Data Analytics Design for Product Sales Analysis with IBM Cognos

Title: Innovation Phase_3

Task: Perform Data Visualization

Introduction:

Data Analytics with Cognos Product Sales Analysis provides organizations with valuable insights into their sales performance. However, to enhance this analytical capability, incorporating machine learning algorithms is essential. This document explores how machine learning can be integrated to predict future sales trends and customer behaviors more accurately.

1. Problem Statement:

In traditional sales analysis, past data is used to make informed decisions about future sales and customer behaviors. While this approach is valuable, it is limited in its ability to adapt to dynamic market conditions and emerging trends. Machine learning algorithms offer the potential to predict future sales trends and customer behaviors more accurately, thereby empowering organizations to make proactive decisions.

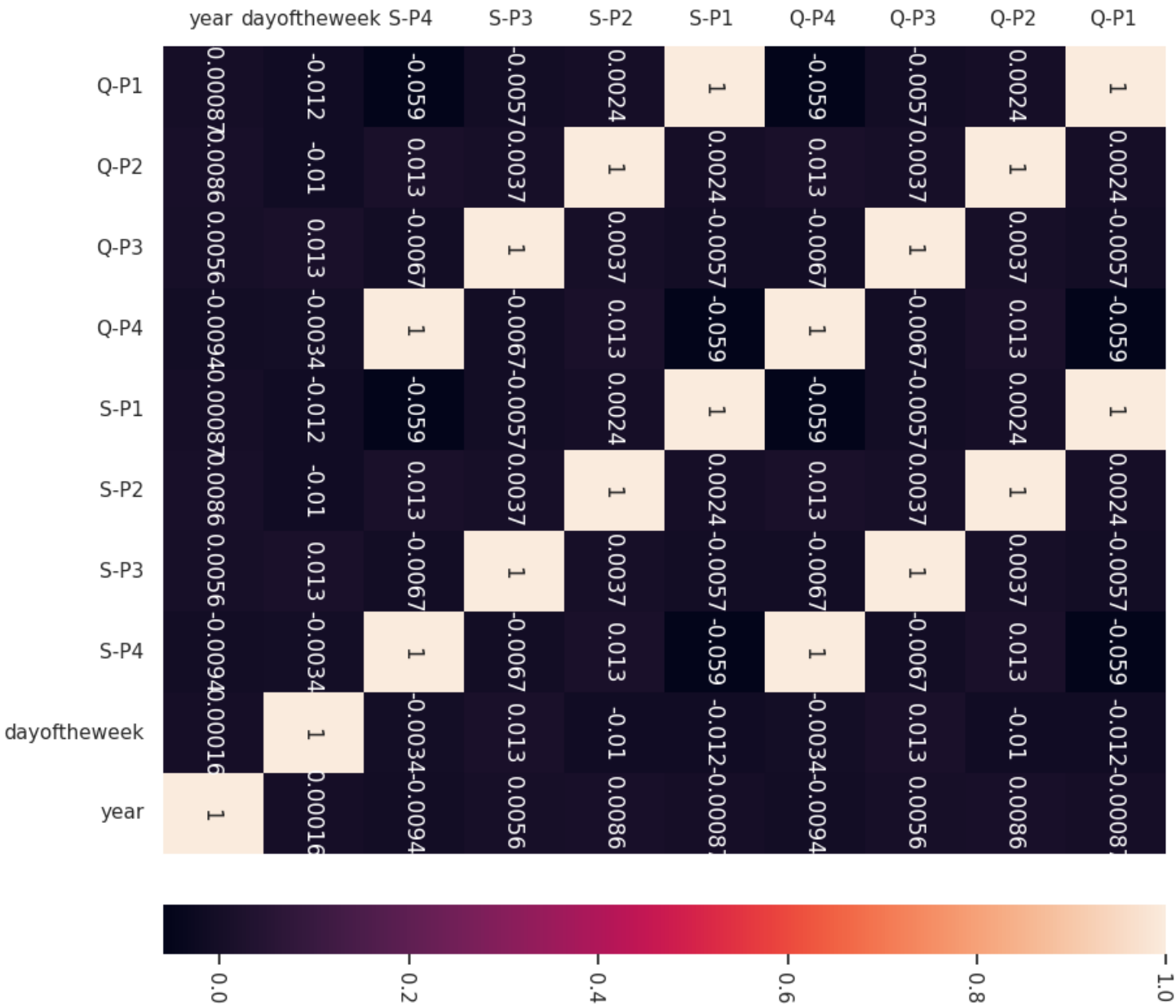
2. Data Visualization

Code and Outputs

1. Code

```
plt.figure(figsize=(10,10))
sns.heatmap(df.corr(),annot=True)
```

Out



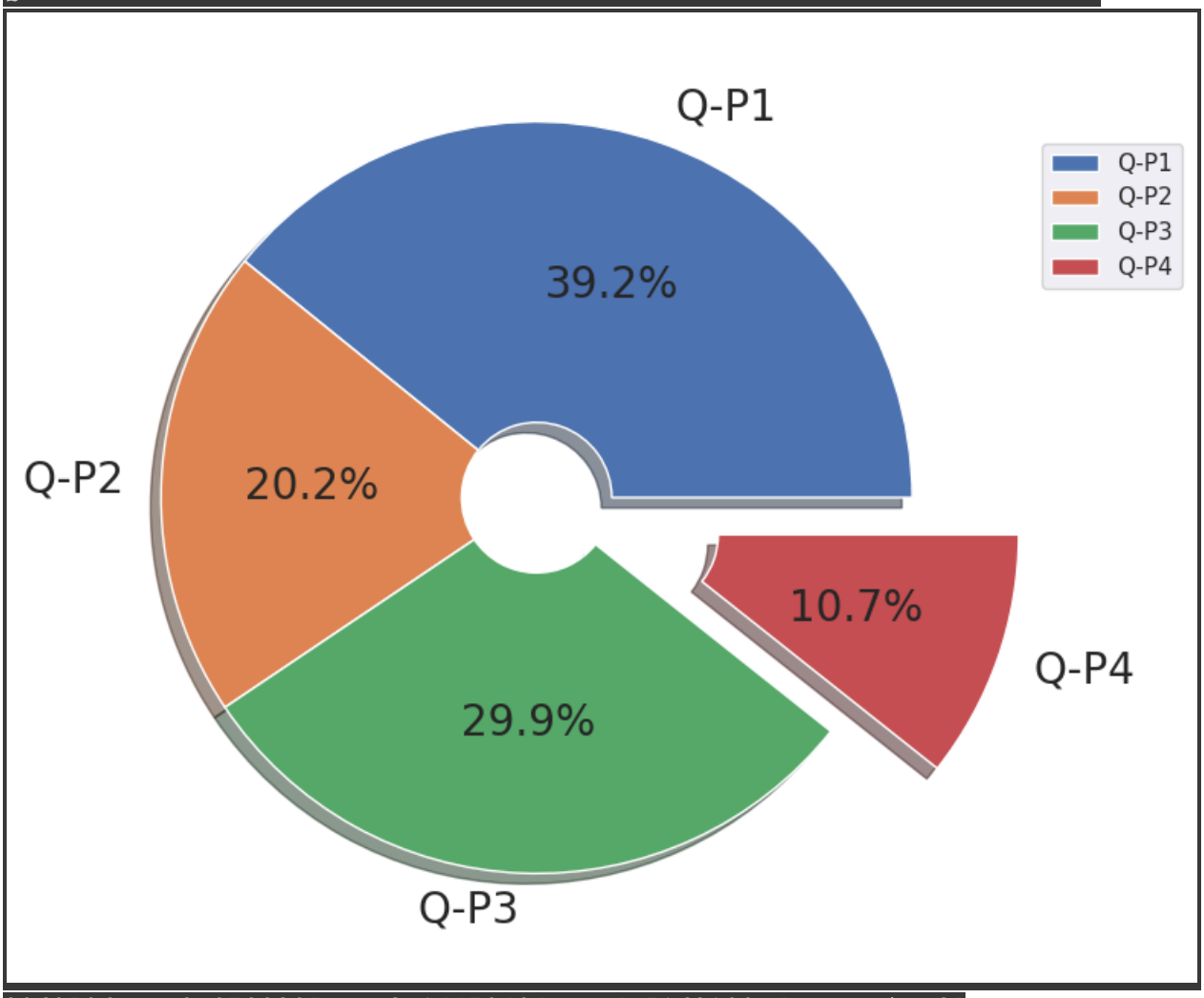
2. Code

```
q = df[["Q-P1","Q-P2","Q-P3","Q-P4"]].sum()

print(q)

plt.figure(figsize=(8,8))
```

```
plt.pie(q,labels=df[["Q-P1","Q-P2","Q-P3","Q-P4"]].sum().index,shadow=True,autopct="%0.01f%%",textprops={"fontsize":20},wedgeprops={'width': 0.8},explode=[0,0,0,0.3])

plt.legend(loc='center right', bbox_to_anchor=(1.2, 0.8));
```

Out

```
Q-P1                                                    1
```




```
8960506 Q-P2 9799295 Q-P3 14470404 Q-P4 5168100 dtype: int64
```

