**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.

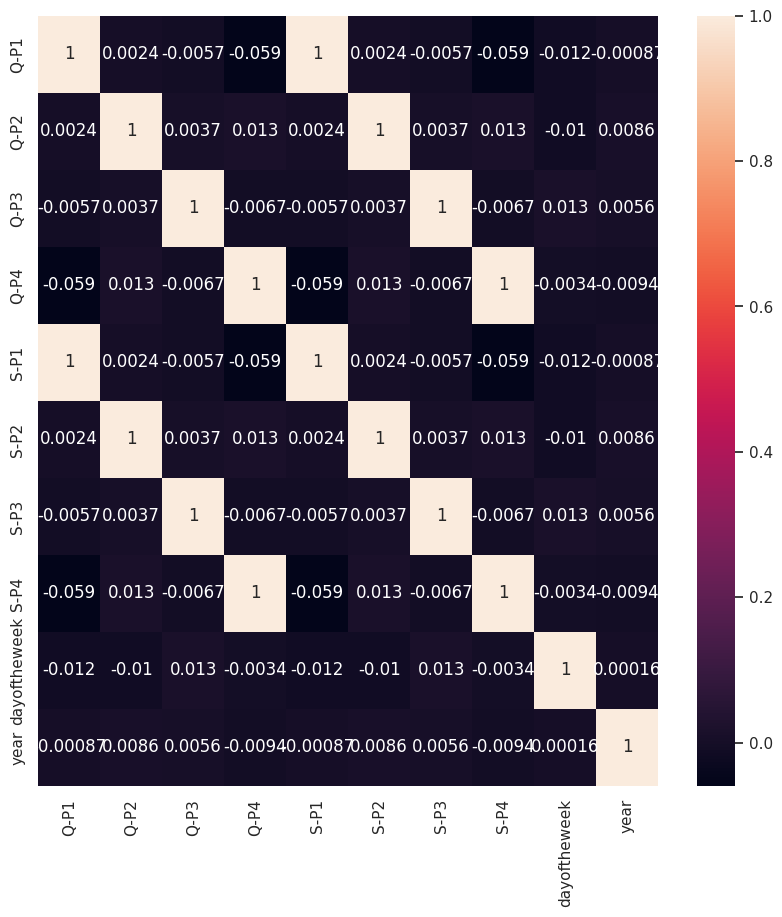
1. **Data Visualization:**

**Code and Outputs :**

**1.Code:**

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

sns.heatmap(df.corr(),annot=True)

**Output**

**2.Code:**

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

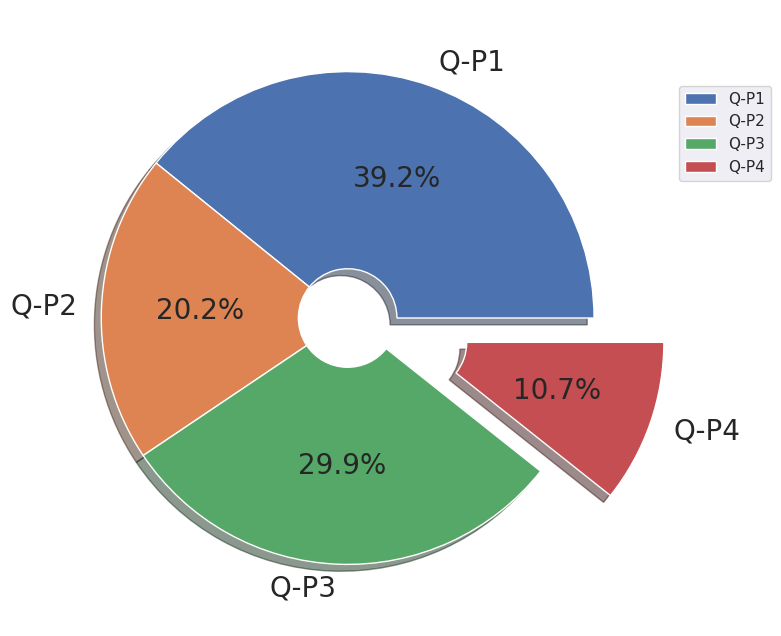
print(q)

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));

**Output:**

Q-P1 18960506 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))

