.xlsx')
```

```
[49] # Filter data dengan KOMISI = 'ST'
     filter = df[(df['KOMISI']=='ST') & (df['PAYMENT_STATUS'] =='PAID')].copy()
```

```
[ ]  revenue = filter.groupby('DEPARTMENT')['NILAI REVENUE PRODUCT'].sum().reset_index()
```

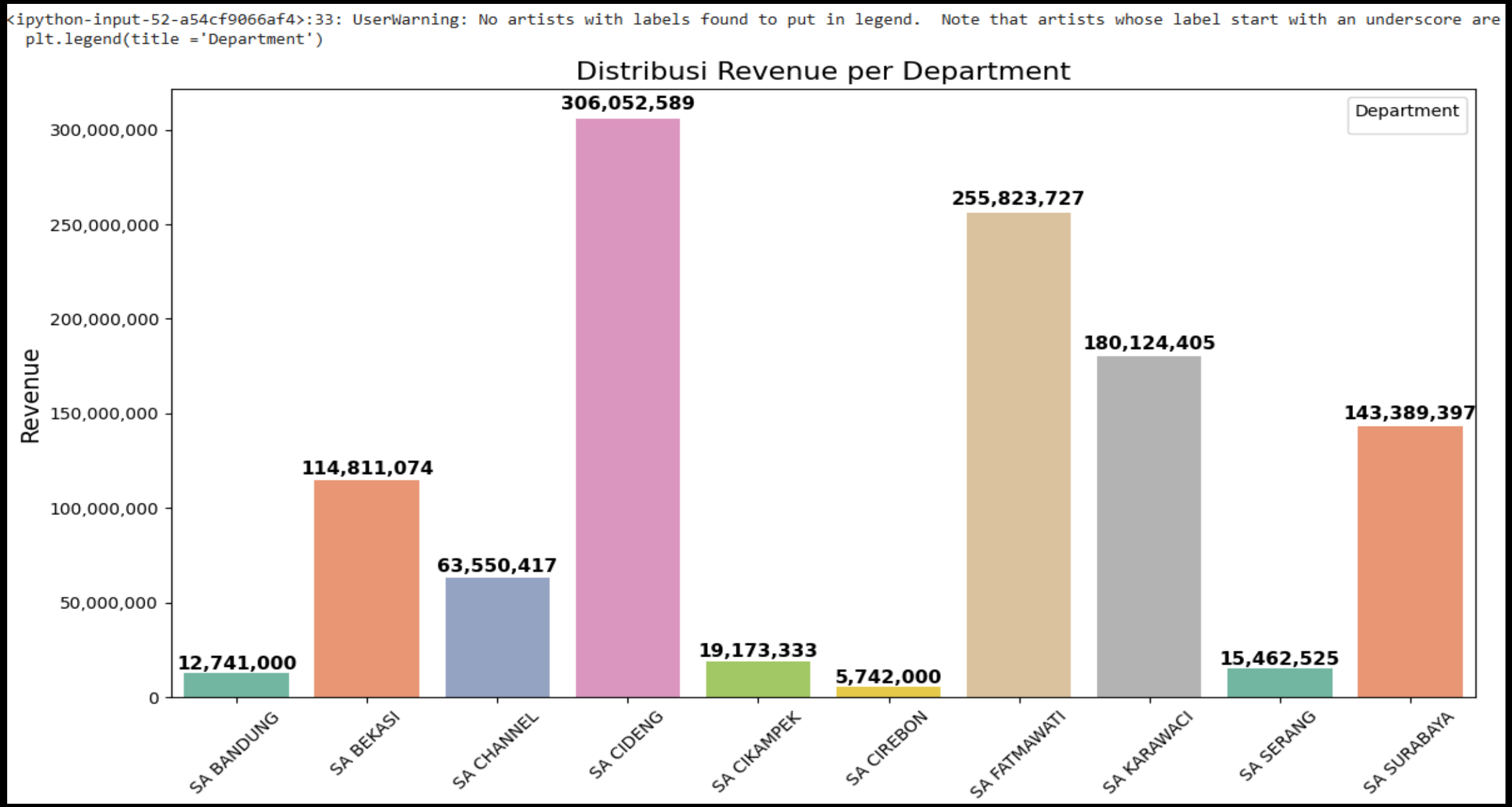
1.

2

BAR PLOT

```
#Membuat barplot
plt.figure(figsize=(12, 7))
ax = sns.barplot(data=revenue,
                 x='DEPARTMENT',
                 y ='NILAI REVENUE PRODUCT',
                 hue='DEPARTMENT',
                 palette='Set2',
                 legend=False)

# fungsi format untuk menampilkan angka dengan koma sebagai pemisah ribuan
def format_millions(x, pos):
   # fromat angka dengan koma tanpa desimal
    return f'{int(x):,}'

    # Terapkan format angka pada sumbu Y
ax.yaxis.set_major_formatter(FuncFormatter(format_millions))

    # Manambahkan angka disetiap bar
for p in ax.patches:
        height = p.get_height()
        ax.text(p.get_x() + p.get_width() / 2, # horizontal position in the mid bar
        height + (0.01 * height),              # slightly vertical position above the bar
               f'{int(height):,}',             # number text with comma
               ha='center',                    # horizontal alignment center
               va='bottom',                    # vertical alignment below the text
               fontsize = 11,
               fontweight='bold')

plt.title('Distribusi Revenue per Department',fontsize = 16)
plt.xlabel('Department', fontsize=14)
plt.ylabel('Revenue', fontsize=14)
plt.xticks(rotation=45)
plt.legend(title ='Department')
plt.tight_layout()
plt.show()
```

```
<ipython-input-52-a54cf9066af4>:33: UserWarning: No artists with labels found to put in legend.  Note that artists whose label start with an underscore are
  plt.legend(title ='Department')
```

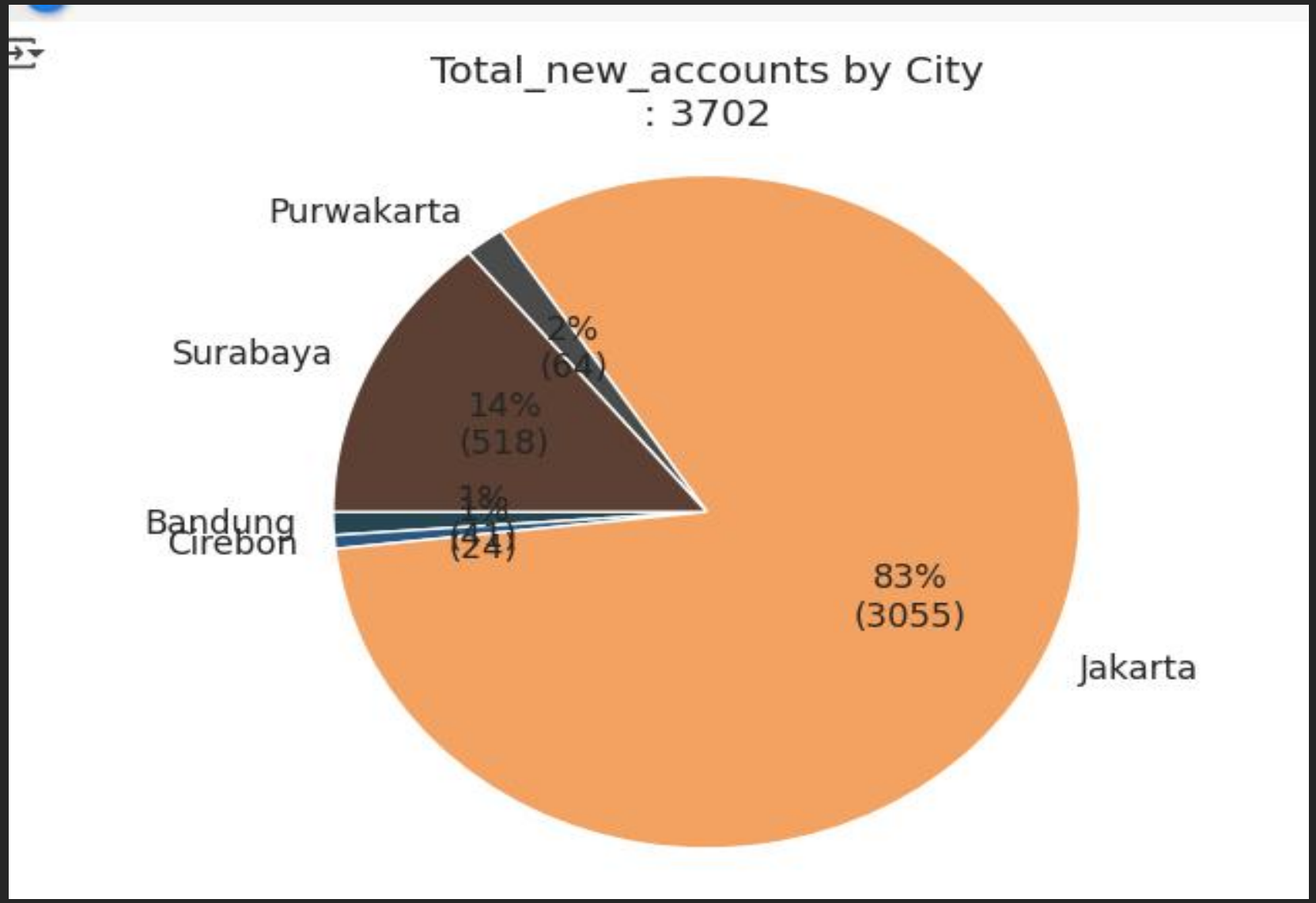


```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
!pip install pandas matplotlib seaborn openpyxl

df = pd.read_excel('detail_transaksi.xlsx')

# Use seaborn style for visual
df = pd.read_excel('detail_transaksi.xlsx')

# filter rows from KOMISI = ST and Payment status = PAID
flt = df[(df['KOMISI']=='ST') & (df['PAYMENT_STATUS']=='PAID')]

# assuming each row is one account, so count rows per city
# Group by city and count total accounts ( counting unique CUST_ACCT)
group_by_city = flt.groupby('CITY').size()

# Calculate total accounts
Total_new_accounts = group_by_city.sum()
```

PIE CHART

```
# Plot pie chart
plt.figure(figsize=(10, 10))

# plot color pie chart
#cl = sns.color_palette(['#264653', '#E63946', '#F4A261', '#2A9D8F'])[:len(group_by_city)]
cl = sns.color_palette(['#264653','#2B5783', '#F4A261', '#4B4B4B', '#5C4033'])[:len(group_by_city)]

# Define a function to show both percentage and absolute counts in label of pie chart
def autopct_format(p):
  absolute = int(round(p * Total_new_accounts /100.0))
  return f"{p:.0f}%\n({absolute})"

plt.pie(
    group_by_city,
    labels=group_by_city.index,
    autopct=autopct_format,
    colors=cl,
    startangle=180,
    textprops={'fontsize': 12}
)
plt.title(f'Total_new_accounts by City\n: {Total_new_accounts}', fontsize=14)
plt.axis('equal')
plt.tight_layout()
```

```
plt.title(f'Total_new_accounts by City\n: {Total_new_accounts}', fontsize=14)
plt.axis('equal')
plt.tight_layout()
plt.show()
```



```
[2]  import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import seaborn as sns
     from sklearn.cluster import KMeans
```