3. Code

```
s=df[["S-P1","S-P2","S-P3","S-P4"]].sum()

print(s)

plt.figure(figsize=(8,8))

plt.pie(s,labels=df[["S-P1","S-P2","S-P3","S-P4"]].sum().index,shadow=True,autopct="%0.01f%%",textprops={"fontsize":20},wedgeprops={'width': 0.8},explode=[0,0,0,0.3])
```

```
plt.legend(loc='center right', bbox_to_anchor=(1.2, 0.8))
```

Out

S-P1 60104804.02

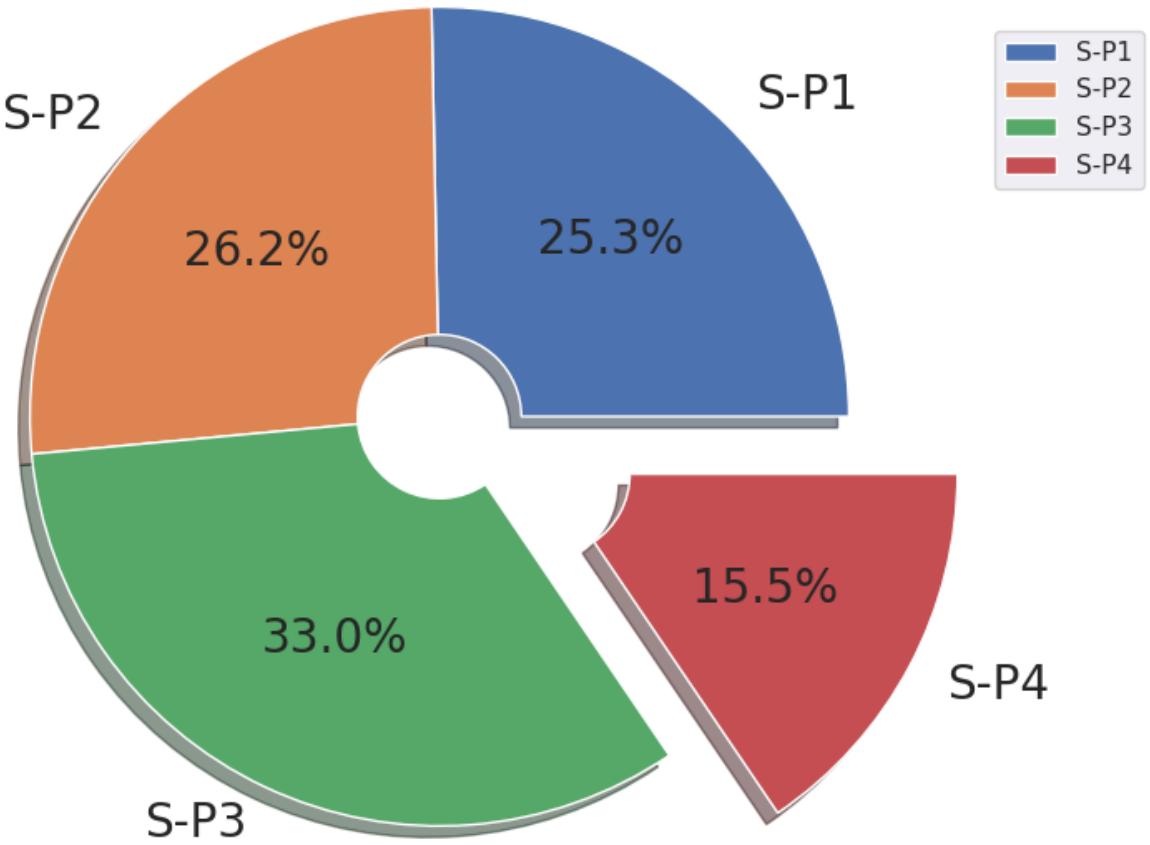
S-P2 62127530.30

S-P3 78429589.68

S-P4 36848553.00

dtype: float64

<matplotlib.legend.Legend at 0x79ead813ff10>

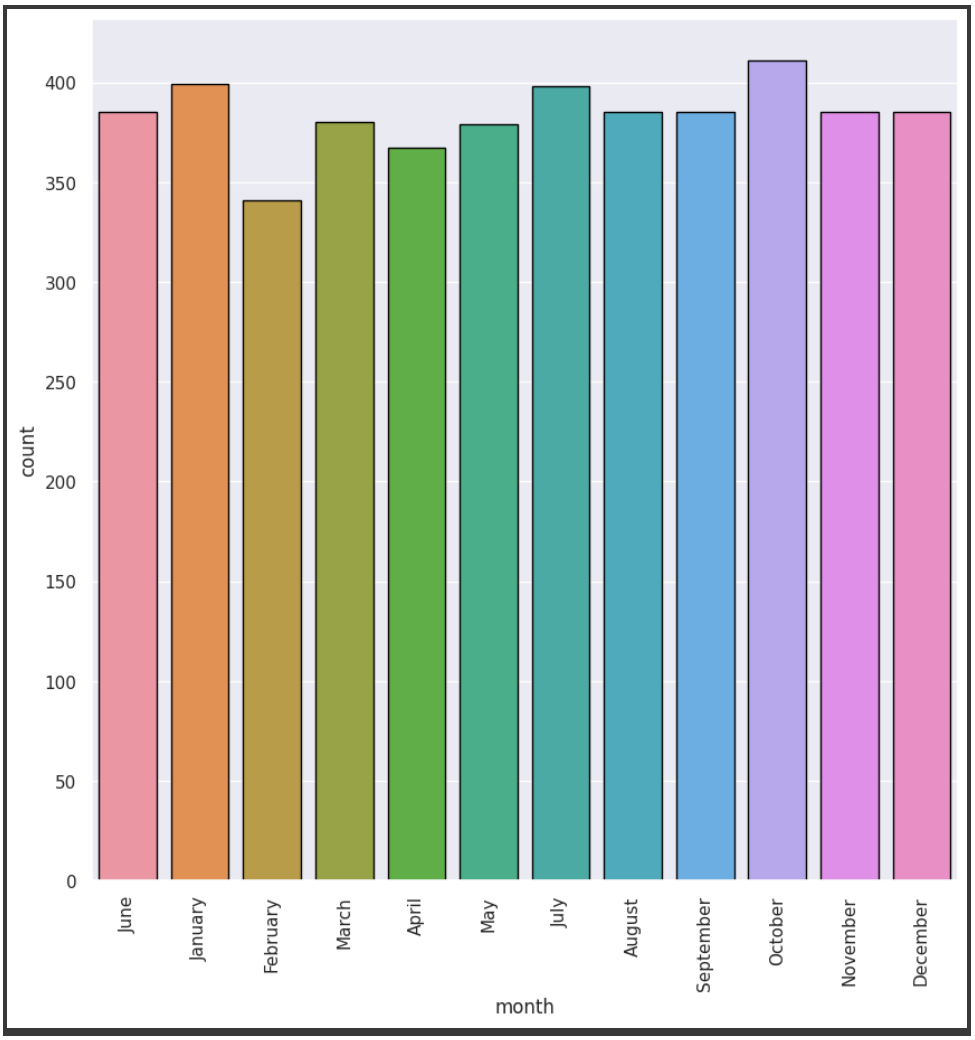


4. Code

```
print(df["month"].value_counts())

plt.figure(figsize=(10,10))

sns.countplot(x="month",data=df,edgecolor="black")

plt.xticks(rotation=90);
```

Out

```
October 411 January 399 July 398 June 385 August 385 September 385
November 385 December 385 March 380 May 379 April 367 February 341 Name:
month, dtype: int64
```