**Output:**

S-P1    60104804.02

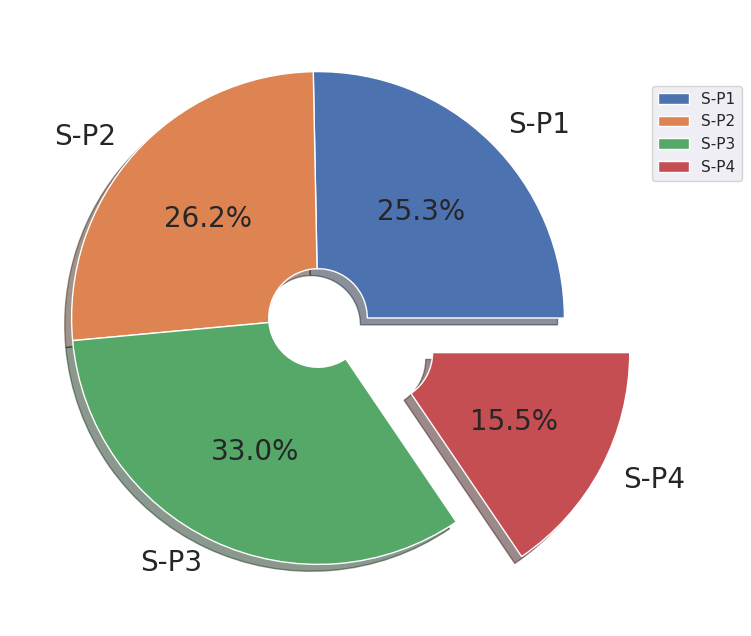
S-P2    62127530.30

S-P3    78429589.68

S-P4    36848553.00

dtype: float64

<matplotlib.legend.Legend at 0x79ead813ff10>



**4.Code:**

print(df["month"].value\_counts())

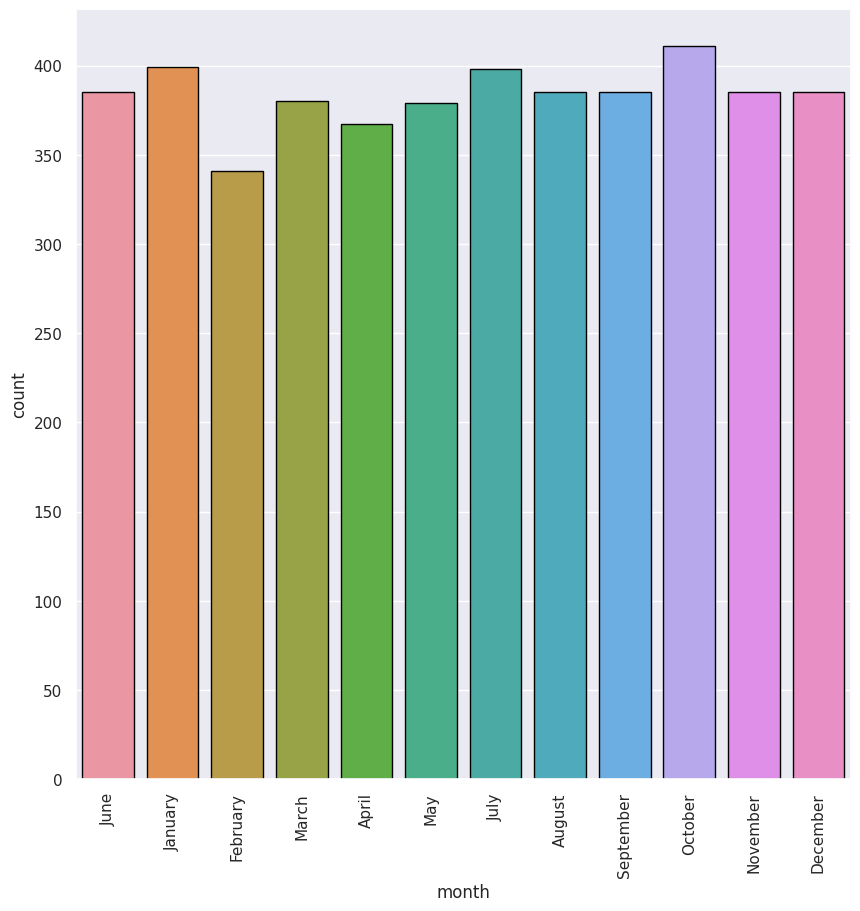
plt.figure(figsize=(10,10))

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

plt.xticks(rotation=90);

**Output:**

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())

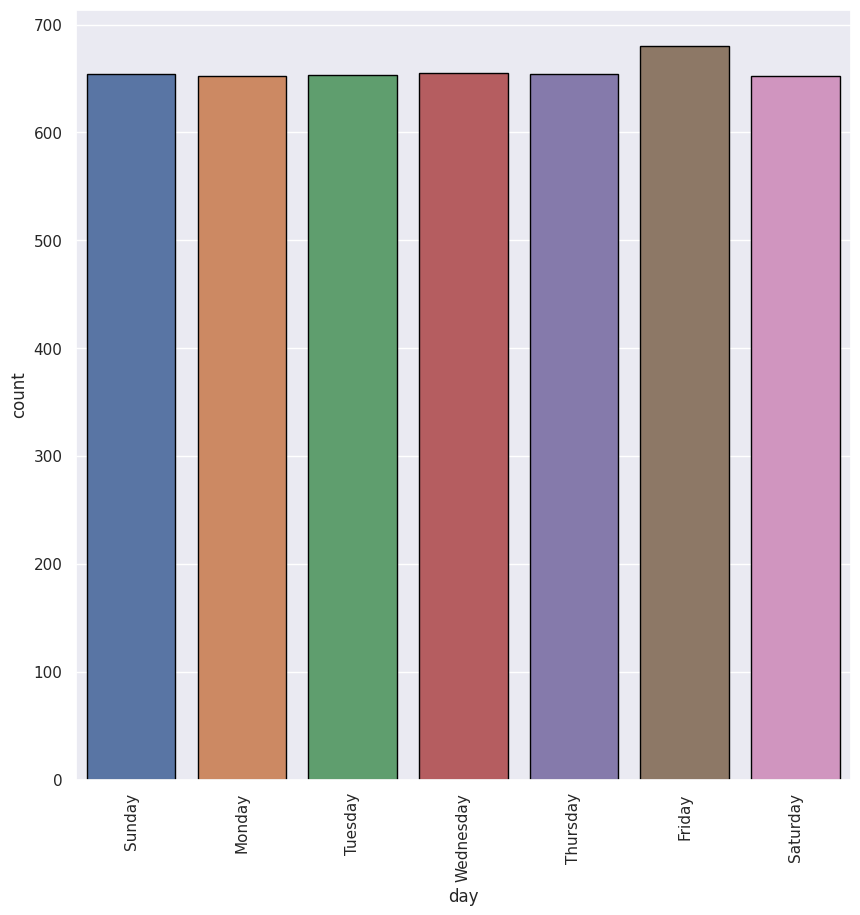
plt.figure(figsize=(10,10))