```
[5]  df = pd.read_excel('detail_transaksi.xlsx')
```

```
[6]  # filter data
     flt = df[(df['KOMISI']=='ST') & (df['PAYMENT_STATUS']=='PAID')]
```

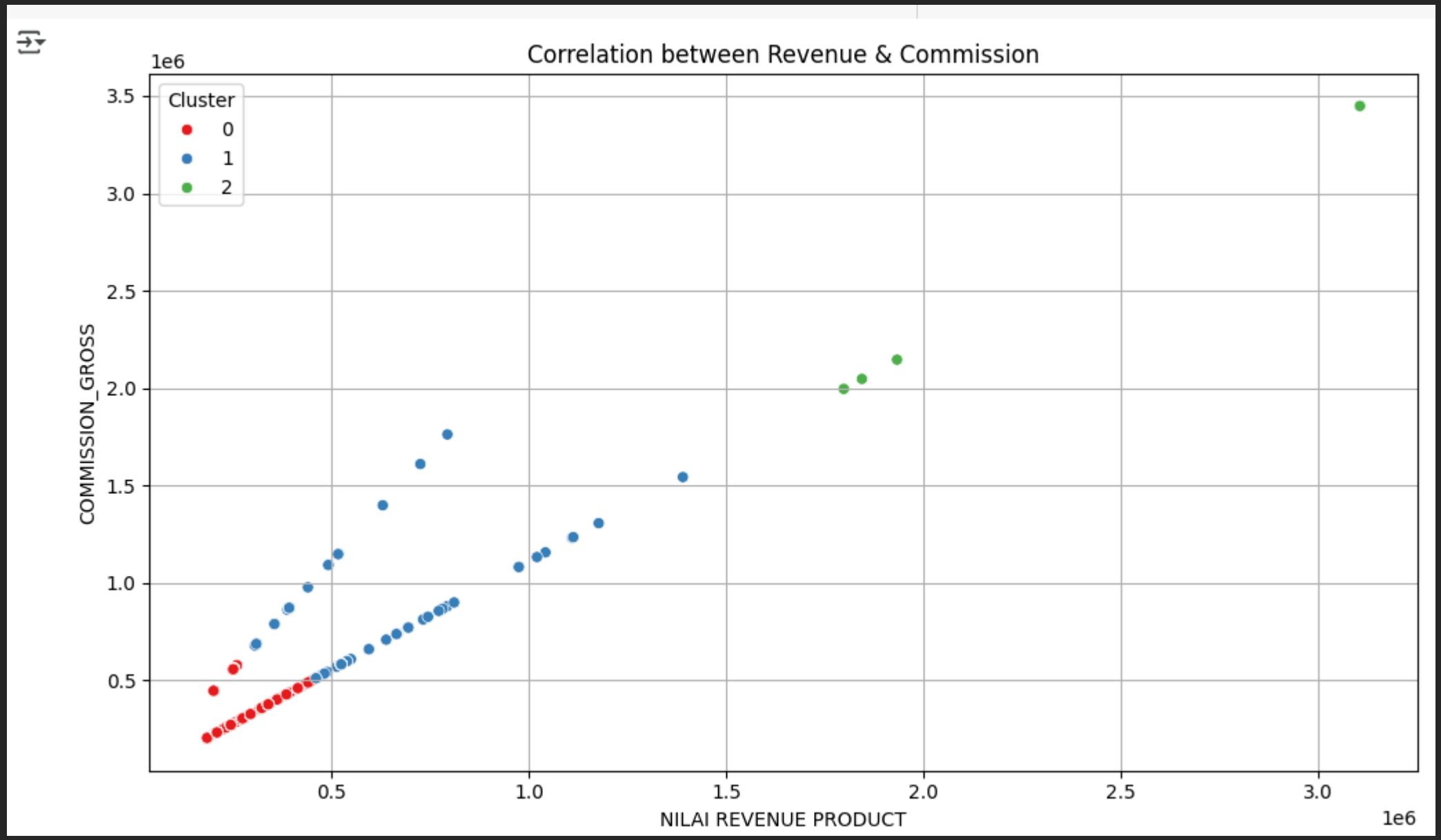
```
     # Select relevant columns and drop missng values
     drop_na = flt[['NILAI REVENUE PRODUCT', 'COMMISSION_GROSS']].dropna()
```

```
[8]  #Kmeans clustering
     from sklearn.cluster import KMeans
     k_means = KMeans(n_clusters=3,random_state=0)
     cluster = k_means.fit_predict(drop_na)
```

```
     # Add cluster label to dataframe
     flt = flt.loc[drop_na.index]
     flt['cluster']=cluster
```

```
[12] # Plot with cluster colors
     plt.figure(figsize=(10,6))
     sns.scatterplot(data=flt, x='NILAI REVENUE PRODUCT', y='COMMISSION_GROSS', hue='cluster', palette='Set1')
     plt.title('Correlation between Revenue & Commission')
     plt.xlabel('NILAI REVENUE PRODUCT')
     plt.ylabel('COMMISSION_GROSS')
     plt.legend(title='Cluster')
     plt.grid(True)
     plt.tight_layout()
     plt.show()
```

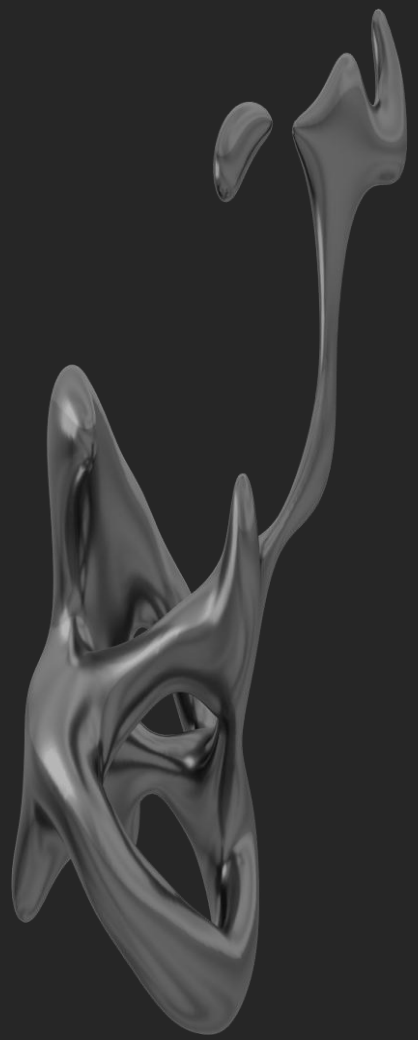
SCATTER PLOT

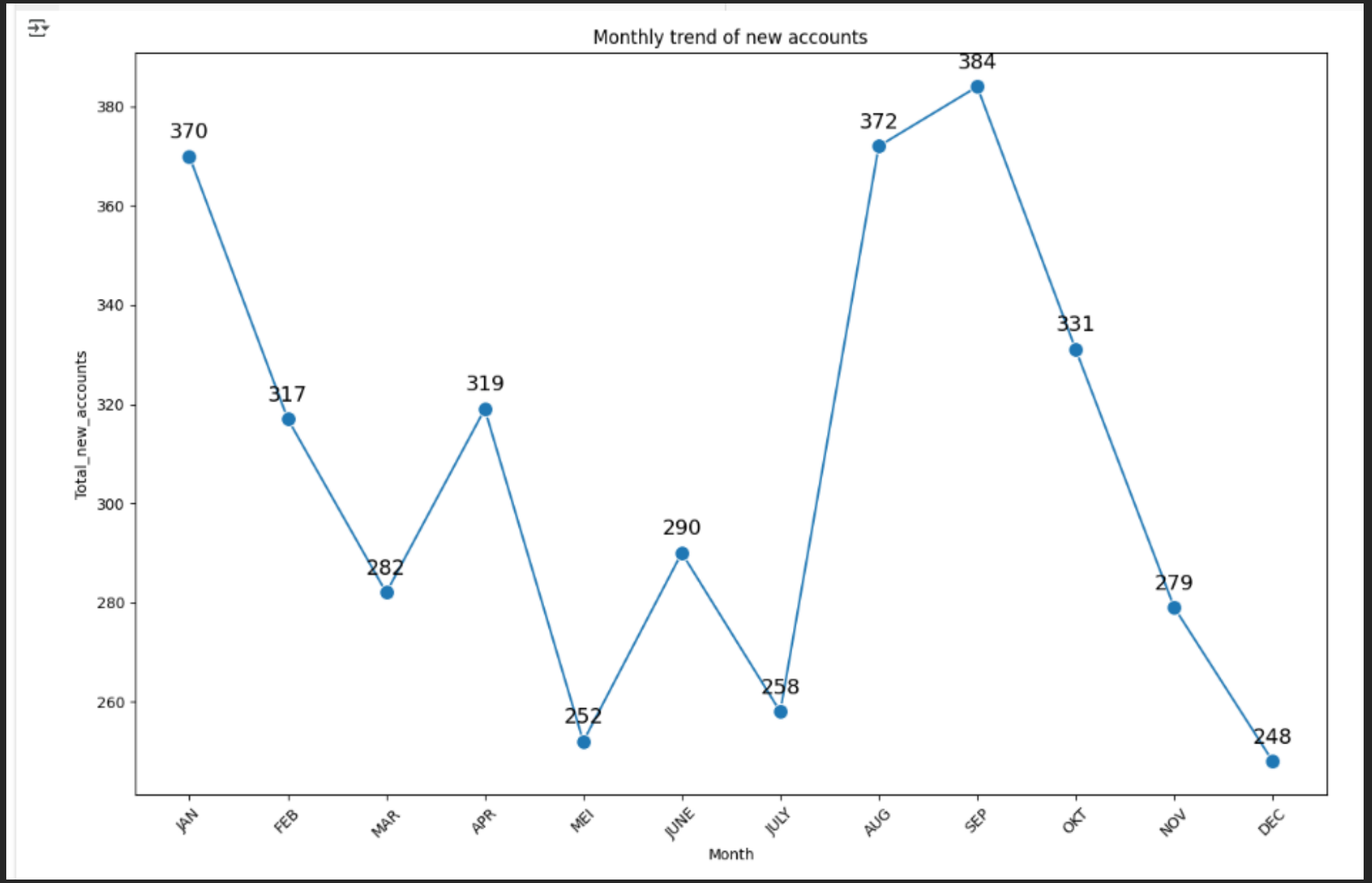

Correlation between Revenue & Commission
1e6
Cluster
0
1
2
COMMISSION_GROSS
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.5
1.0
1.5
2.0
2.5
3.0
NILAI REVENUE PRODUCT
1e6

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Membaca data
df = pd.read_excel('detail_transaksi.xlsx')

# Filter rows where KOMISI = ST AND PAYMENT_STATUS = PAID
flt = df[(df['KOMISI'] == 'ST') & (df['PAYMENT_STATUS'] == 'PAID')]

# Group by TERM PAYMENT and count unique accounts [CUST_ACCT]
Total_new_accounts = flt.groupby('TERM PAYMENT')['CUST_ACCT'].nunique().reset_index()

# Define the correct order of the months
month_order = ['JAN', 'FEB', 'MAR', 'APR', 'MEI', 'JUNE', 'JULY', 'AUG', 'SEP', 'OKT', 'NOV', 'DEC']

# Ensure 'TERM PAYMENT' is the categorically correct order
Total_new_accounts['TERM PAYMENT'] = pd.Categorical(
    Total_new_accounts['TERM PAYMENT'],
    categories=month_order,
    ordered=True
)

# Sorting by month order
Total_new_accounts = Total_new_accounts.sort_values('TERM PAYMENT')

# Line plotting
plt.figure(figsize=(12, 8))
sns.lineplot(data=Total_new_accounts, x='TERM PAYMENT', y='CUST_ACCT', marker='o', markersize=10)
plt.title('Monthly trend of new accounts')
plt.xlabel('Month')
```

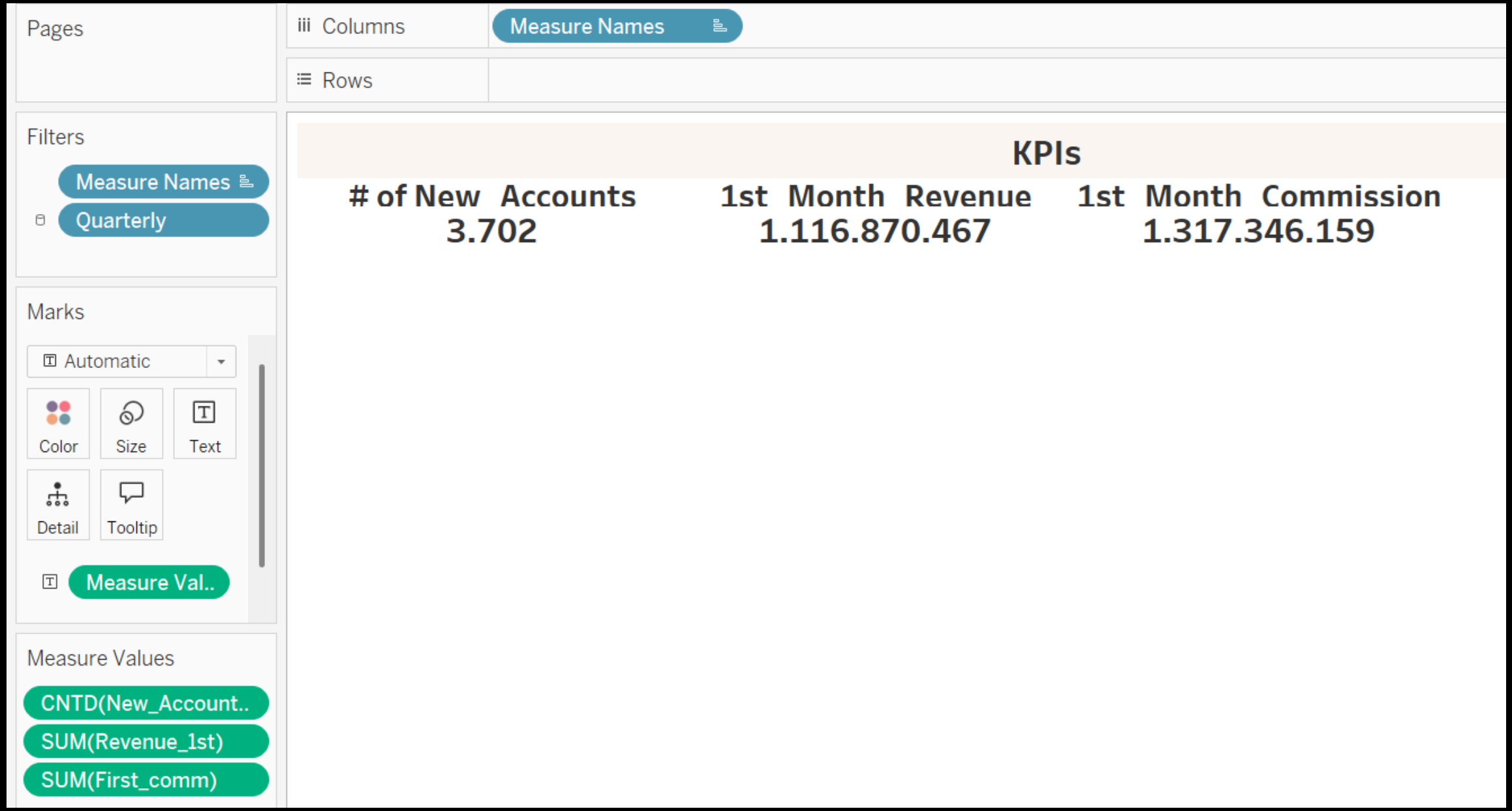
```
plt.ylabel('Total_new_accounts')
plt.grid(True)
plt.xticks(rotation=45)
plt.tight_layout()

# Adding numbered label in each marker
for i, row in Total_new_accounts.iterrows():
    plt.annotate(
        f'{row.CUST_ACCT}',
        (row['TERM PAYMENT'], row.CUST_ACCT),
        textcoords='offset points',
        xytext=(0, 12),  # label's position is above the dot
        ha='center',
        fontsize=14
    )

plt.show()
```