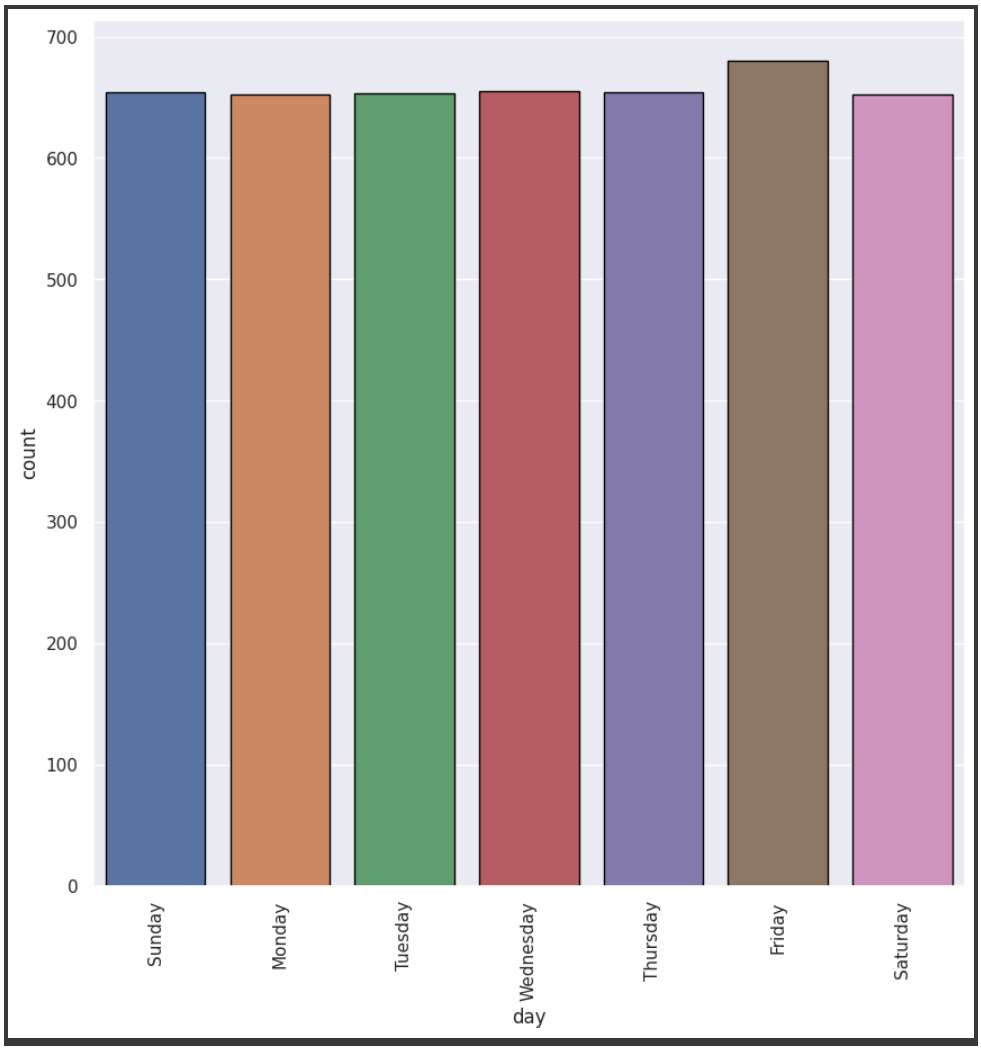


5. Code

```
print(df["day"].value_counts())

plt.figure(figsize=(10,10))

sns.countplot(x="day",data=df,edgecolor="black")

plt.xticks(rotation=90);
```

Out

```
Friday 680 Wednesday 655 Sunday 654 Thursday 654 Tuesday 653
Monday 652 Saturday 652 Name: day, dtype: int64
```

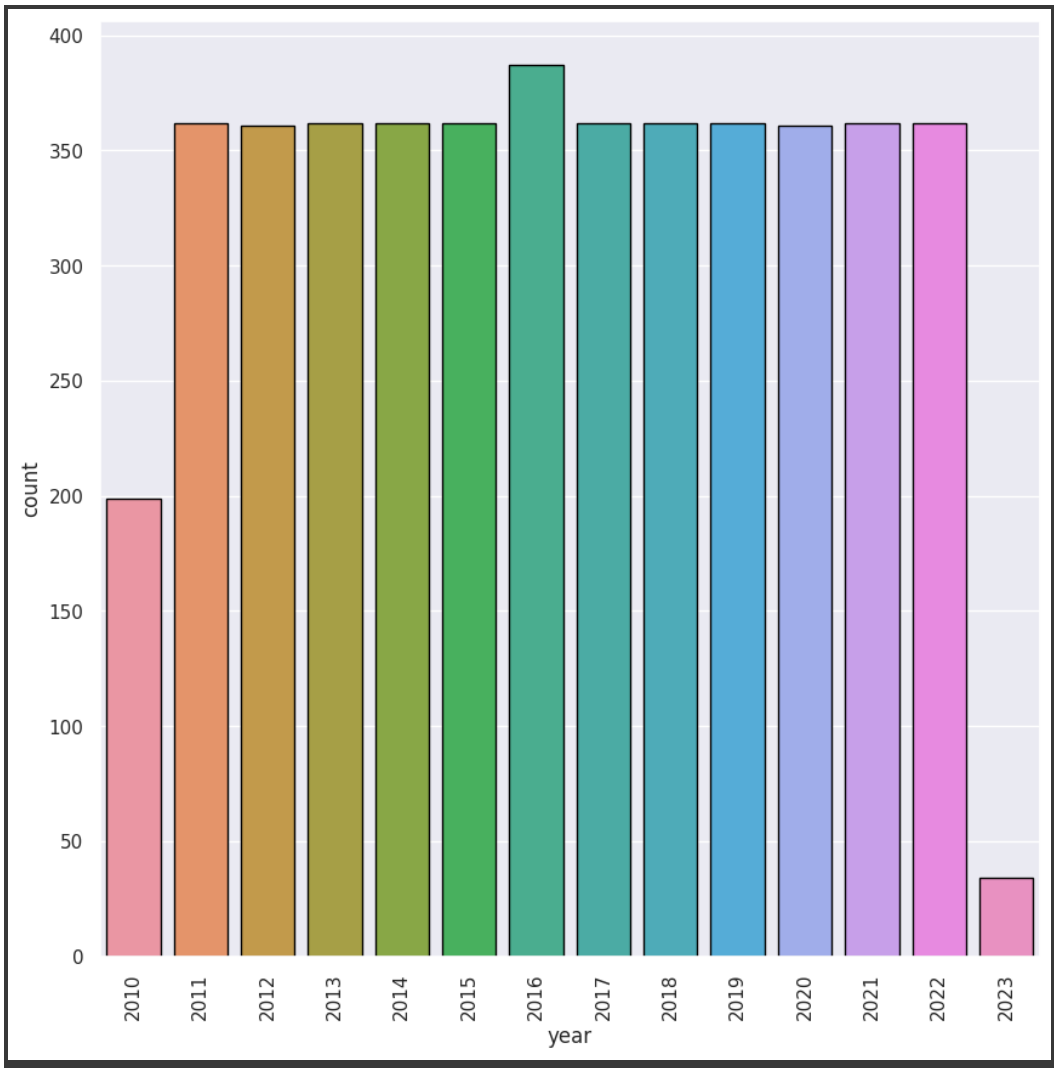


6. Code

```
print(df["year"].value_counts())

plt.figure(figsize=(10,10))

sns.countplot(x="year",data=df,edgecolor="black")

plt.xticks(rotation=90);
```

Out

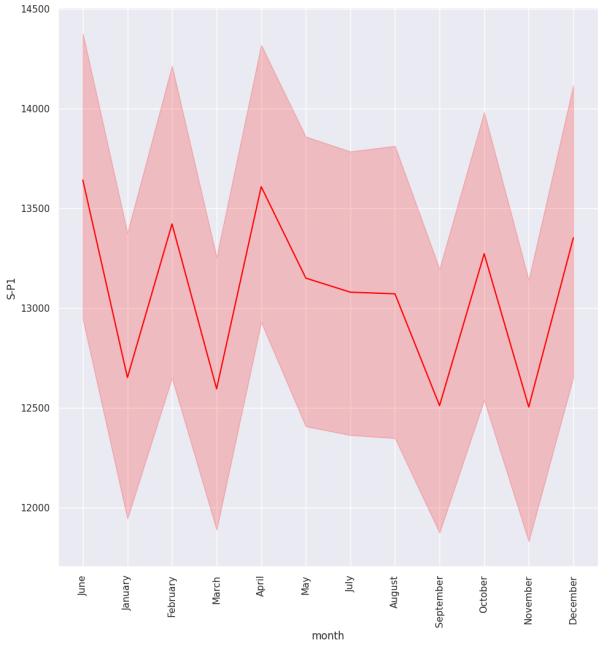
```
2016 387 2011 362 2013 362 2014 362 2015 362 2017 362 2018 362
2019 362 2021 362 2022 362 2012 361 2020 361 2010 199 2023 34
Name: year, dtype: int64
```