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

plt.xticks(rotation=90);

**Output:**

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



1. **Code:**

print(df["year"].value\_counts())

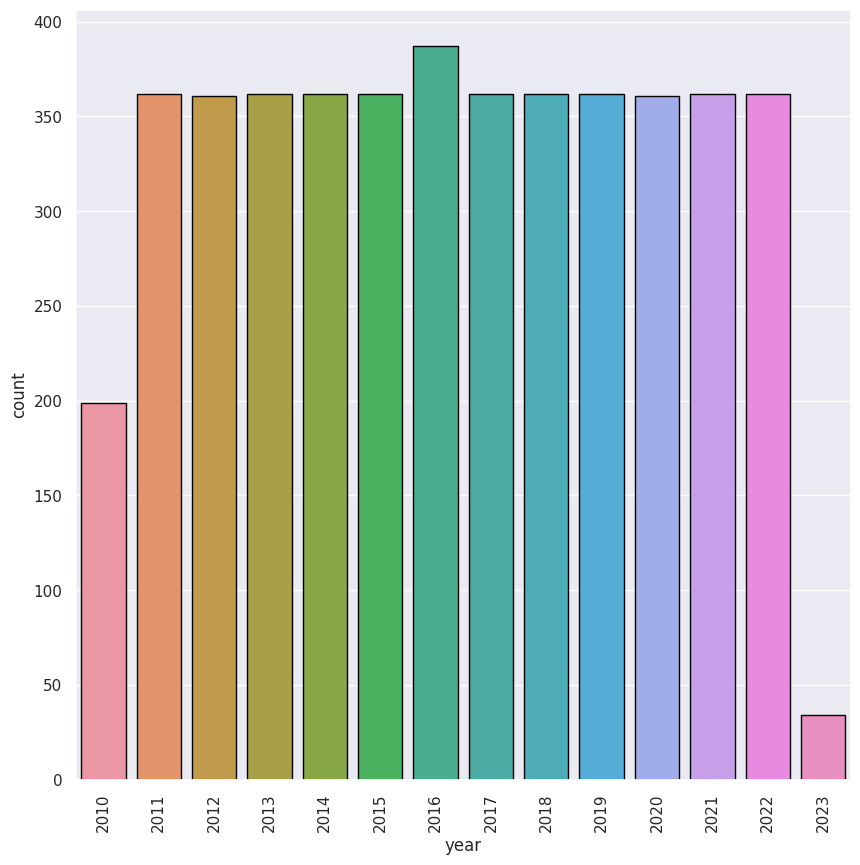
plt.figure(figsize=(10,10))

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

plt.xticks(rotation=90);

**Output:**

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



1. **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);