LINE PLOT

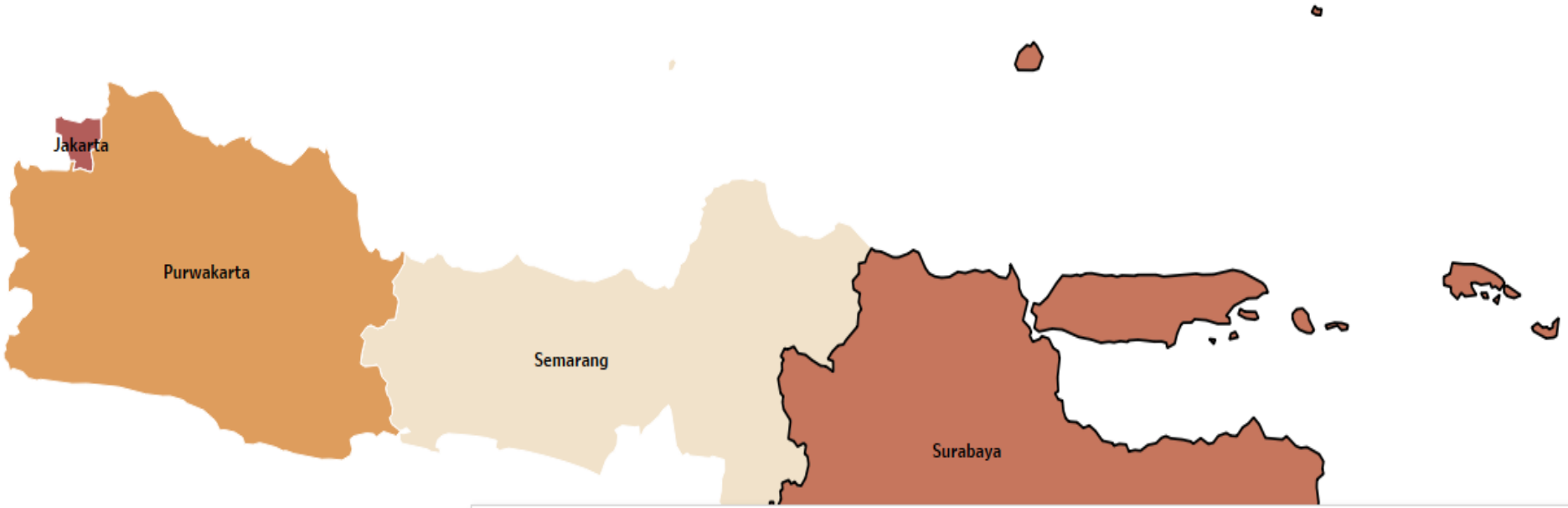



Monthly trend of new accounts
Total_new_accounts
Month
380
360
340
320
300
280
260
370
317
282
319
252
290
258
372
384
331
279
248
JAN
FEB
MAR
APR
MEI
JUNE
JULY
AUG
SEP
OKT
NOV
DEC

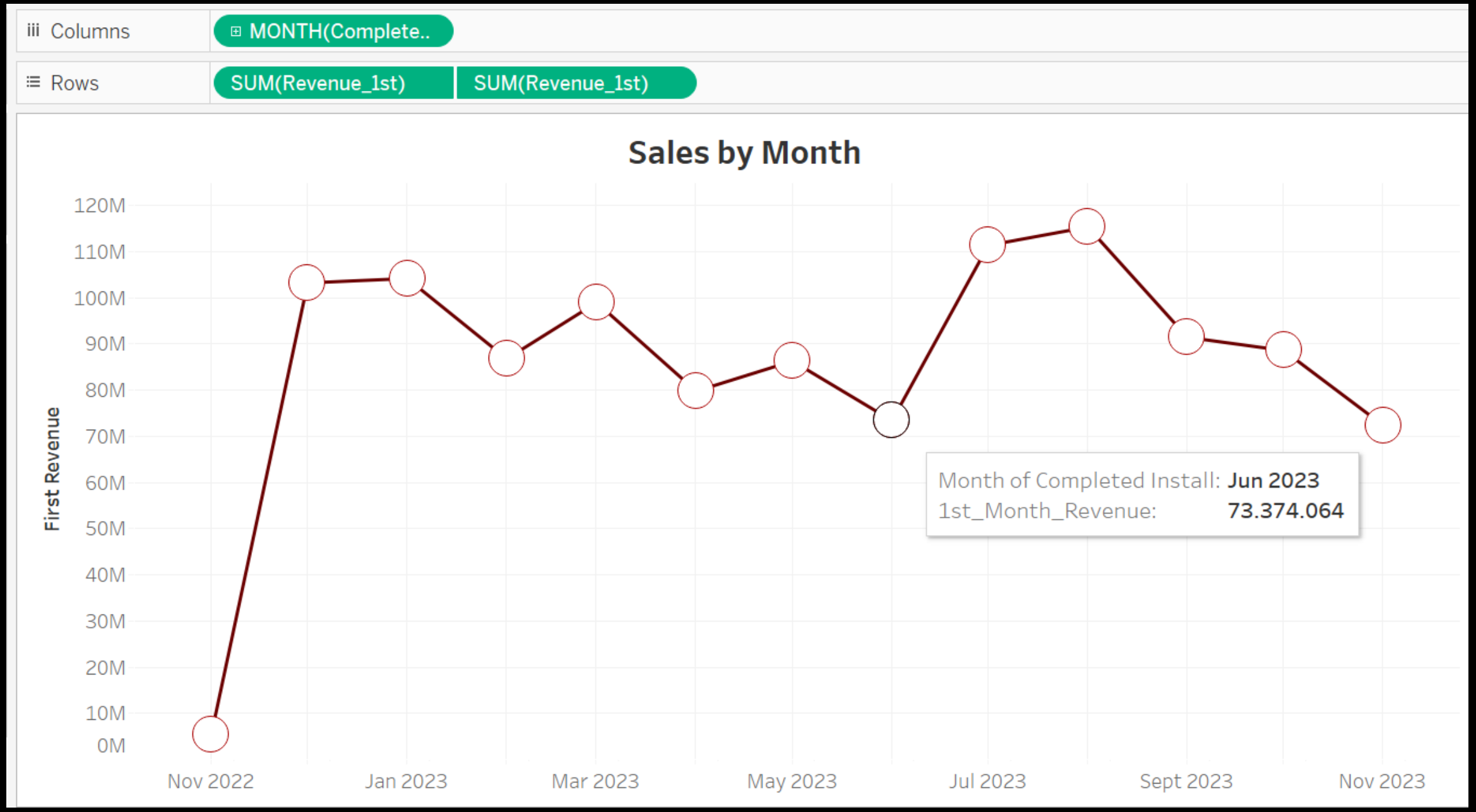
TABLEAU

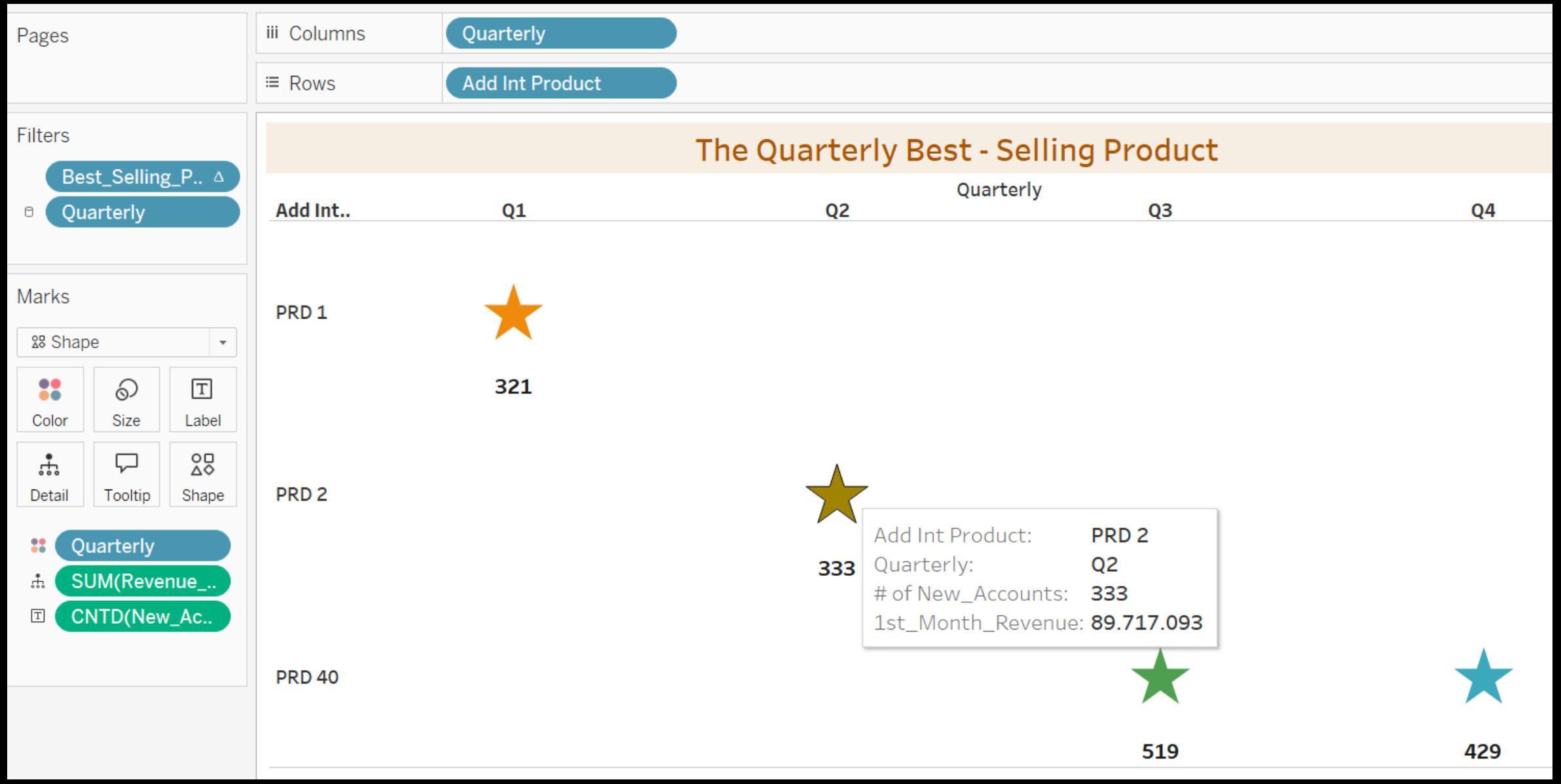

Pages
Columns
Measure Names
Rows
Filters
Measure Names
Quarterly
Marks
Automatic
Color
Size
Text
Detail
Tooltip
Measure Val..
Measure Values
CNTD(New_Account..
SUM(Revenue_1st)
SUM(First_comm)
KPIs
# of New Accounts
3.702
1st Month Revenue
1.116.870.467
1st Month Commission
1.317.346.159

Sales_by_City



City_size:	8
City:	Surabaya
Distinct count of New_Accounts:	518
Distinct count of Churn_account:	149
Distinct count of Sales Code:	12
Difference in Ratio_new_account_to_total_agent from the Previous along City:	51,80
1st_Month_Revenue:	143.389.397
2ND_Comm:	103.232.275,5