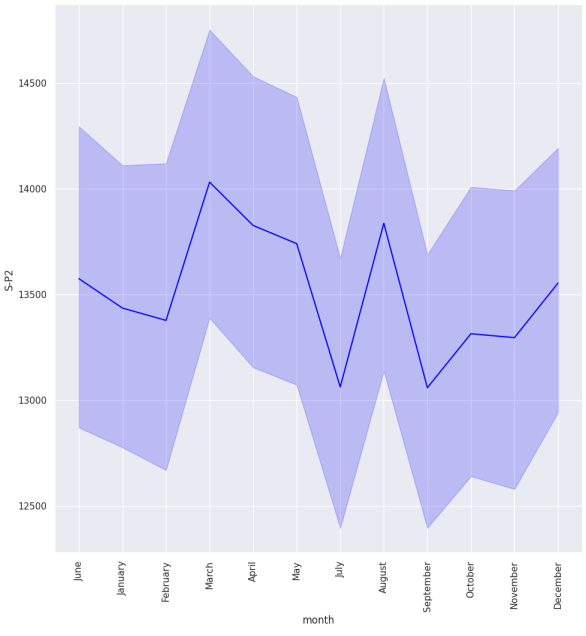


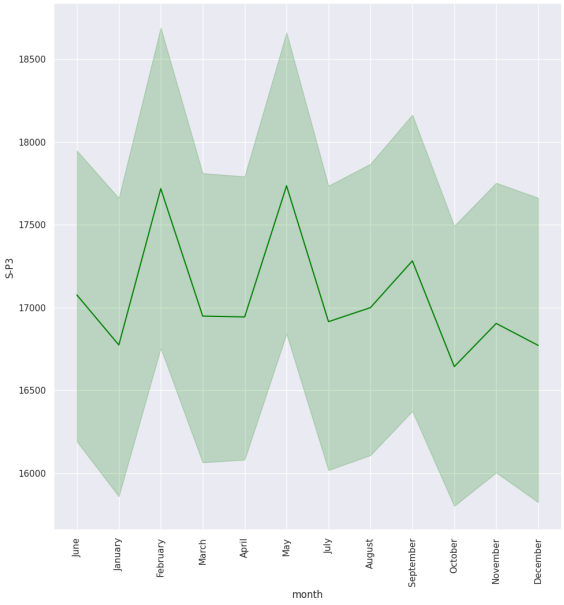
7. Code

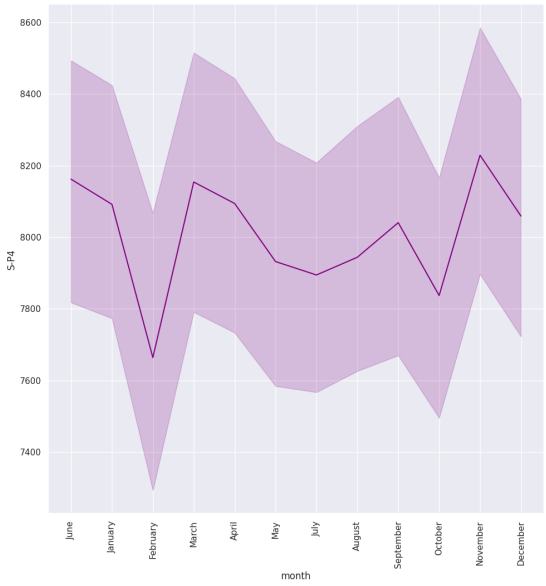
```
sns.relplot(x="month",y="S-P1",data=df,kind="line",height=10,color="red")

plt.xticks(rotation=90);

sns.relplot(x="month",y="S-P2",data=df,kind="line",height=10,color="blue")

plt.xticks(rotation=90);

sns.relplot(x="month",y="S-P3",data=df,kind="line",height=10,color="green")

plt.xticks(rotation=90);

sns.relplot(x="month",y="S-P4",data=df,kind="line",height=10,color="purple")

plt.xticks(rotation=90);
```

Out









8. Code

```
df.groupby("month")[["S-P1","S-P2","S-P3","S-P4"]].sum()
```

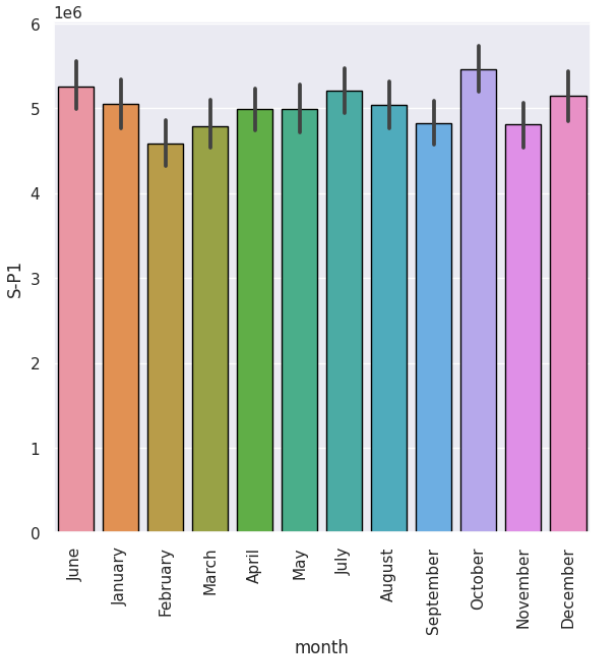
Out

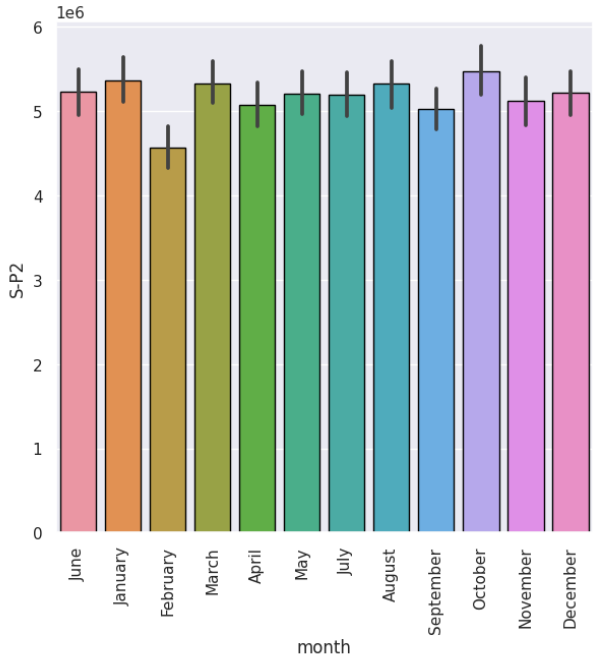
month	S-P1	S-P2	S-P3	S-P4
April	4994236.73	5074402.86	6218523.18	2970628.94
August	5032438.40	5327280.10	6545224.52	3058499.06
December	5140424.45	5218441.32	6457398.84	3102797.75
February	4576731.88	4561845.56	6042134.70	2613444.46
January	5048012.61	5360970.86	6693223.04	3228692.16
July	5205647.20	5199104.32	6732490.94	3142091.18
June	5251837.27	5226404.36	6574600.92	3142454.81
March	4786119.89	5332035.10	6440791.96	3098619.57
May	4983870.83	5207752.08	6722008.66	3006278.94
November	4813933.47	5119068.16	6508476.92	3168215.50
October	5454847.24	5472326.62	6840809.64	3221134.36
September	4816704.05	5027898.96	6653906.36	3095696.27

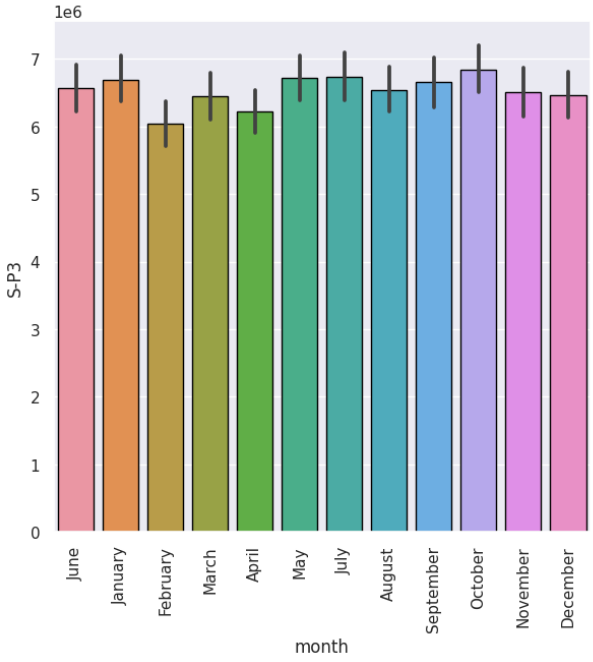
9. Code

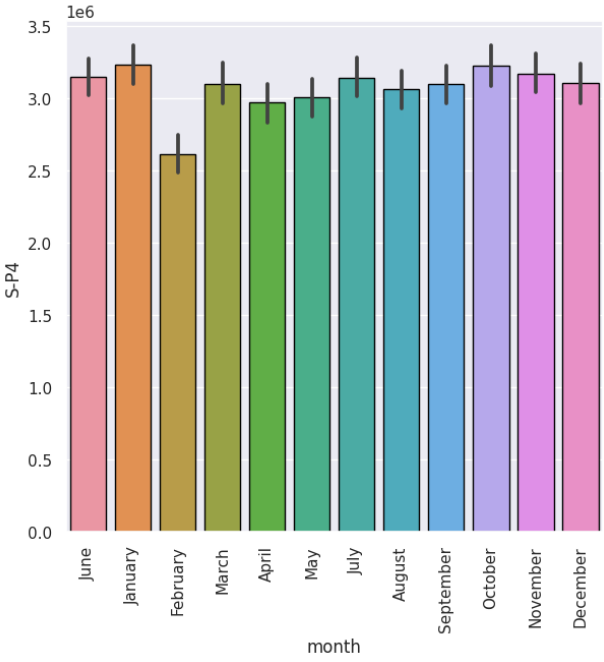
```
plt.figure(figsize=(15,15),dpi=100)

plt.subplot(2,2,1)

sns.barplot(x="month",y="S-P1",data=df,edgecolor="black",estim
ator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,2)

sns.barplot(x="month",y="S-P2",data=df,edgecolor="black",estim
ator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,3)

sns.barplot(x="month",y="S-P3",data=df,edgecolor="black",estim
ator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,4)

sns.barplot(x="month",y="S-P4",data=df,edgecolor="black",estim
ator=sum)

plt.xticks(rotation=90)

plt.subplots_adjust(hspace=0.3);
```

Out


1e6
S-P1
month
June
January
February
March
April
May
July
August
September
October
November
December


1e6
S-P2
month
June
January
February
March
April
May
July
August
September
October
November
December


1e6
S-P3
month
June
January
February
March
April
May
July
August
September
October
November
December


1e6
S-P4
month
June
January
February
March
April
May
July
August
September
October
November
December