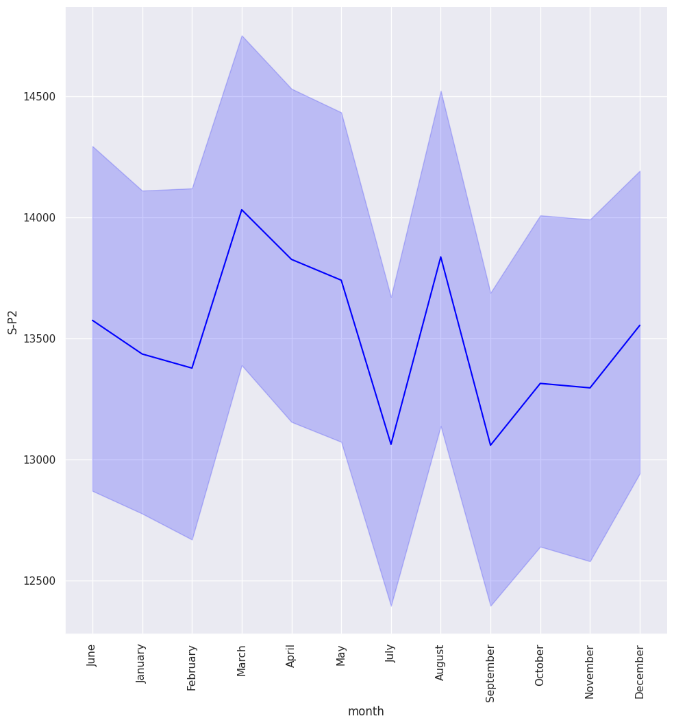
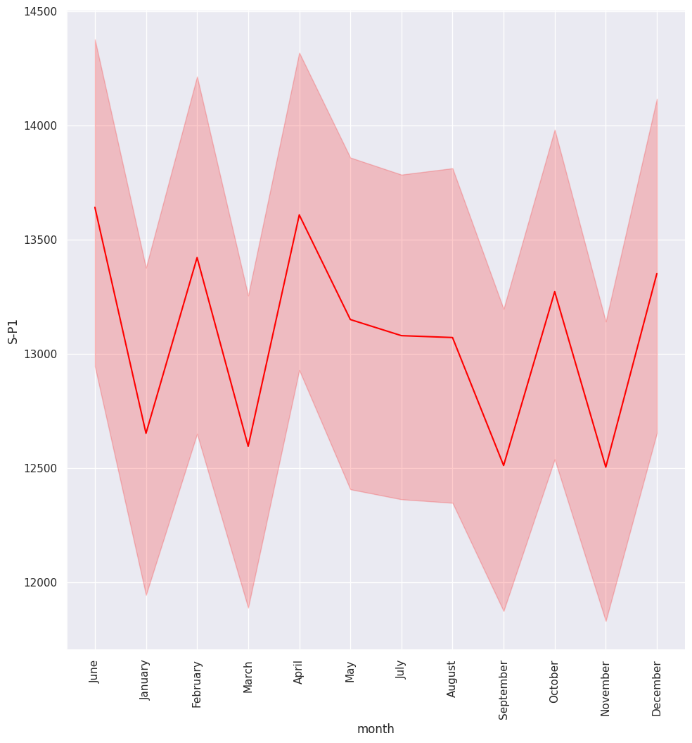
sns.relplot(x="month",y="S-P3",data=df,kind="line",height=10,color="green")

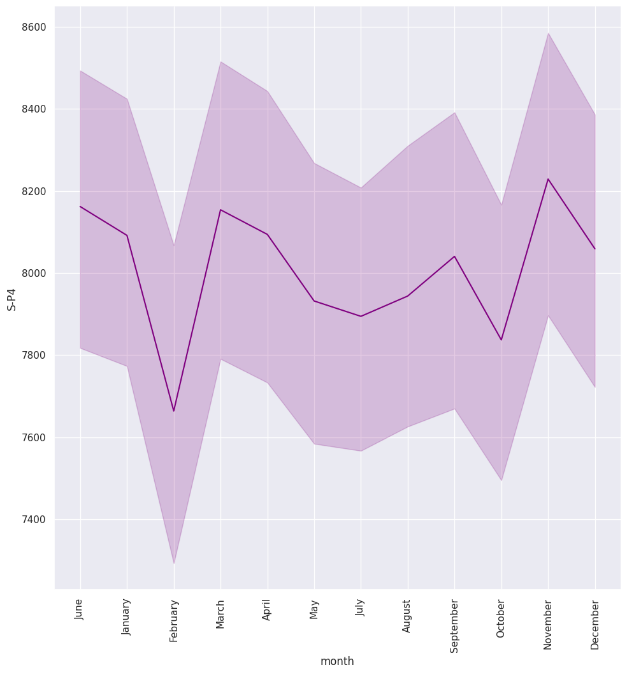
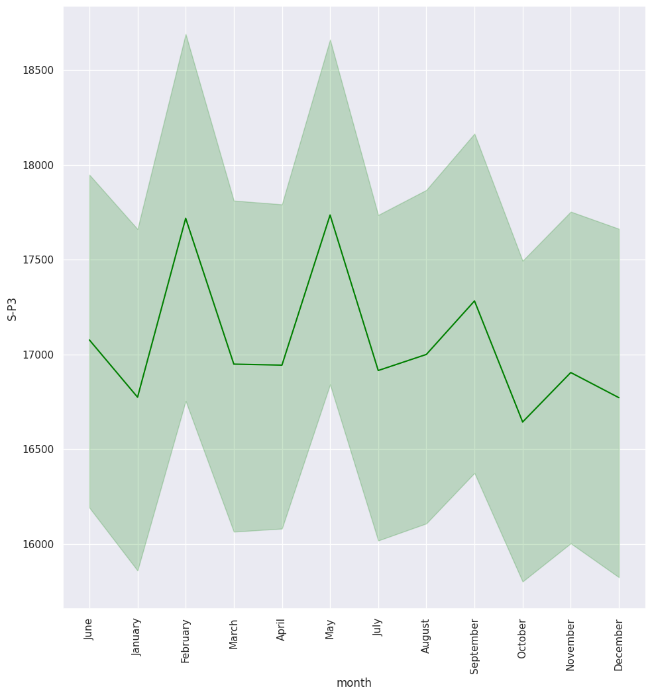
plt.xticks(rotation=90);