Columns
MONTH(Complete..
Rows
SUM(Revenue_1st)
SUM(Revenue_1st)
Sales by Month
First Revenue
120M
110M
100M
90M
80M
70M
60M
50M
40M
30M
20M
10M
0M
Nov 2022
Jan 2023
Mar 2023
May 2023
Jul 2023
Sept 2023
Nov 2023
Month of Completed Install: Jun 2023
1st_Month_Revenue: 73.374.064


Pages
Columns
Quarterly
Rows
Add Int Product
Filters
Best_Selling_P..
Quarterly
Marks
Shape
Color
Size
Label
Detail
Tooltip
Shape
Quarterly
SUM(Revenue_..
CNTD(New_Ac..
The Quarterly Best - Selling Product
Quarterly
Add Int..
Q1
Q2
Q3
Q4
PRD 1
321
PRD 2
333
Add Int Product: PRD 2
Quarterly: Q2
# of New_Accounts: 333
1st_Month_Revenue: 89.717.093
PRD 40
519
429

Pages

Columns

Rows

Filters

Best_selling_Rev

Quarterly

Marks

Automatic

Color | Size | Label | Detail | Tooltip

SUM(Revenue_1st)

SUM(Revenue_1st)

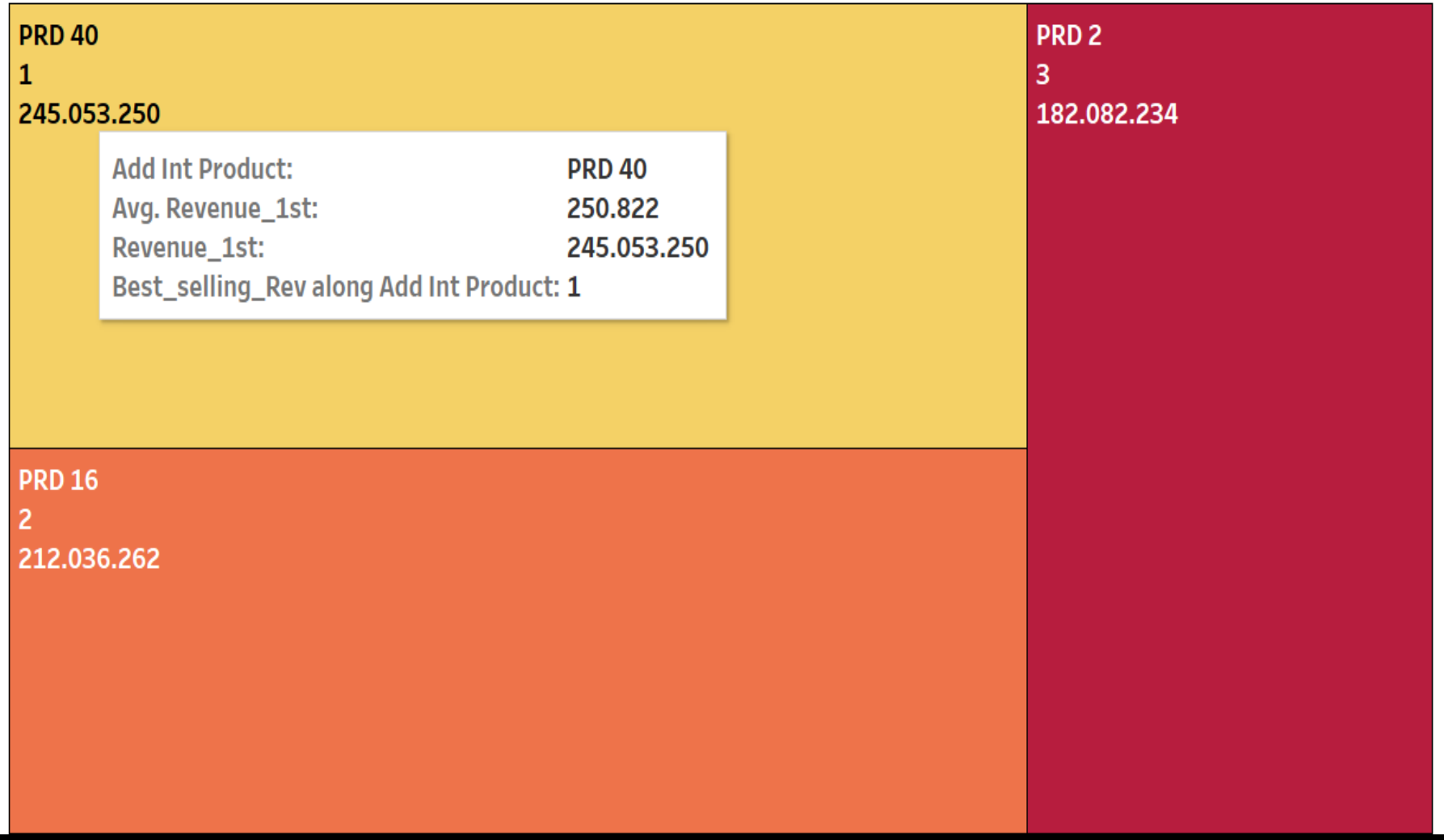
Add Int Product

Best_selling_Rev

SUM(Revenue_1st)

AVG(Revenue_1st)

The TOP 3 Revenue Generating Products



Filters

Quarterly

Marks

Automatic

Color | Size | Label

Detail | Tooltip | Shape

AGG(Profit_margin_size)

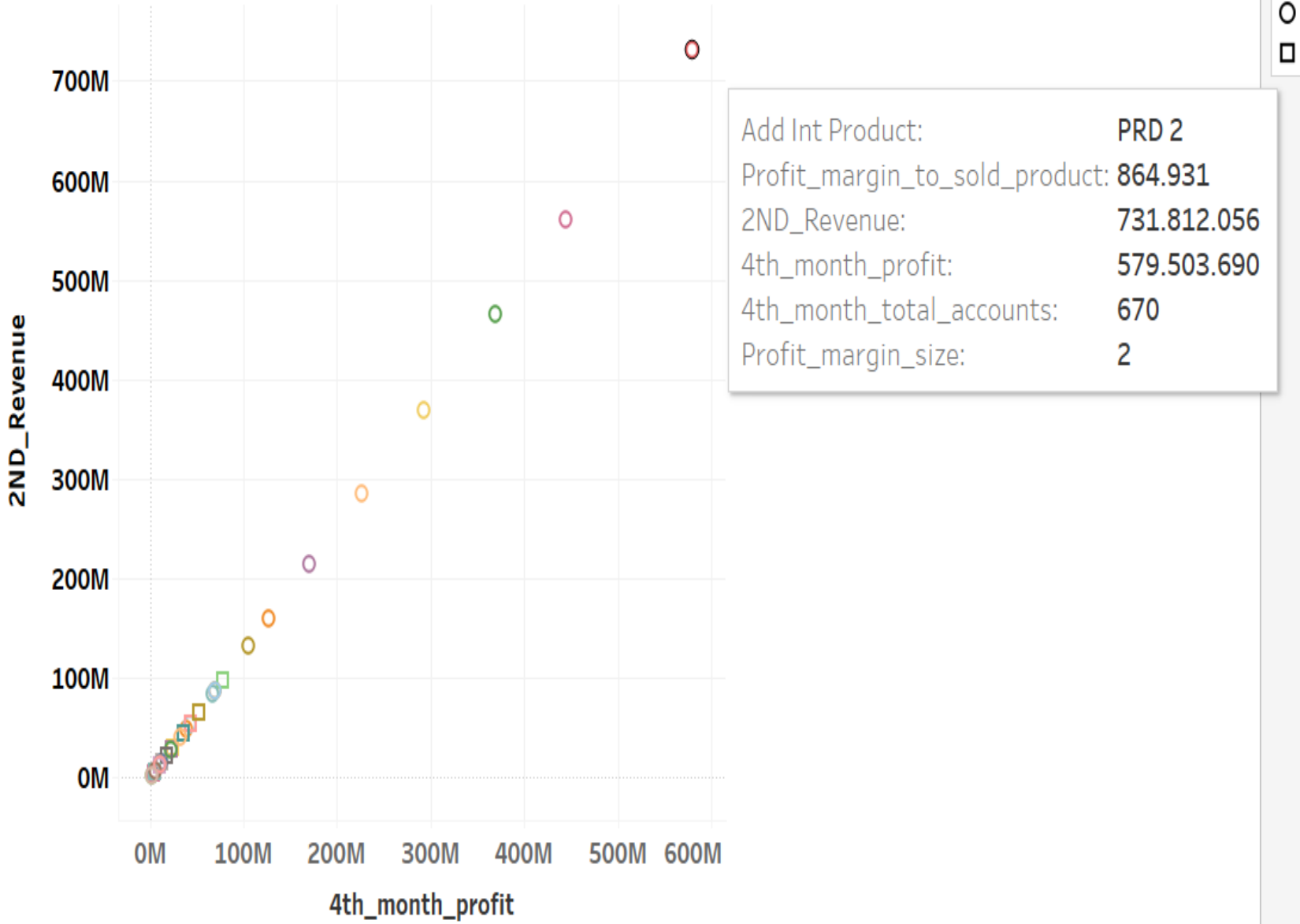
AGG(Profit_margin_to_sold_p..

Add Int Product

Add Int Product

AGG(4th_month_total_accou..

4th-Month Profit Margin by Product and its correlation to Revenue



Add Int Product:	PRD 2
Profit_margin_to_sold_product:	864.931
2ND_Revenue:	731.812.056
4th_month_profit:	579.503.690
4th_month_total_accounts:	670
Profit_margin_size:	2

AGG(Profit_..

2

7

Pages

Columns CNTD(New_Accounts)

Rows Add Int Product

Filters

YEAR(Completed Ins..

Add Int Product

Quarterly

Marks

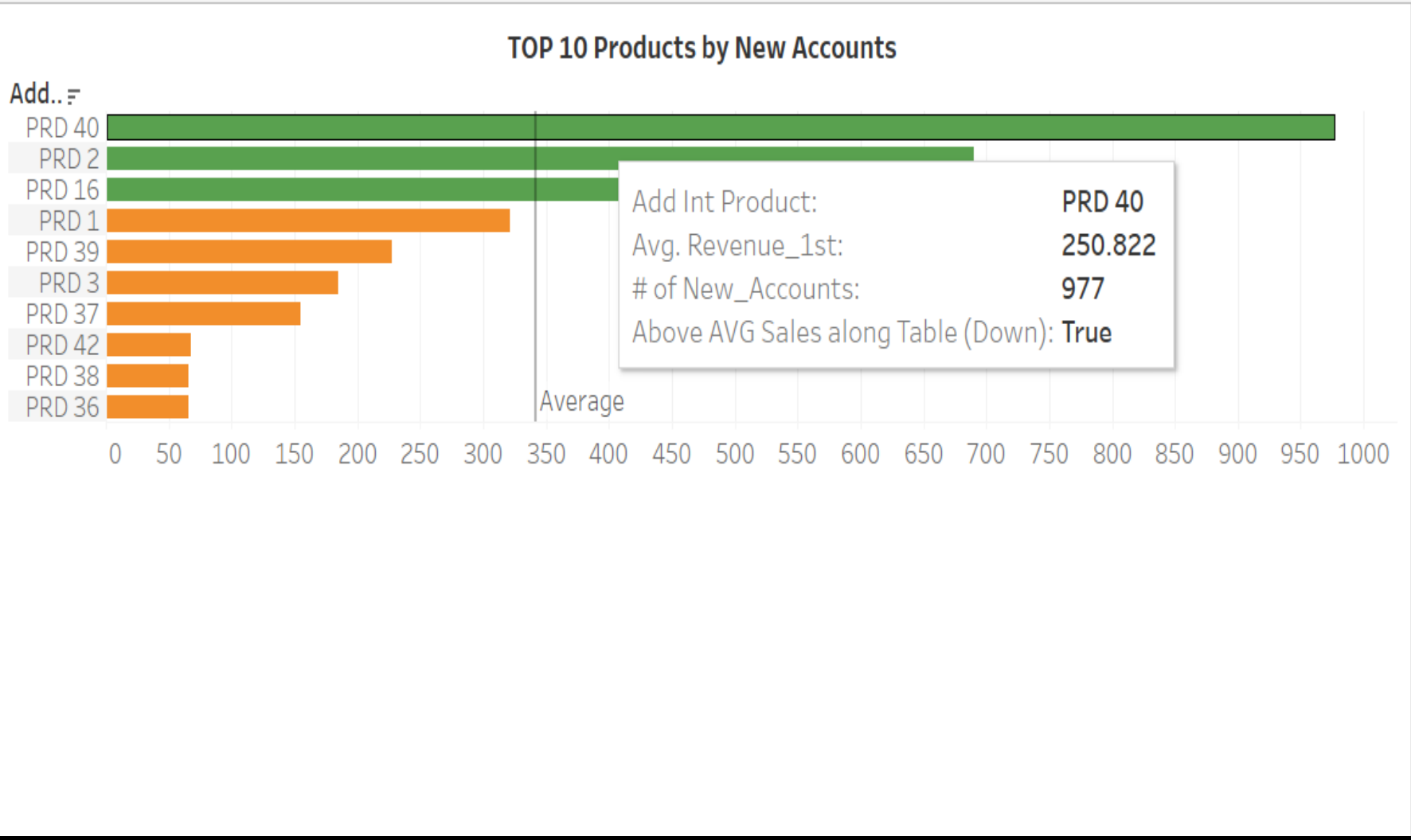
Automatic

Color Size Label

Detail Tooltip

Above AVG Sales

AVG(Revenue_1st)



AGG(Above ...