10. Code

```
df.groupby ("month")[["Q-P1","Q-P2","Q-P3","Q-P4"]].sum()
```

Out

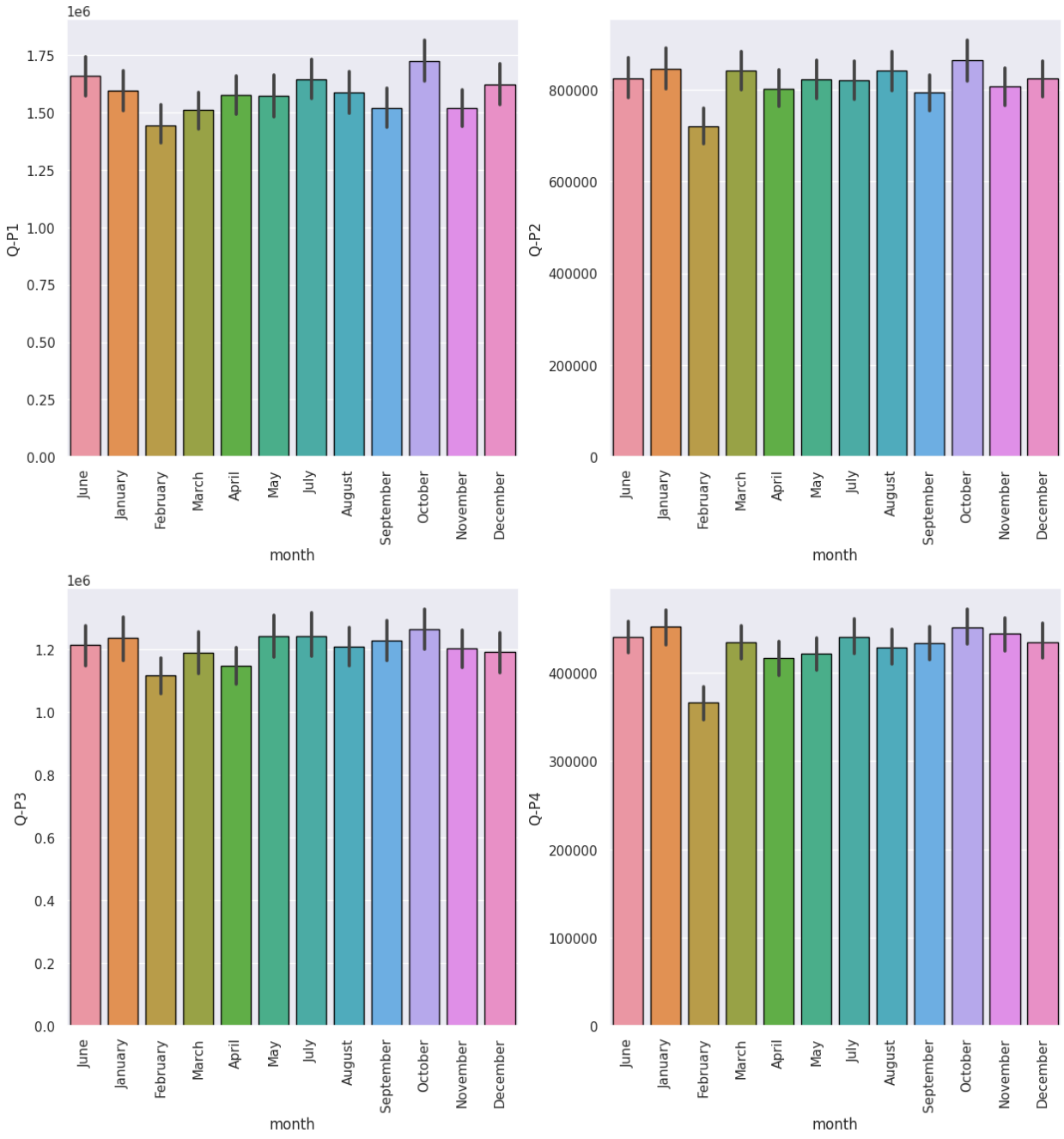
month	Q-P1	Q-P2	Q-P3	Q-P4
April	1575469	800379	1147329	416638
August	1587520	840265	1207606	428962
December	1621585	823098	1191402	435175
February	1443764	719534	1114785	366542
January	1592433	845579	1234912	452832
July	1642160	820048	1242157	440686
June	1656731	824354	1213026	440737
March	1509817	841015	1188338	434589
May	1572199	821412	1240223	421638
November	1518591	807424	1200826	444350
October	1720772	863143	1262142	451772
September	1519465	793044	1227658	434179

11. Code

```
plt.figure(figsize=(15,15),dpi=100)

plt.subplot(2,2,1)

sns.barplot(x="month",y="Q-P1",data=df,edgecolor="black",estim
ator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,2)
```

```
sns.barplot(x="month",y="Q-P2",data=df,edgecolor="black",estim
ator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,3)

sns.barplot(x="month",y="Q-P3",data=df,edgecolor="black",estim
ator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,4)

sns.barplot(x="month",y="Q-P4",data=df,edgecolor="black",estim
ator=sum)

plt.xticks(rotation=90)

plt.subplots_adjust(hspace=0.3);
```

Out


1e6
Q-P1
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
June
January
February
March
April
May
July
August
September
October
November
December
month
Q-P2
800000
600000
400000
200000
0
Q-P3
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Q-P4
400000
300000
200000
100000
0

12. Code

```
week_t=df[df["dayoftheweek"]<5]

weekend_t=df[df["dayoftheweek"]>=5]

print(week_t.groupby("day")[["S-P1","S-P2","S-P3","S-P4"]].sum
())
```

Out

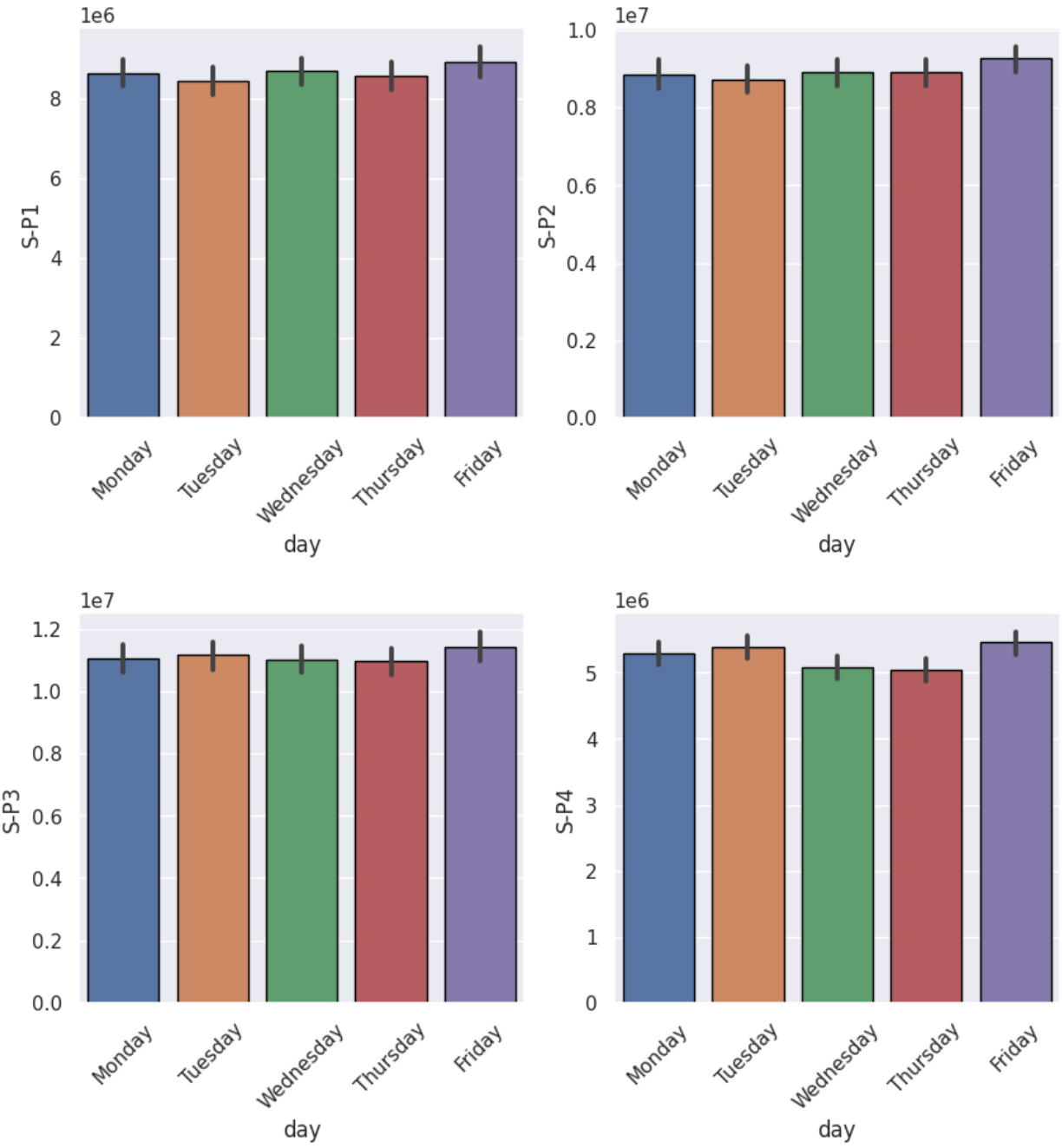
day	S-P1	S-P2	S-P3	S-P4
Friday	8913637.41	9267831.02	11428877.58	5463169.99
Monday	8636791.80	8864347.08	11064892.06	5292577.61
Thursday	8577981.96	8909481.54	10951554.44	5043013.35
Tuesday	8433525.06	8738326.90	11156338.30	5384854.07
Wednesday	8693537.97	8908067.72	11017830.20	5086827.20