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

plt.xticks(rotation=90);

**Output:**

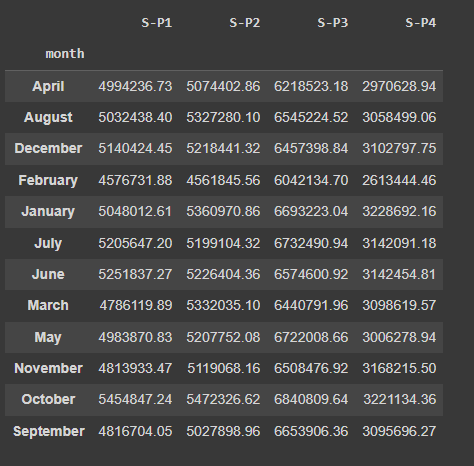




1. **Code:**

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

**Output:**



1. **Code:**

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

plt.subplot(2,2,1)

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

plt.xticks(rotation=90);

plt.subplot(2,2,2)

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

plt.xticks(rotation=90);

plt.subplot(2,2,3)

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

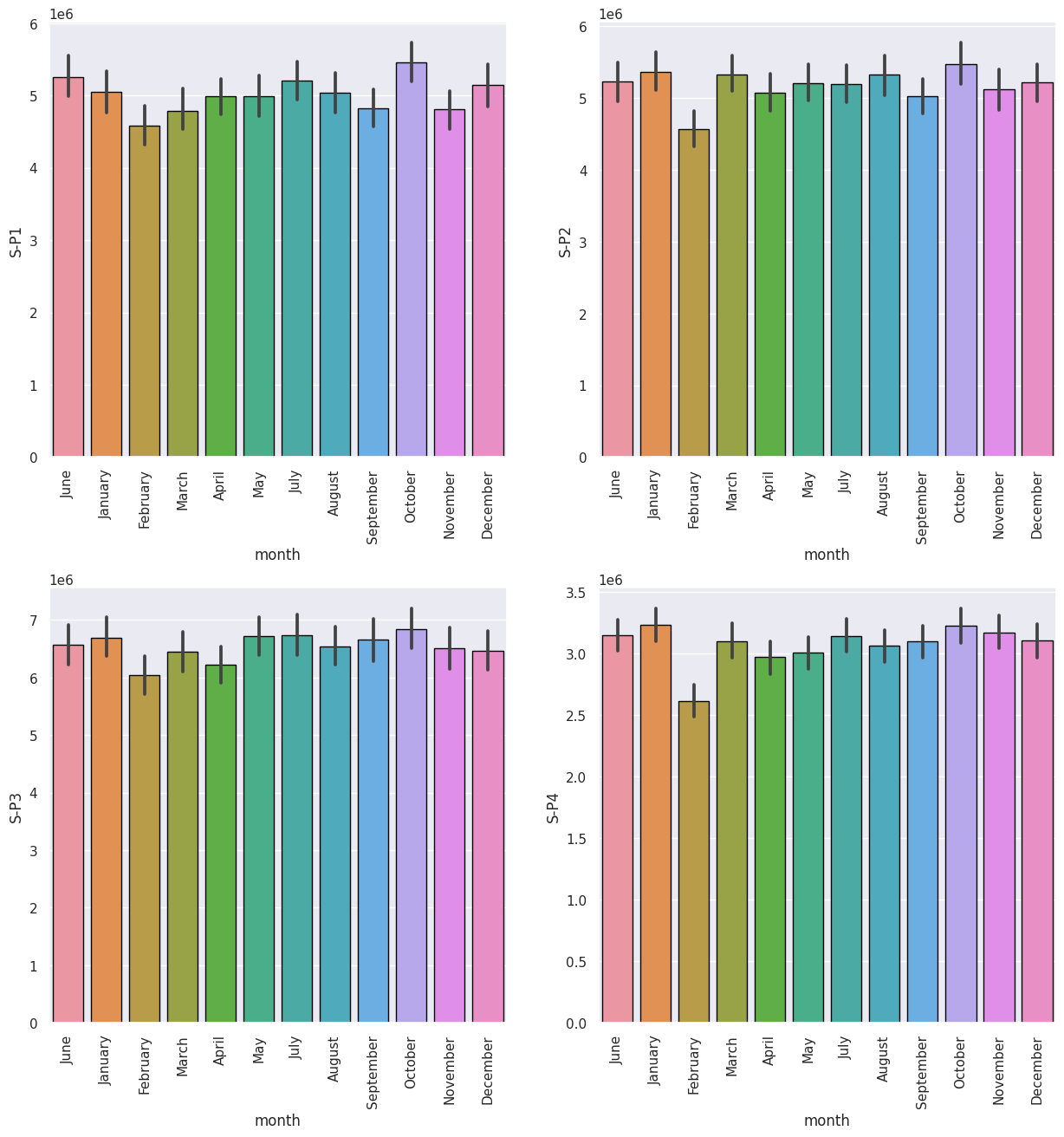
plt.xticks(rotation=90);

plt.subplot(2,2,4)