False

True

Pages

Columns CNTD(New_Accounts)

Rows Add Int Product

Filters

YEAR(Completed..

Add Int Product

Quarterly

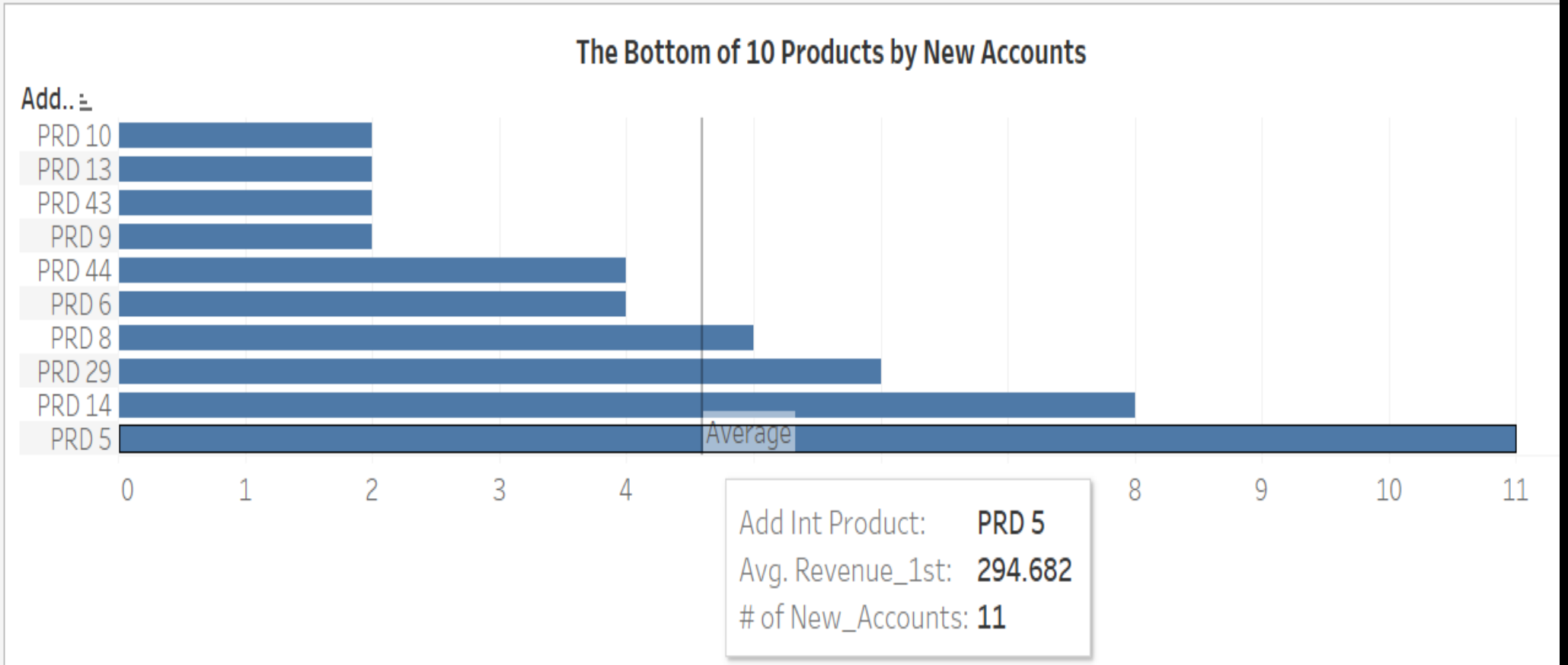
Marks

Automatic

Color Size Label

Detail Tooltip

AVG(Revenue_..



Pages

Columns: Department

Rows: SUM(1ST_Revenue)

Filters

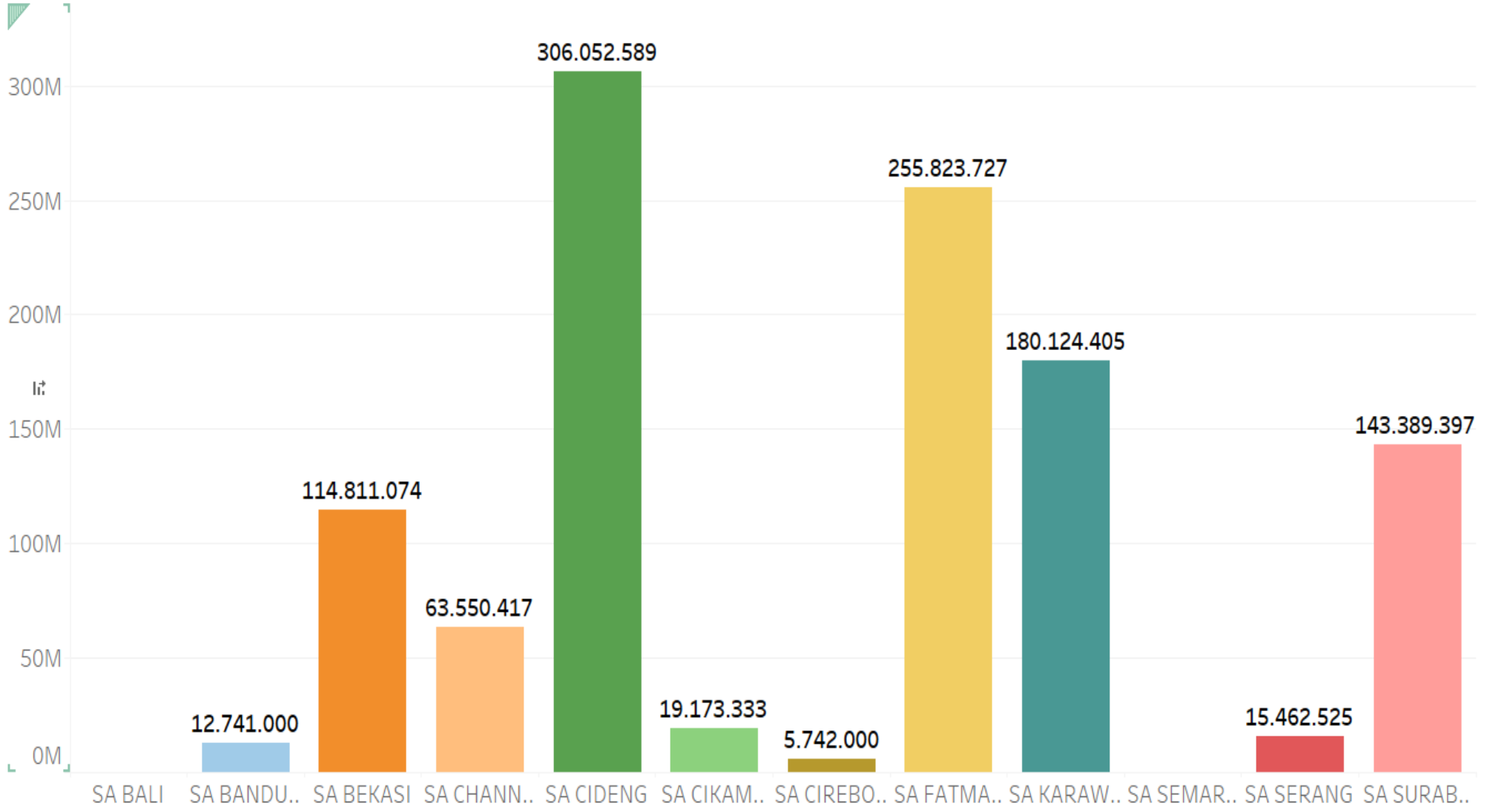
Marks

Automatic

Color | Size | Label | Detail | Tooltip

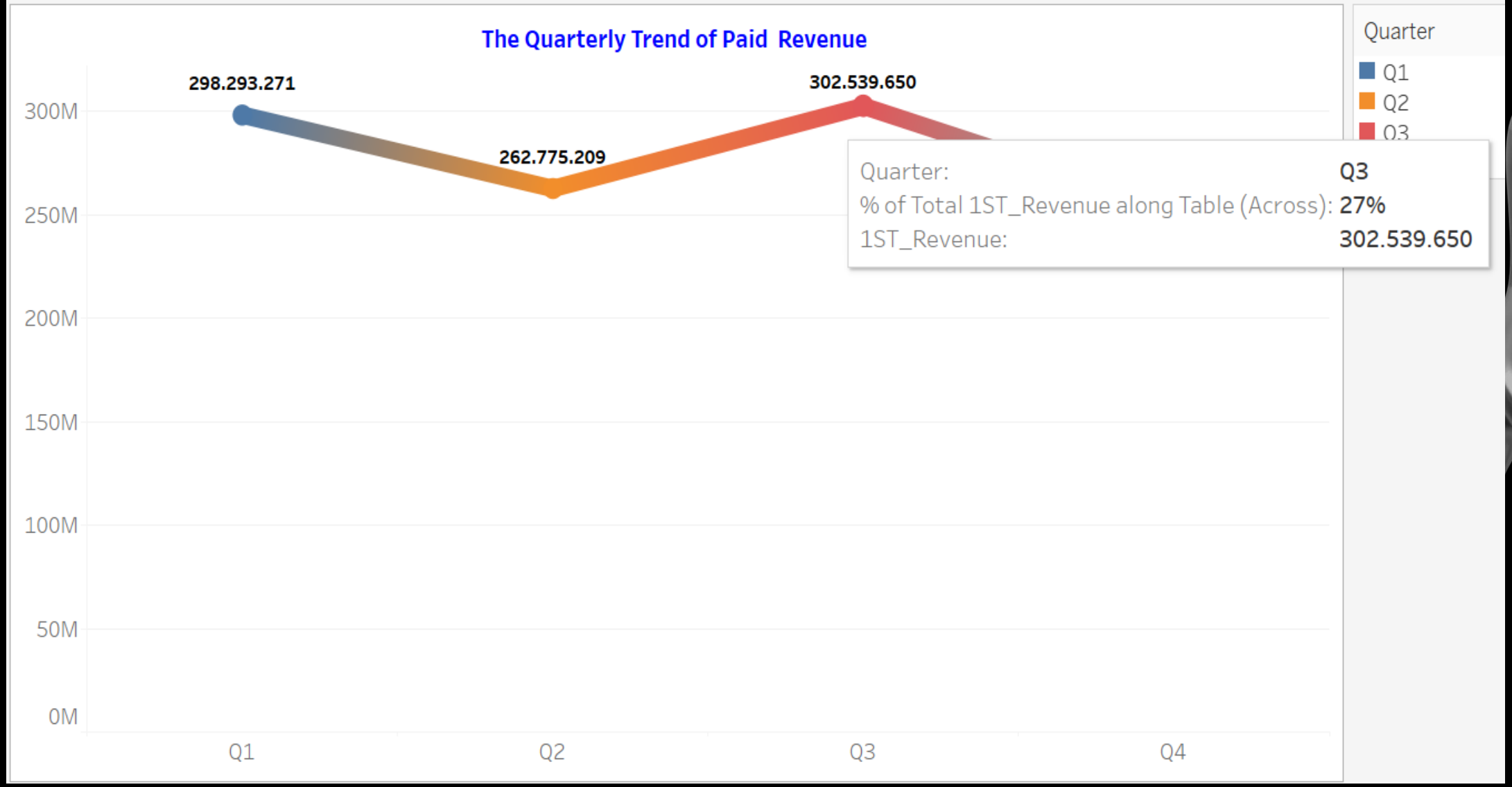
Department

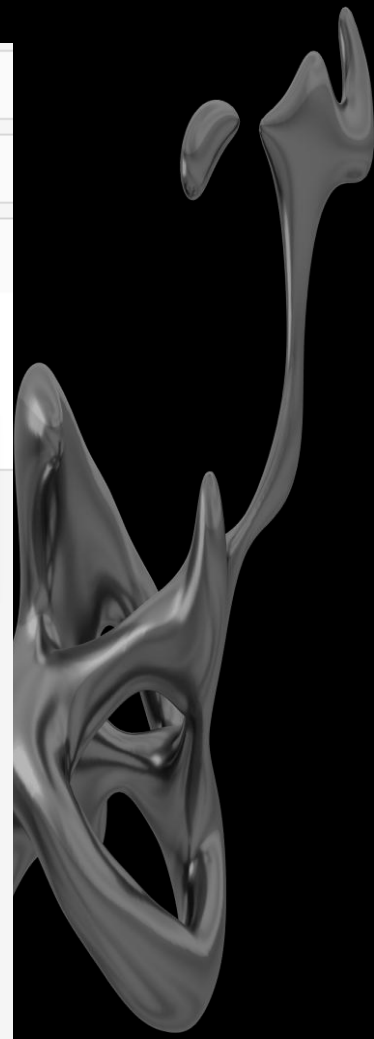
SUM(1ST_Reve..