13. Code

```
plt.figure(figsize=(10,10),dpi=100)

plt.subplot(2,2,1)

sns.barplot(x="day",y="S-P1",data=week_t,edgecolor="black",est
imator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,2)

sns.barplot(x="day",y="S-P2",data=week_t,edgecolor="black",est
imator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,3)

sns.barplot(x="day",y="S-P3",data=week_t,edgecolor="black",est
imator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,4)
```

```
sns.barplot(x="day",y="S-P4",data=week_t,edgecolor="black",est
imator=sum)

plt.xticks(rotation=45)

plt.subplots_adjust(hspace=0.5);
```

Out



14. Code

```
print(weekend_t.groupby("day")[["S-P1","S-P2","S-P3","S-P4"]].
sum())
```

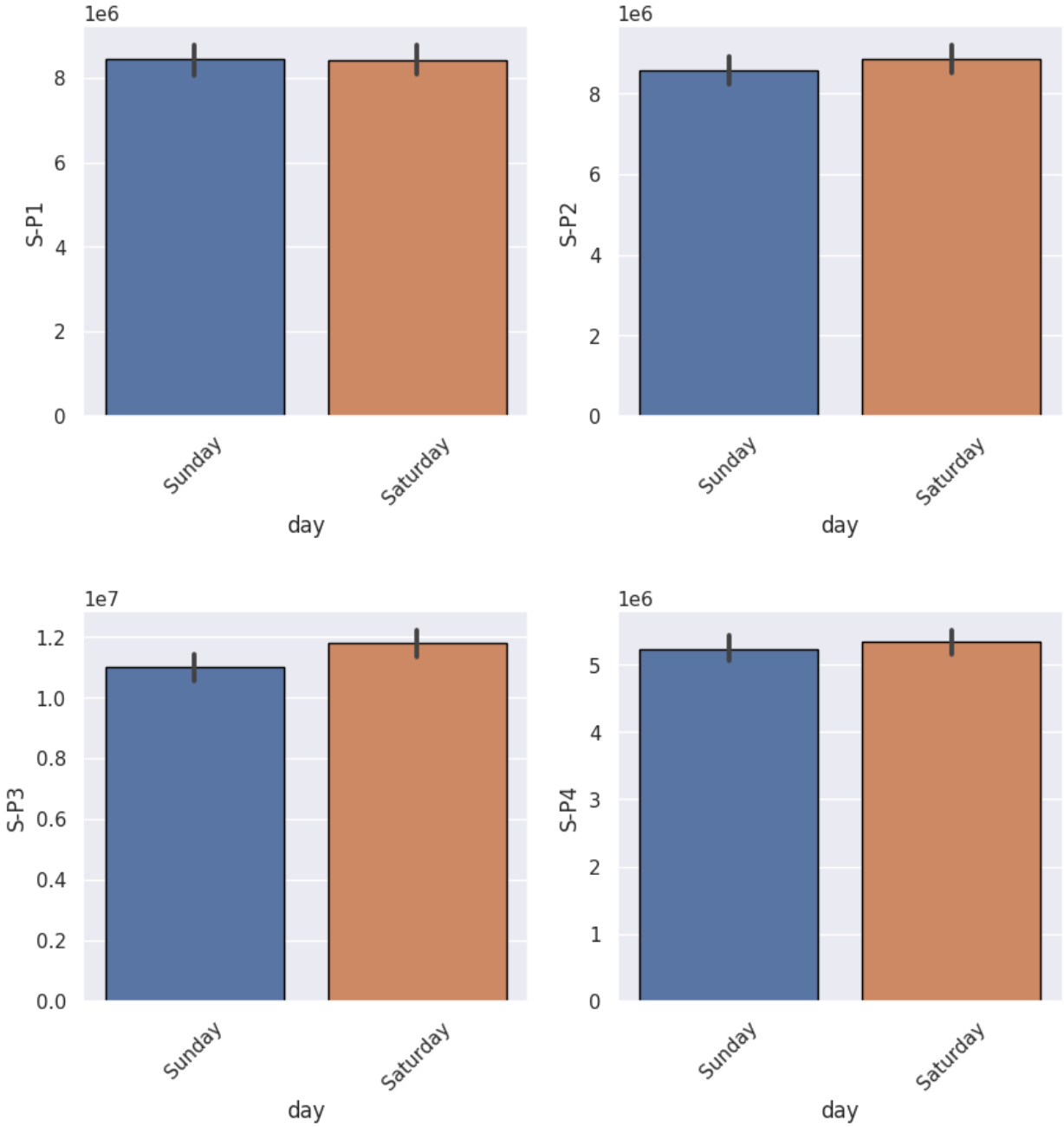
Out

day	S-P1	S-P2	S-P3	S-P4
Saturday	8409578.88	8853201.36	11796375.26	5339977.85
Sunday	8439750.94	8586274.68	11013721.84	5238132.93

15. Code

```
plt.figure(figsize=(10,10),dpi=100)

plt.subplot(2,2,1)

sns.barplot(x="day",y="S-P1",data=weekend_t,edgecolor="black",
estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,2)

sns.barplot(x="day",y="S-P2",data=weekend_t,edgecolor="black",
estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,3)

sns.barplot(x="day",y="S-P3",data=weekend_t,edgecolor="black",
estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,4)

sns.barplot(x="day",y="S-P4",data=weekend_t,edgecolor="black",
estimator=sum)

plt.xticks(rotation=45)

plt.subplots_adjust(hspace=0.5);
```

Out



16. Code

```
df.groupby("year")[["S-P1","S-P2","S-P3","S-P4"]].agg(["sum"])
```

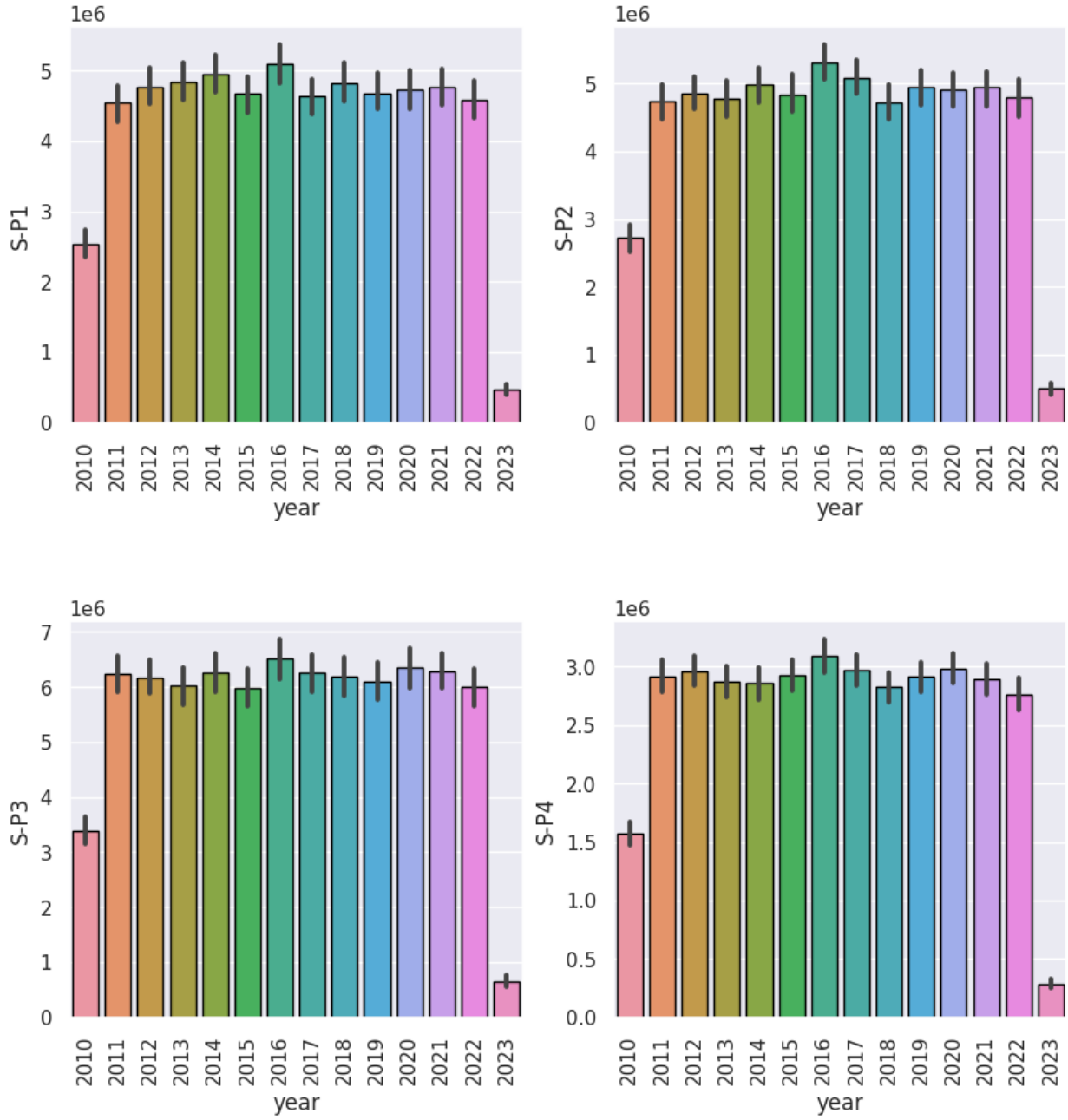
Out

	S-P1	S-P2	S-P3	S-P4
	sum	sum	sum	sum
year				
2010	2543459.01	2720100.92	3385462.08	1567523.37
2011	4542819.22	4741147.10	6235075.86	2921603.06
2012	4771163.83	4861987.50	6173911.16	2965210.14
2013	4833682.57	4771369.88	6017809.74	2868491.69
2014	4954522.97	4979797.38	6265406.18	2865119.20
2015	4669720.66	4833806.20	5987988.90	2933224.96
2016	5096066.64	5313116.54	6507718.12	3096444.92
2017	4628545.53	5085909.96	6269568.74	2969944.46
2018	4825792.44	4727313.22	6198517.96	2824392.64
2019	4681354.56	4946303.16	6106237.04	2912519.44
2020	4732093.58	4904826.88	6343643.88	2984618.00
2021	4758100.26	4948382.68	6294208.06	2894394.98
2022	4591000.05	4797040.54	5993479.36	2760400.89
2023	476482.70	496428.34	650562.60	284665.25