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

plt.xticks(rotation=90)

plt.subplots\_adjust(hspace=0.3);

 **Output:**

1. **Code:**

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

**Output:**



1. **Code:**

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

plt.subplot(2,2,1)

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

plt.xticks(rotation=90);

plt.subplot(2,2,2)

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

plt.xticks(rotation=90);

plt.subplot(2,2,3)

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

plt.xticks(rotation=90);

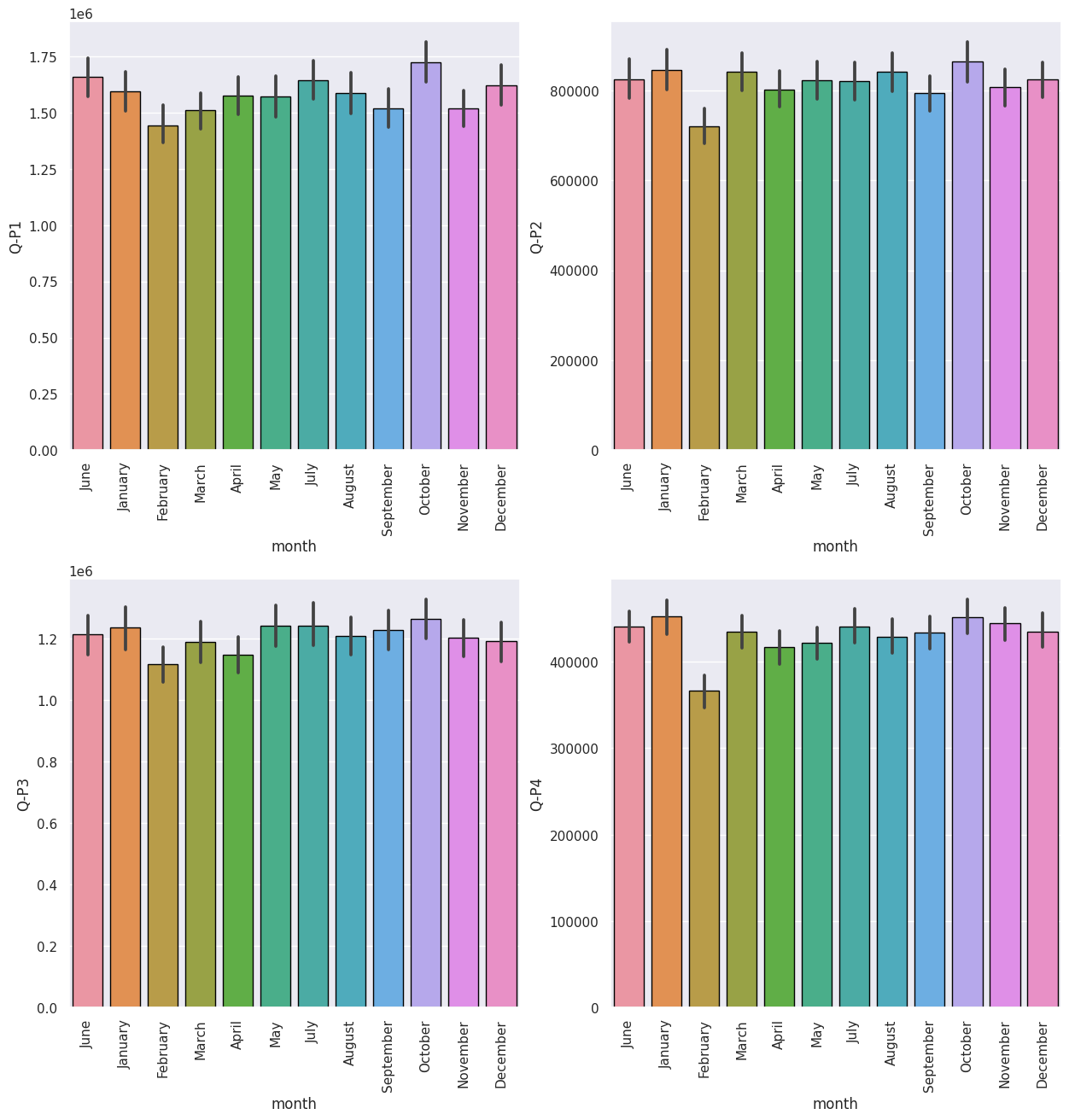
plt.subplot(2,2,4)

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

plt.xticks(rotation=90)

plt.subplots\_adjust(hspace=0.3);

**Output:**



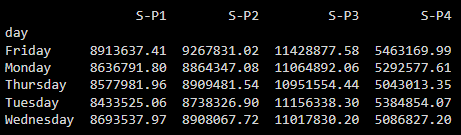
1. **Code:**

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())