Columns: Quarter

Rows: SUM(1ST_Revenue)





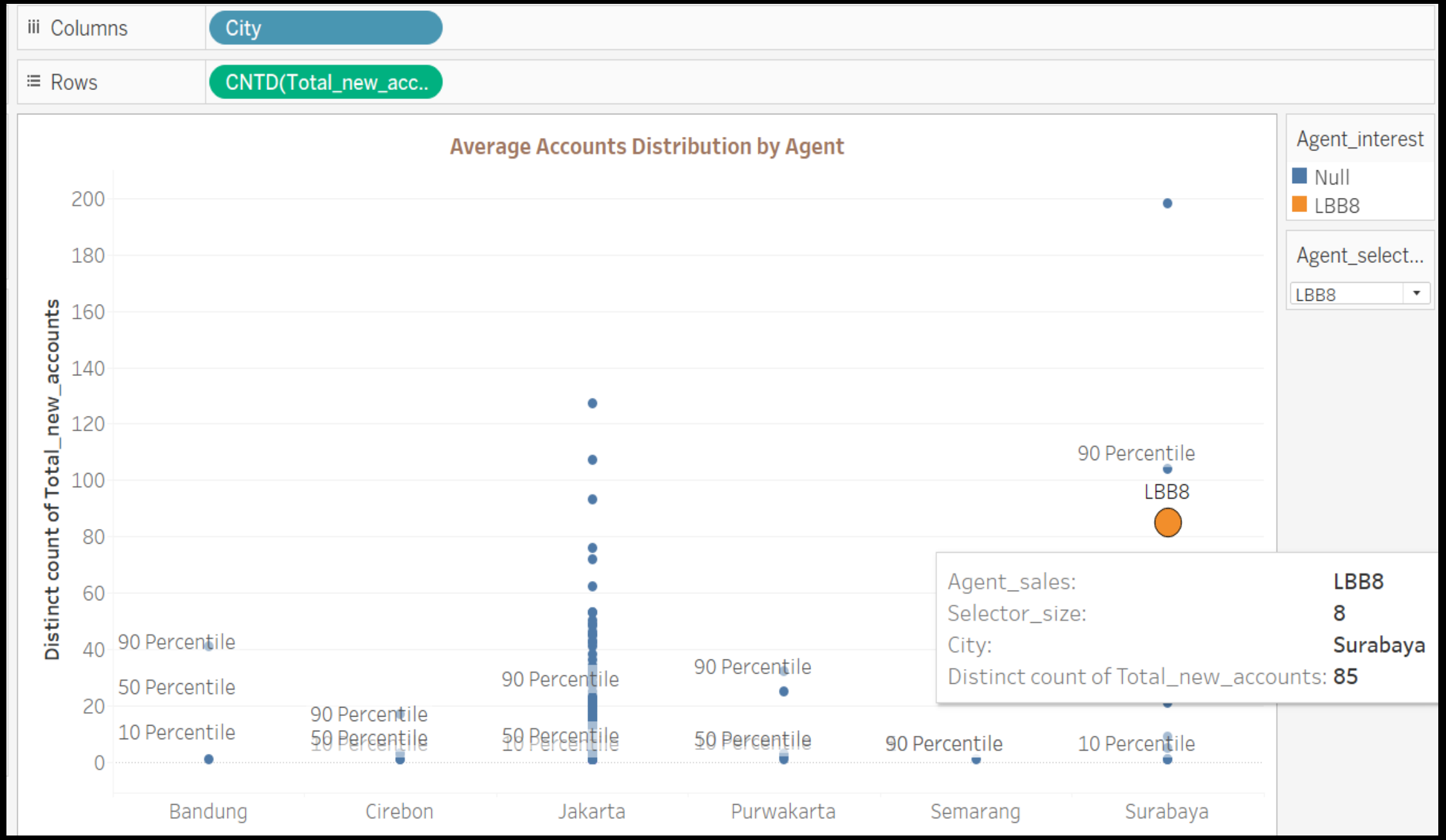
Columns: Measure Names

Rows: Month to month

The Monthly Recurring Commission

Month ..	1st_comm	2nd_comm	3rd_comm
1	127.034.914	31.608.798	83.351.733
2	127.780.093	31.422.538	43.537.835
3	102.676.193	54.900.005	32.157.833
4	115.747.217	51.900.216	26.255.394
5	95.126.448	51.437.226	25.685.605
6	96.281.113	43.219.374	42.218.480
7	87.924.458	54.193.778	47.269.467
8	132.082.290	39.011.973	45.682.156
9	134.389.144	42.349.873	38.839.869
10	106.452.210	38.831.627	46.855.846
11	104.904.866	58.579.253	33.844.923
12	86.947.213	36.565.941	61.876.712

Month to month: 2
1st_comm: 127.780.093


Columns
City
Rows
CNTD(Total_new_acc..
Average Accounts Distribution by Agent
Distinct count of Total_new_accounts
200
180
160
140
120
100
80
60
40
20
0
90 Percentile
50 Percentile
10 Percentile
LBB8
Bandung
Cirebon
Jakarta
Purwakarta
Semarang
Surabaya
Agent_interest
Null
LBB8
Agent_select...
LBB8
Agent_sales: LBB8
Selector_size: 8
City: Surabaya
Distinct count of Total_new_accounts: 85

Pages

Columns: Monthly

Rows: CNT(Churn_account)

Filters

Quarterly

Marks

Line

Color | Size | Label

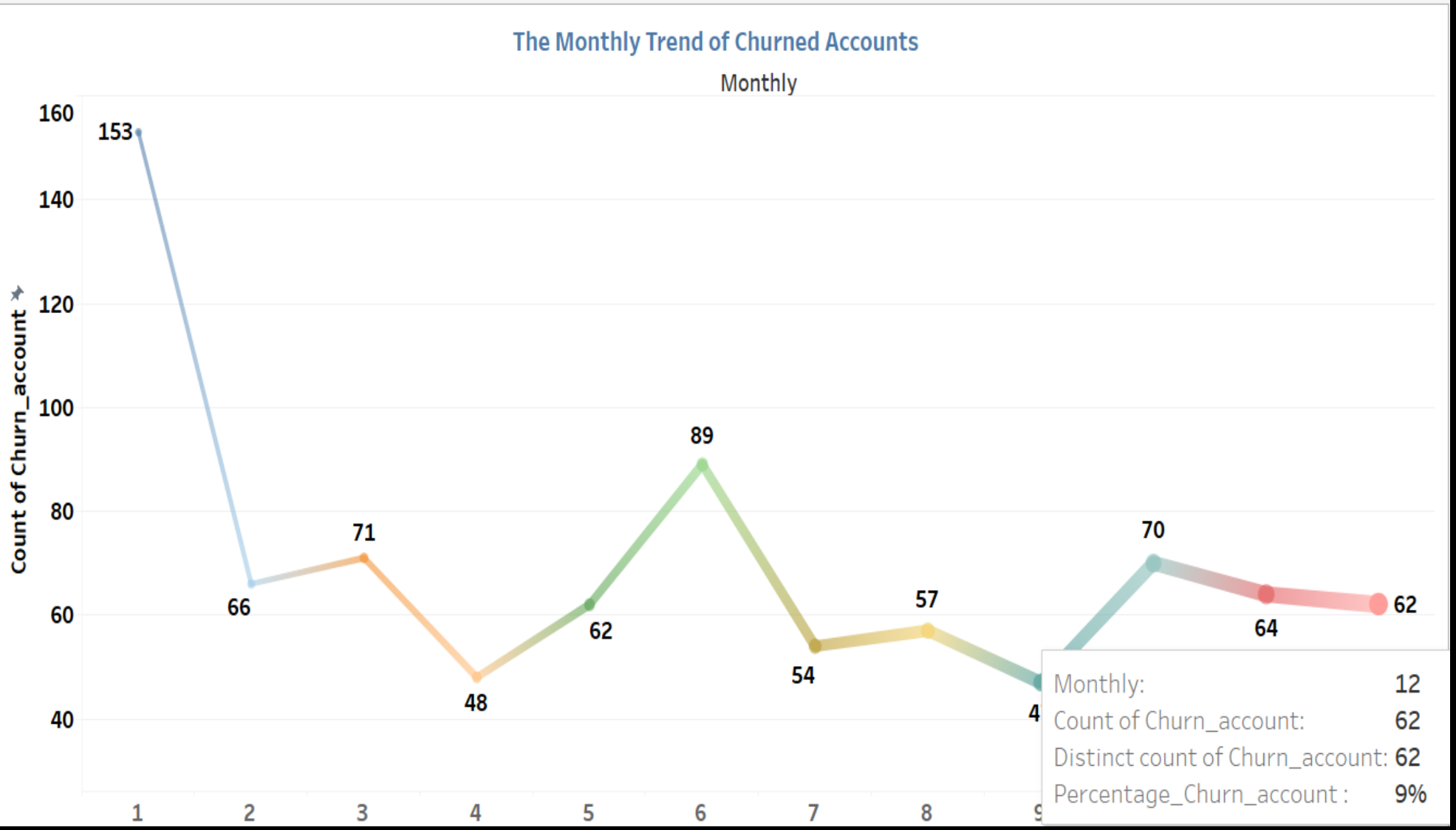
Detail | Tooltip | Path

Monthly

Monthly

CNTD(Churn_a..

AGG(Percenta..



Pages

Columns: ISOWEEK(Churn D..

Rows: CNT(Churn_account)