17. Code

```
plt.figure(figsize=(10,10),dpi=100)

plt.subplot(2,2,1)

sns.barplot(x="year",y="S-P1",data=df,edgecolor="black",estimator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,2)

sns.barplot(x="year",y="S-P2",data=df,edgecolor="black",estimator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,3)

sns.barplot(x="year",y="S-P3",data=df,edgecolor="black",estimator=sum)

plt.xticks(rotation=90);

plt.subplot(2,2,4)

sns.barplot(x="year",y="S-P4",data=df,edgecolor="black",estimator=sum)

plt.xticks(rotation=90)

plt.subplots_adjust(hspace=0.5);
```

Out



18. Code

```
df[["S-P1","S-P2","S-P3","S-P4"]].agg(["sum","max","min","mean
"])
```

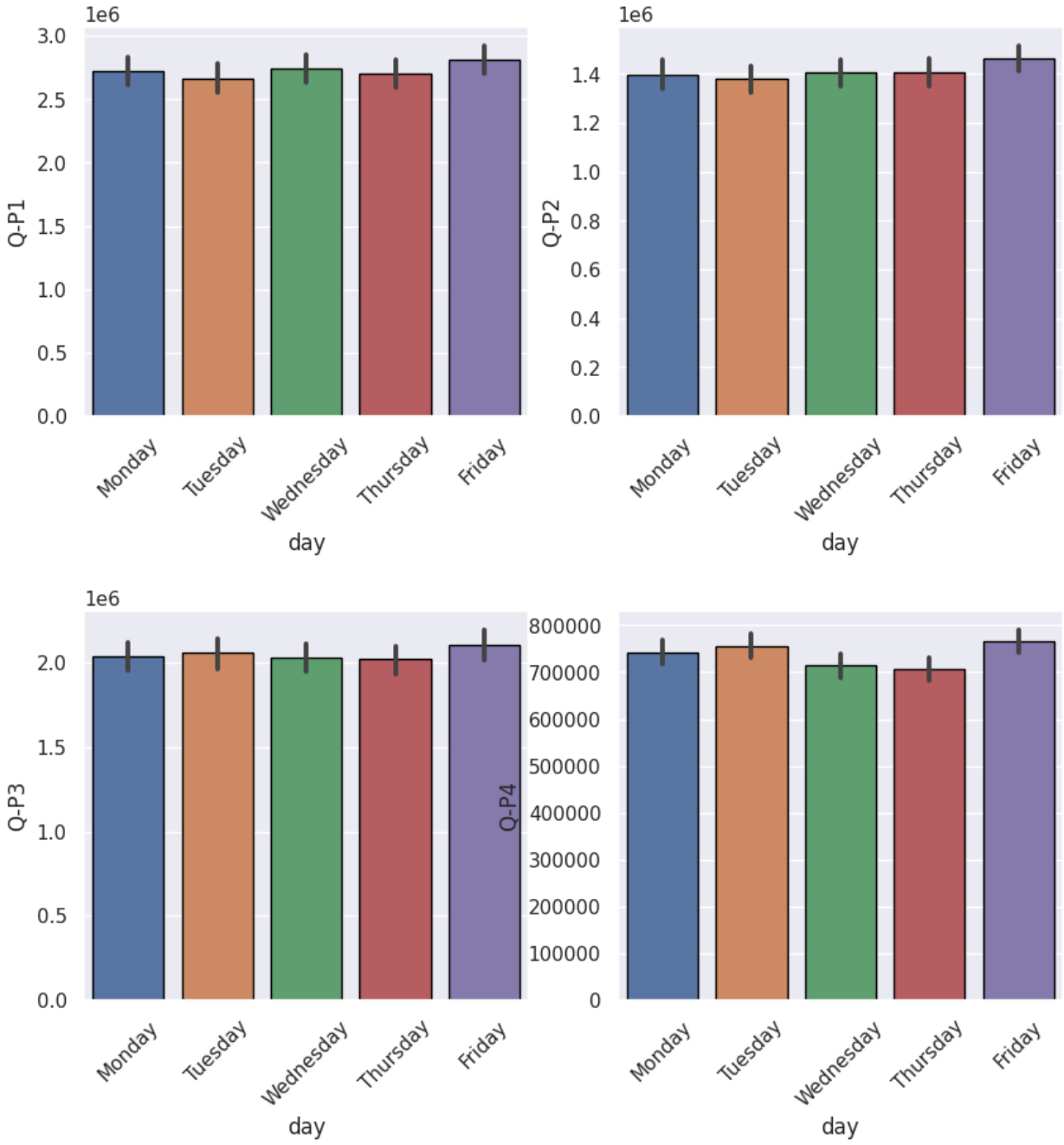
Out

	S-P1	S-P2	S-P3	S-P4
sum	6.010480e+07	6.212753e+07	7.842959e+07	3.684855e+07
max	2.535366e+04	2.534732e+04	3.252000e+04	1.426000e+04
min	8.051800e+02	1.591340e+03	1.355000e+03	1.782500e+03
mean	1.306626e+04	1.350598e+04	1.704991e+04	8.010555e+03

19. Code

```
plt.figure(figsize=(10,10),dpi=100)

plt.subplot(2,2,1)

sns.barplot(x="day",y="Q-P1",data=week_t,edgecolor="black",est
imator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,2)

sns.barplot(x="day",y="Q-P2",data=week_t,edgecolor="black",est
imator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,3)

sns.barplot(x="day",y="Q-P3",data=week_t,edgecolor="black",est
imator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,4)

sns.barplot(x="day",y="Q-P4",data=week_t,edgecolor="black",est
imator=sum)

plt.xticks(rotation=45)

plt.subplots_adjust(hspace=0.5);
```