**Output:**



**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",estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,2)

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

plt.xticks(rotation=45);

plt.subplot(2,2,3)

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

plt.xticks(rotation=45);

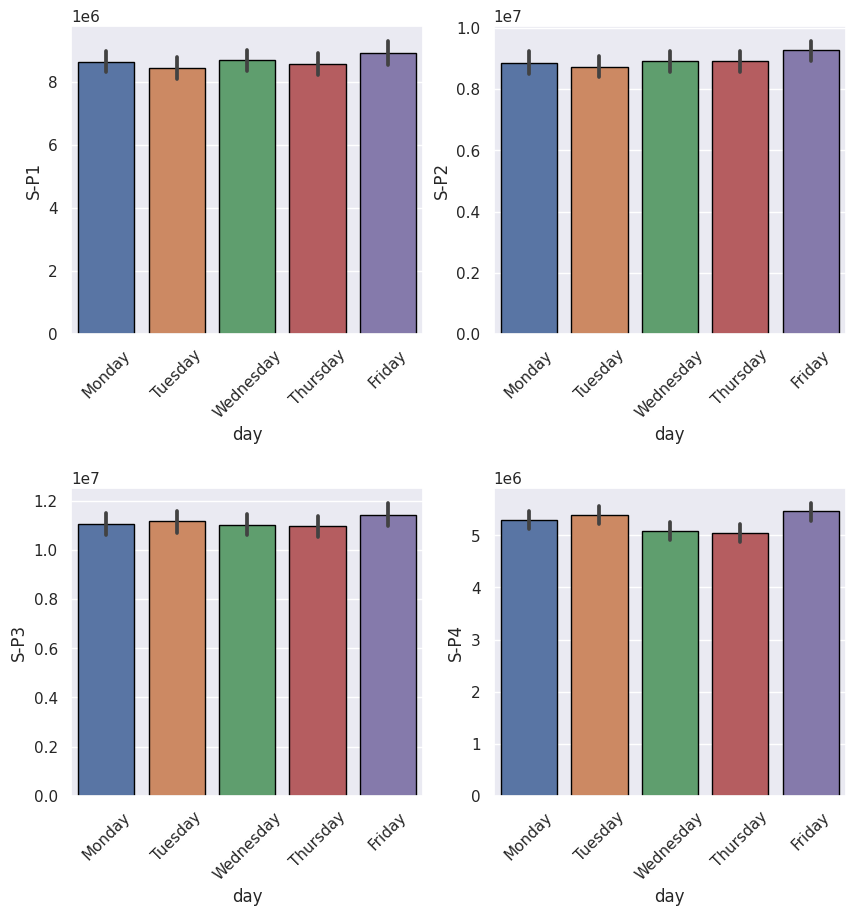
plt.subplot(2,2,4)

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

plt.xticks(rotation=45)

plt.subplots\_adjust(hspace=0.5);

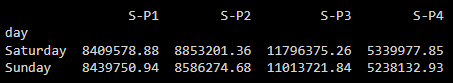
**Output:**



1. **Code:**

print(weekend\_t.groupby("day")[["S-P1","S-P2","S-P3","S-P4"]].sum())

**Output:**



1. **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);

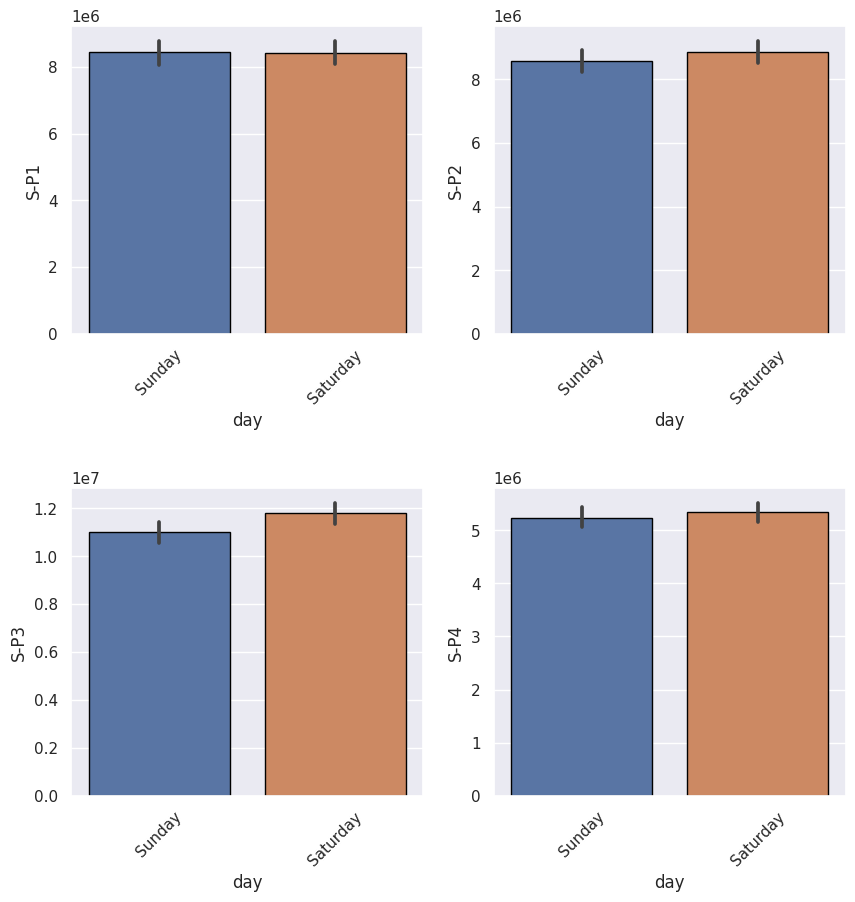
plt.subplot(2,2,4)