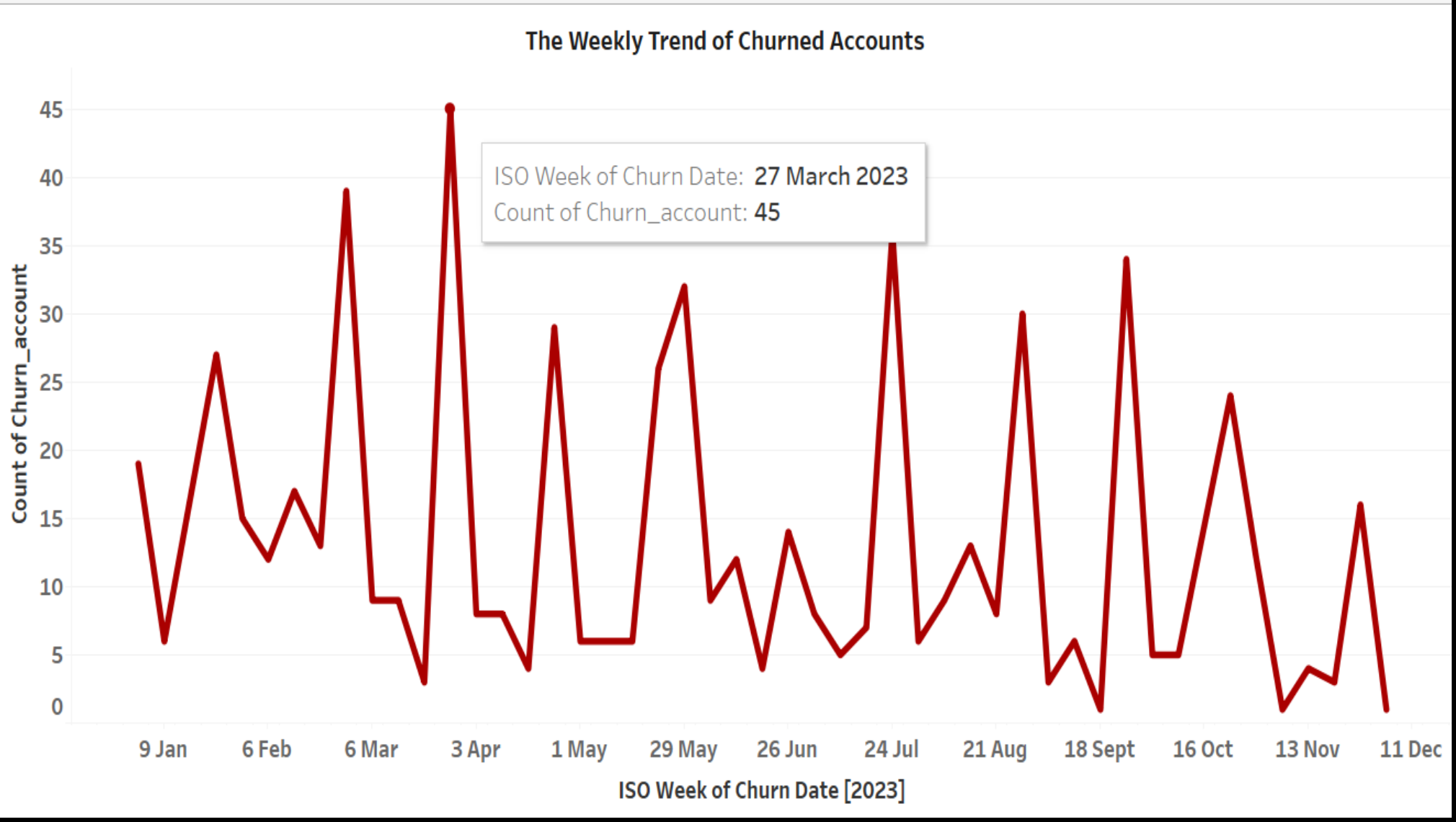
Filters

- YEAR(Churn Dat..
- ISOWEEK(Churn ..
- Quarterly

Marks

Automatic

Color | Size | Label | Detail | Tooltip | Path



Filters

Quarterly

Marks

Pie

Color | Size | Label

Detail | Tooltip | Angle

Department

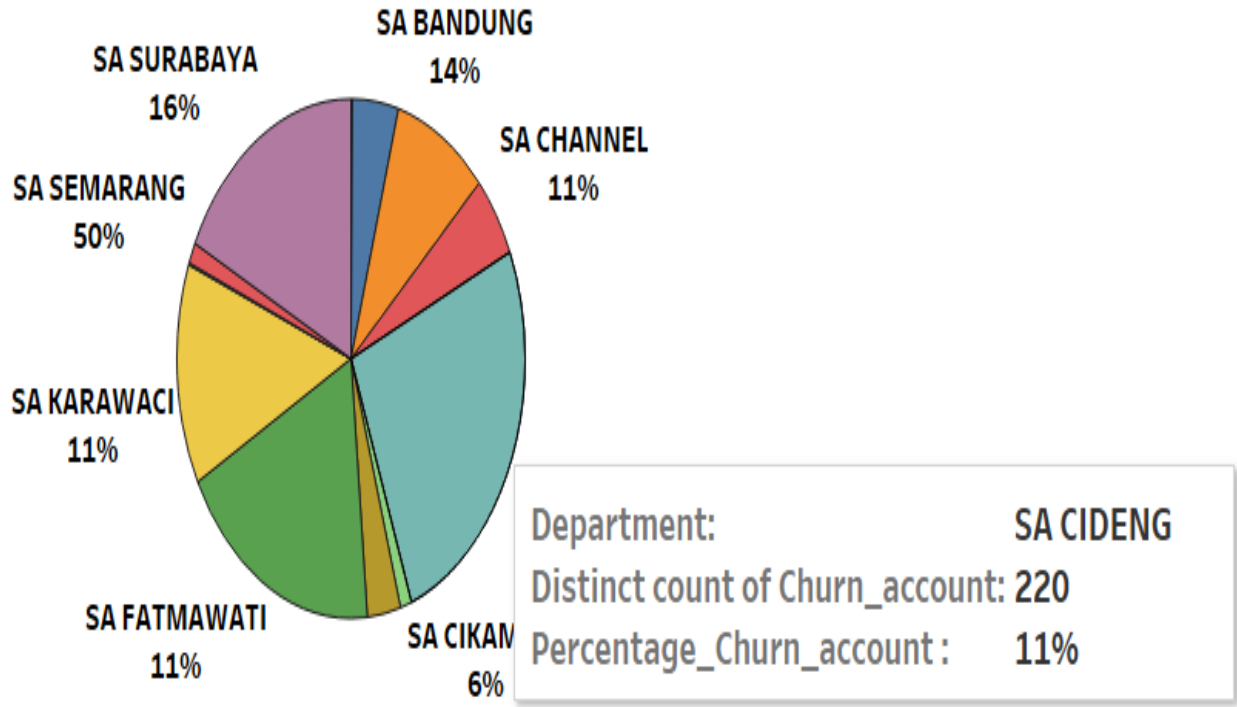
CNTD(Churn_a..

CNTD(Churn_a..

Department

AGG(Percentag..

The Churned Accounts by Department

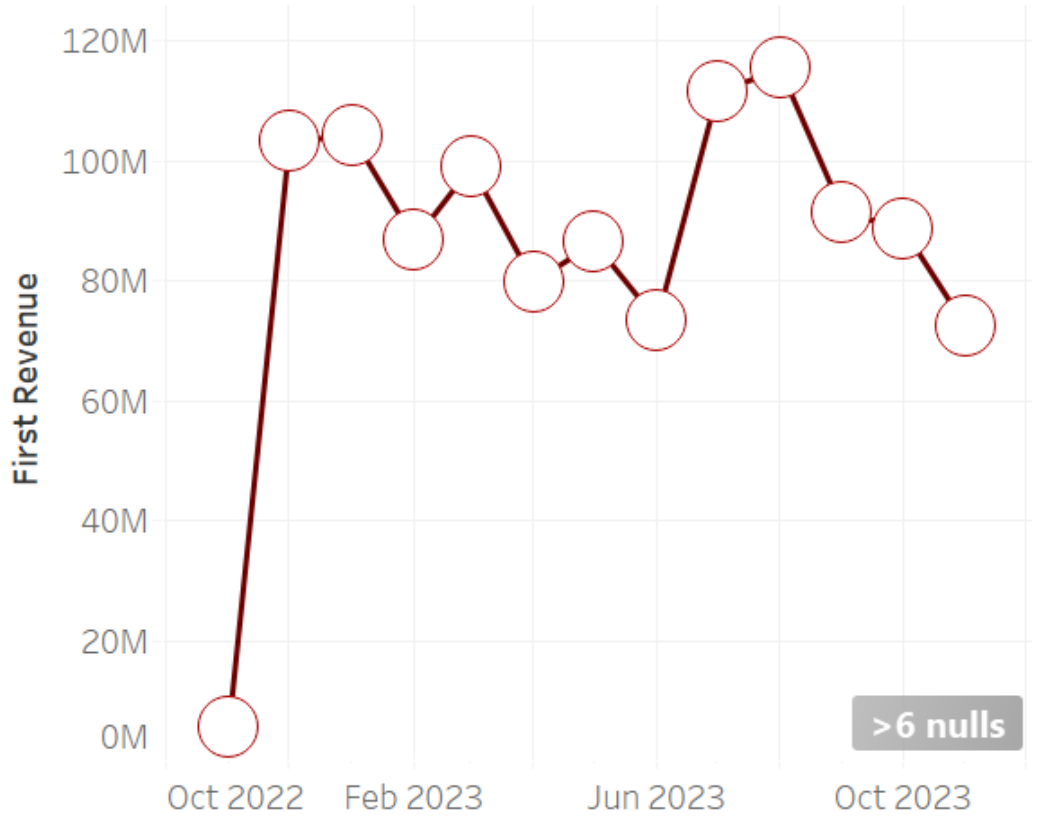


AREA SALES DASHBOARD

Quarte.. (All)

# of New Accounts	1st Month Revenue	1st Month Commission
3.702	1.116.870.467	1.317.346.159

Sales by Month

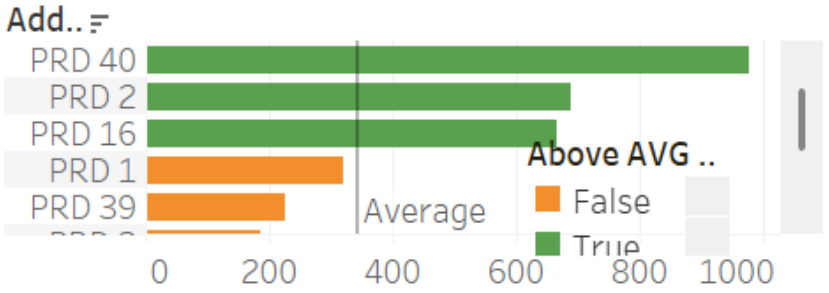


Sales_by_City

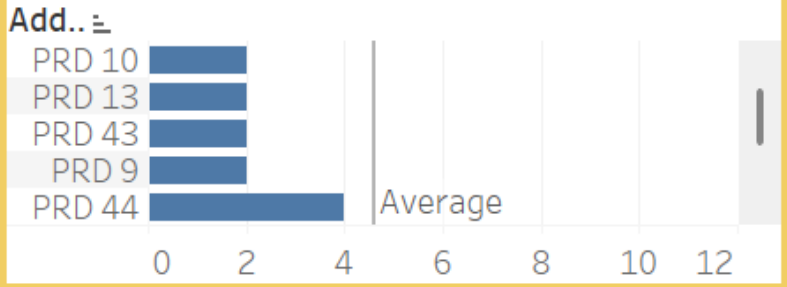
Although Jakarta recorded the highest number of new accounts(3055),
with over 300 sales agents, its productivity was not proportional. In contrast, Surabaya achieved 518 new accounts with only around 11-12 agents, resulting in a much higher productivity score of 51. Meanwhile, the highest revenue was recorded in August and the lowest in November, with 72.343.666

Sales by Product Dashboard

TOP 10 Products by New Accounts

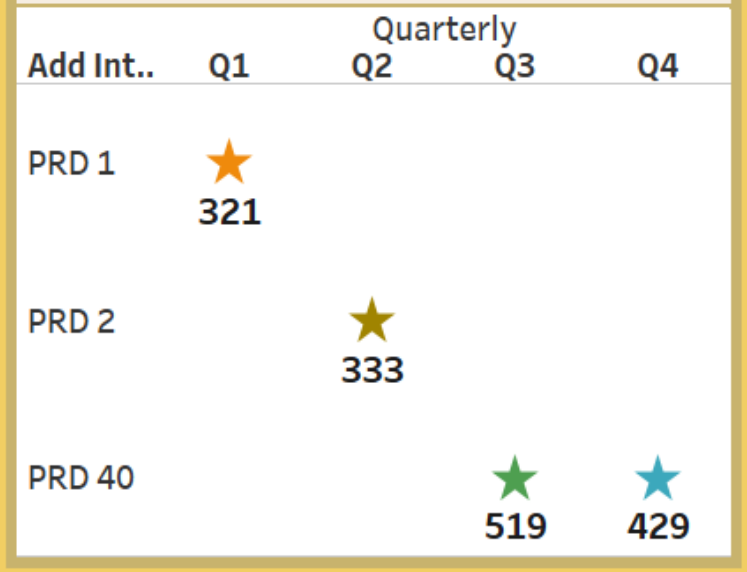


The Bottom of 10 Products by New Accounts

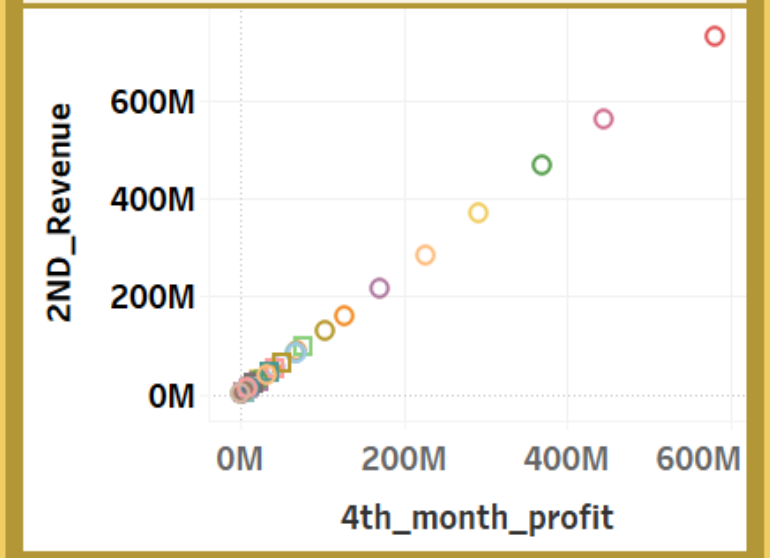


Product 2 dominates nearly all segments, with new accounts and revenue, the best-selling product in Q3, and the highest profit margin. The bottom of some products have an average revenue of more than 250k, it takes further customer segmentation and marketing strategy to delve into the cause of this product to be a favourite