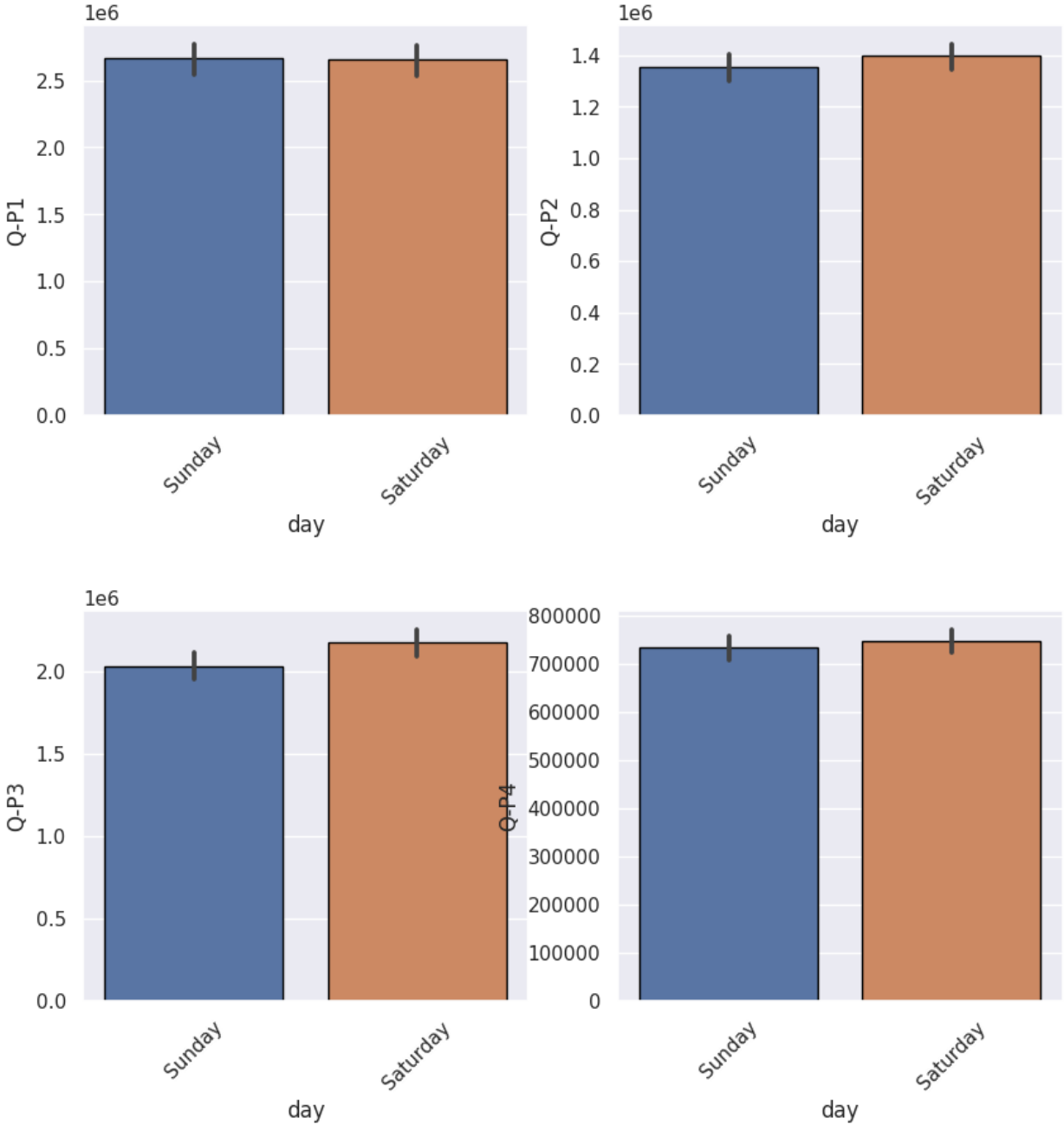
Out


1e6
Q-P1
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Monday
Tuesday
Wednesday
Thursday
Friday
day
1e6
Q-P2
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Monday
Tuesday
Wednesday
Thursday
Friday
day
1e6
Q-P3
2.0
1.5
1.0
0.5
0.0
Monday
Tuesday
Wednesday
Thursday
Friday
day
Q-P4
800000
700000
600000
500000
400000
300000
200000
100000
0
Monday
Tuesday
Wednesday
Thursday
Friday
day

20. Code

```
plt.figure(figsize=(10,10),dpi=100)

plt.subplot(2,2,1)

sns.barplot(x="day",y="Q-P1",data=weekend_t,edgecolor="black",
estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,2)

sns.barplot(x="day",y="Q-P2",data=weekend_t,edgecolor="black",
estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,3)

sns.barplot(x="day",y="Q-P3",data=weekend_t,edgecolor="black",
estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,4)

sns.barplot(x="day",y="Q-P4",data=weekend_t,edgecolor="black",
estimator=sum)

plt.xticks(rotation=45)

plt.subplots_adjust(hspace=0.5);
```

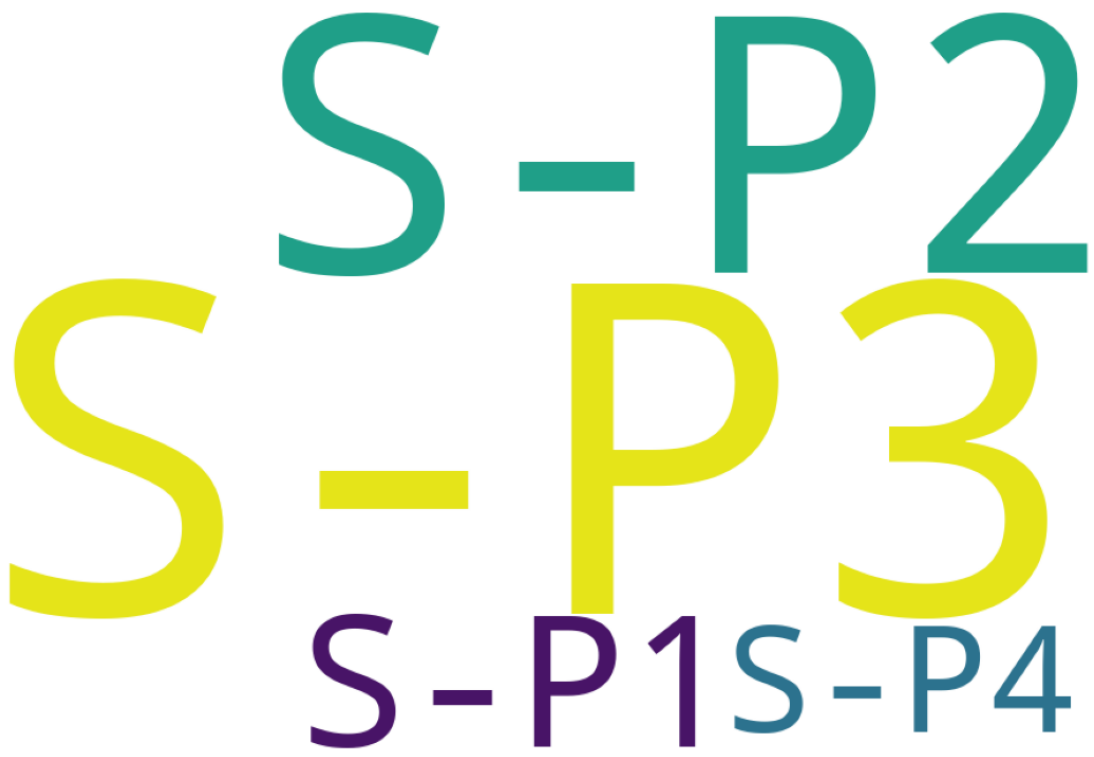
Out


1e6
Q-P1
2.5
2.0
1.5
1.0
0.5
0.0
Sunday
Saturday
day
1e6
Q-P2
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Sunday
Saturday
day
1e6
Q-P3
2.0
1.5
1.0
0.5
0.0
Sunday
Saturday
day
Q-P4
800000
700000
600000
500000
400000
300000
200000
100000
0
Sunday
Saturday
day

21. Code

```
from wordcloud import WordCloud as word

d=df[["S-P1","S-P2","S-P3","S-P4"]].sum()

wc = word(background_color='white', width=1000, height=600)

wc.generate_from_frequencies(d)

plt.figure(figsize=(15,15),dpi=100)

plt.imshow(wc)

plt.axis('off')

plt.show()
```

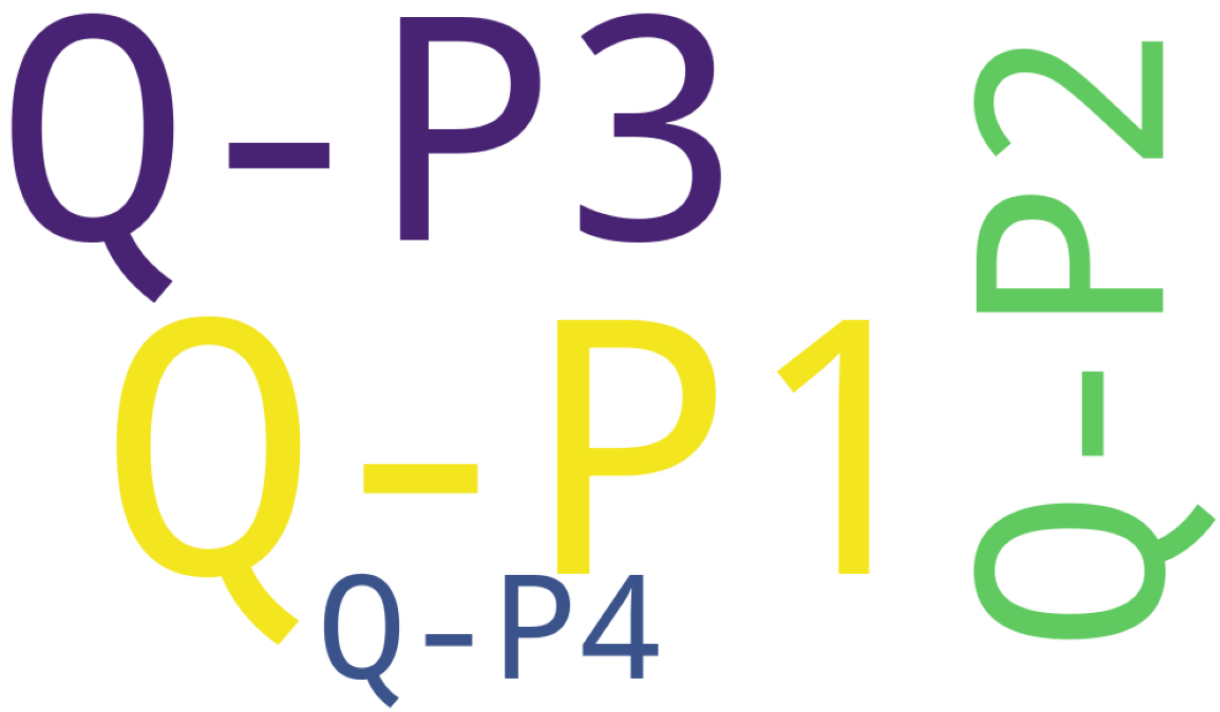
Out



22. Code

```
q=df[["Q-P1","Q-P2","Q-P3","Q-P4"]].sum()

wc = word(background_color='white', width=1000, height=600)

wc.generate_from_frequencies(q)

plt.figure(figsize=(15,15),dpi=100)

plt.imshow(wc)

plt.axis('off')

plt.show()
```

Out



https://colab.research.google.com/drive/1d3PCu5_NhTyP80NYDCE7BUkgj3mwzrt_?usp=sharing#scrollTo=LVYY3VdNIZVG