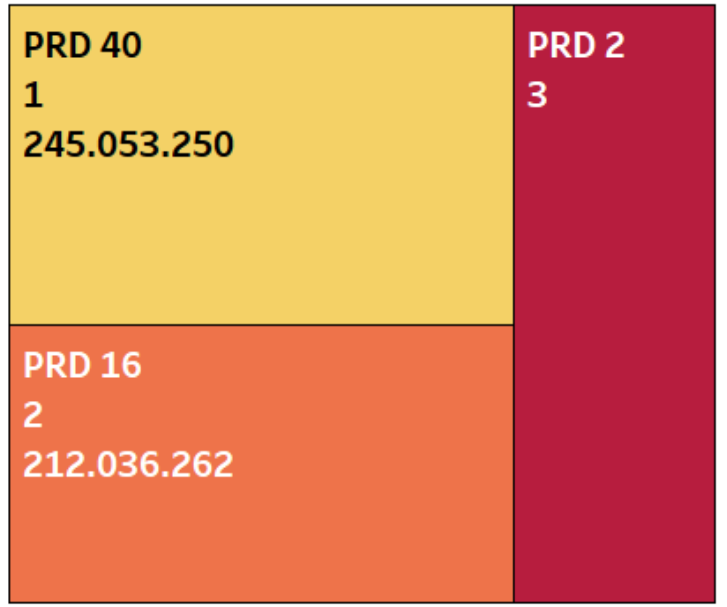
The Quarterly Best - Selling Product



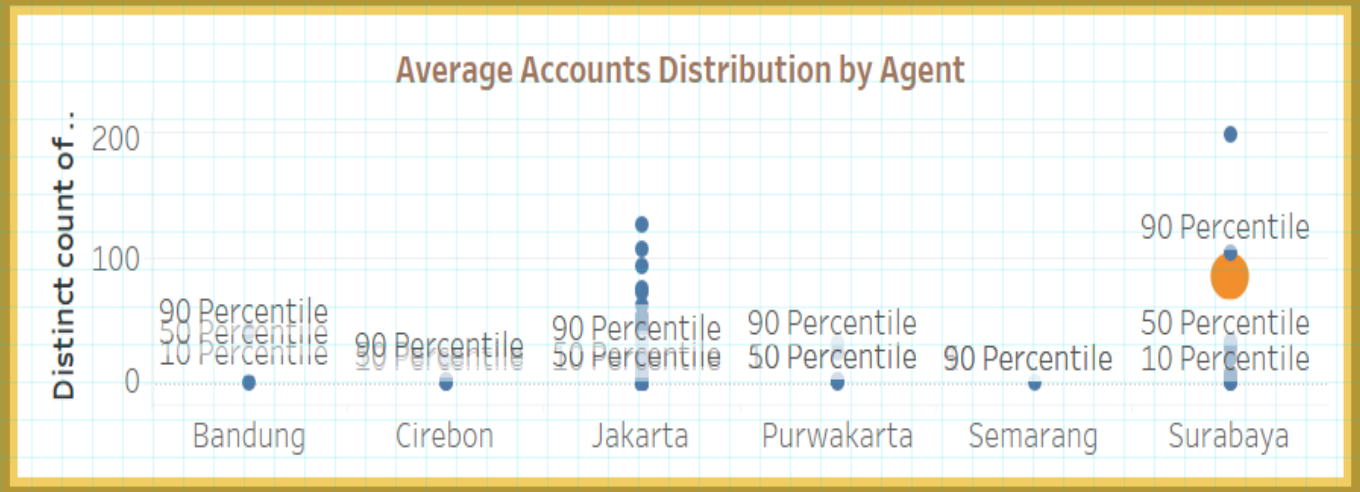
4th-Month Profit Margin by Product and its correlation to Revenue



The TOP 3 Revenue Generating Products



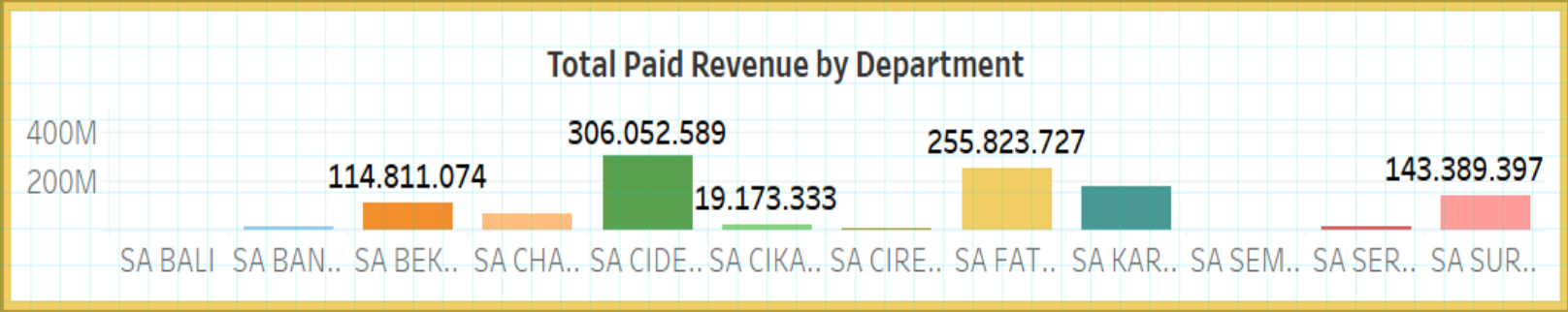
Revenue - New Accounts - Commission Sales Dashboard

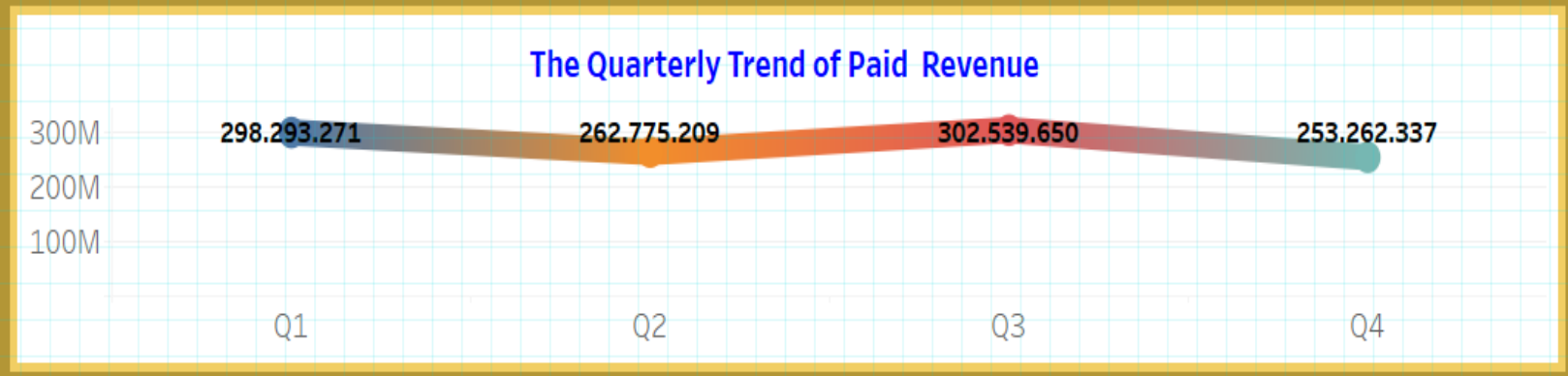


Agent.. LBB8

Agent_interest
- Null
- LBB8

SA Cideng has the highest revenue. Furthermore, a few of the sales agents in Jakarta have an average of more than percentile- 90 of the account distribution, while Surabaya is served by a single agent, nevertheless, the commission amount consistently decline over the 1st, 4th, and 7th ..



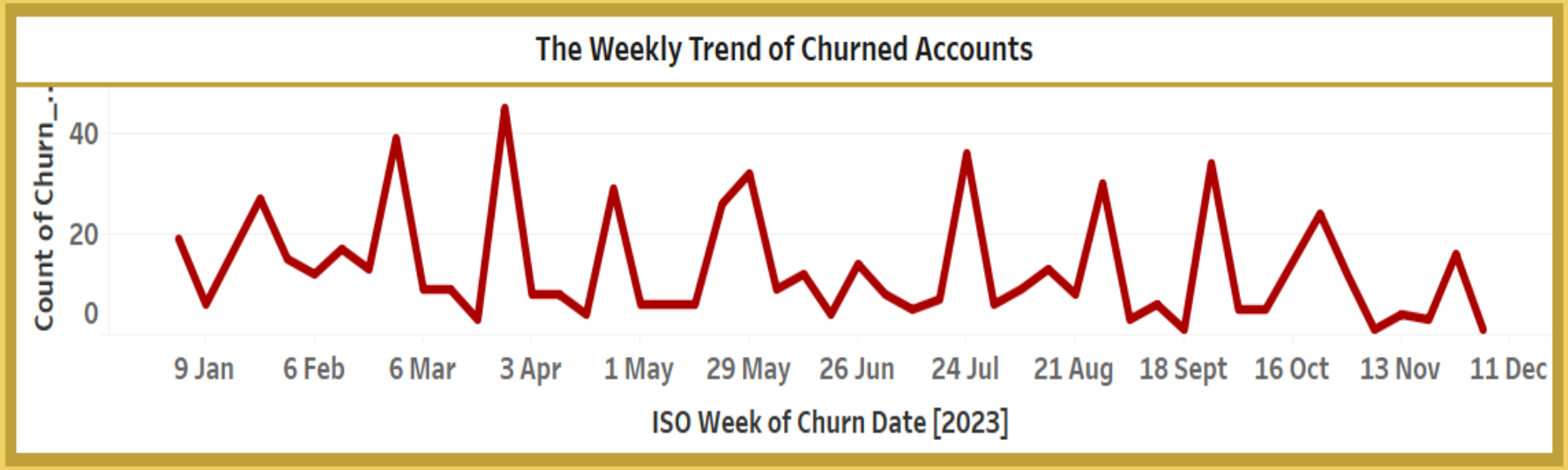


The Monthly Recurring Commission

Month ..	1st_comm	2nd_comm	3rd_comm
1	127.034.914	31.608.798	83.351.733
2	127.780.093	31.422.538	43.537.835
3	102.676.193	54.900.005	32.157.833
4	115.747.217	51.900.216	26.255.394
5	95.126.448	51.437.226	25.685.605
6	96.281.113	43.219.374	42.218.480
7	87.924.458	54.193.778	47.269.467
8	132.082.290	39.011.973	45.682.156
9	134.389.144	42.349.873	38.839.869
10	106.452.210	38.831.627	46.855.846
11	104.904.866	58.579.253	33.844.923
12	86.947.213	36.565.941	61.876.712

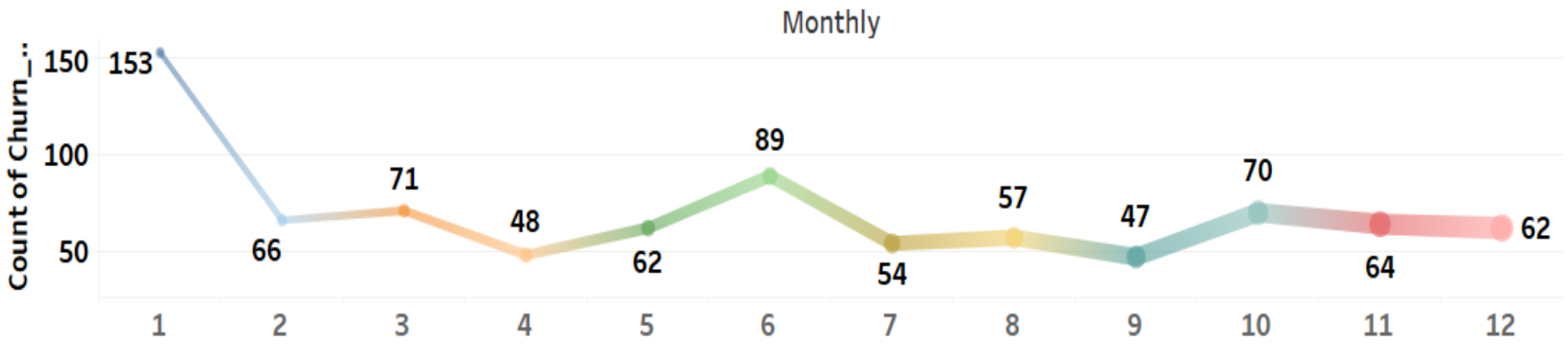
Churn Dashboard

The Weekly Trend of Churned Accounts

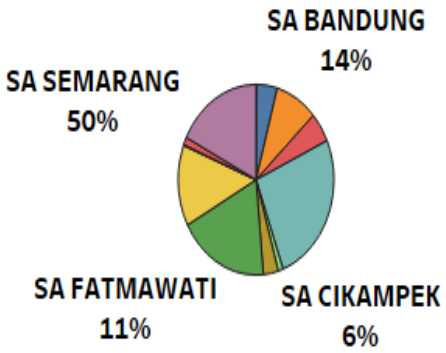


In the opening year, January had 153 churned accounts, following 89 accounts in June. The number of churned accounts rose steadily on a weekly basis during the period from the end of February through March, and SA Cideng had the biggest contribution to churned accounts

The Monthly Trend of Churned Accounts



The Churned Accounts by Department



CONCLUSION

The 200%-commission scheme over three periods is utterly ineffective in retaining subscribers, it only works to acquire new accounts. The equalization of selling skills shown in Surabaya, which outperformed Jakarta, requires further investigation. This may be due to competent leadership and management being significantly more effective than relying on a few well-skilled agents. For product development, focusing on long-term competitive products based on customer segmentation is essential for acquiring loyal customers.


TERIMA KASIH