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

plt.xticks(rotation=45)

plt.subplots\_adjust(hspace=0.5);

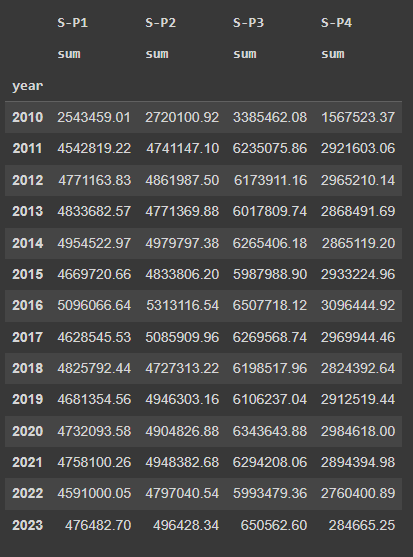
**Output:**



1. **Code:**

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

**Output:**



1. **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);

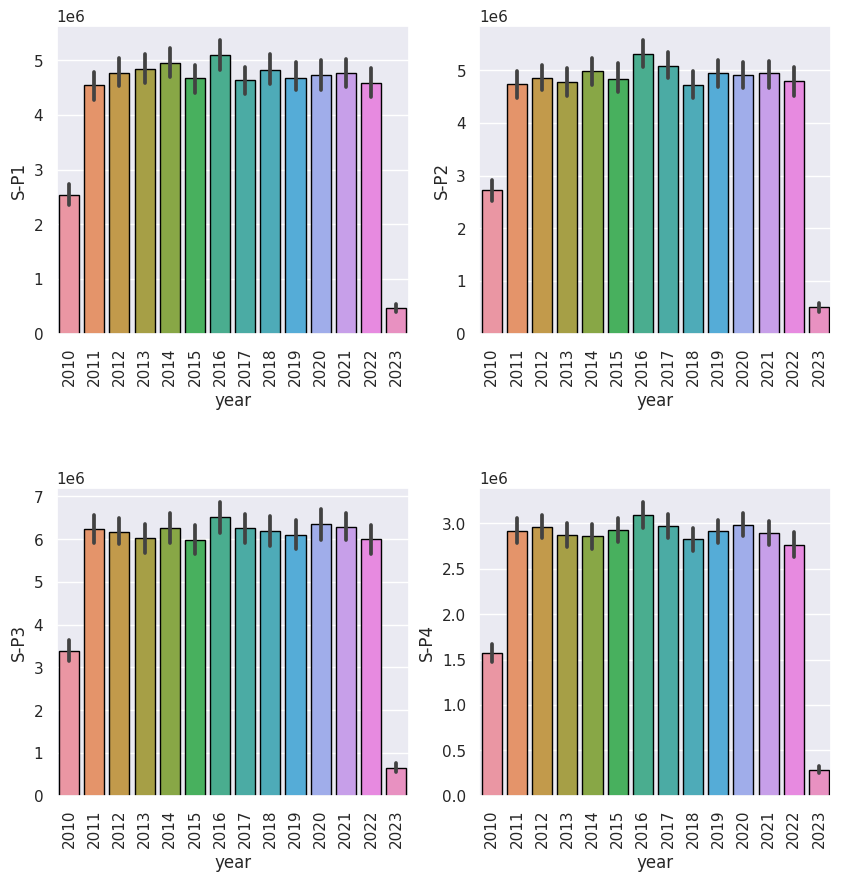
plt.subplot(2,2,4)

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

plt.xticks(rotation=90)

plt.subplots\_adjust(hspace=0.5);

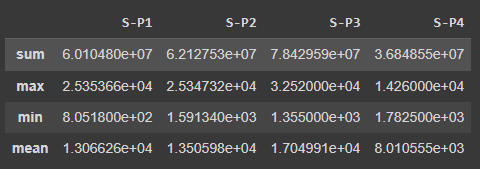
**Output:**



1. **Code:**

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

**Output:**



1. **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",estimator=sum)

plt.xticks(rotation=45);

plt.subplot(2,2,2)

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

plt.xticks(rotation=45);

plt.subplot(2,2,3)

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

plt.xticks(rotation=45);

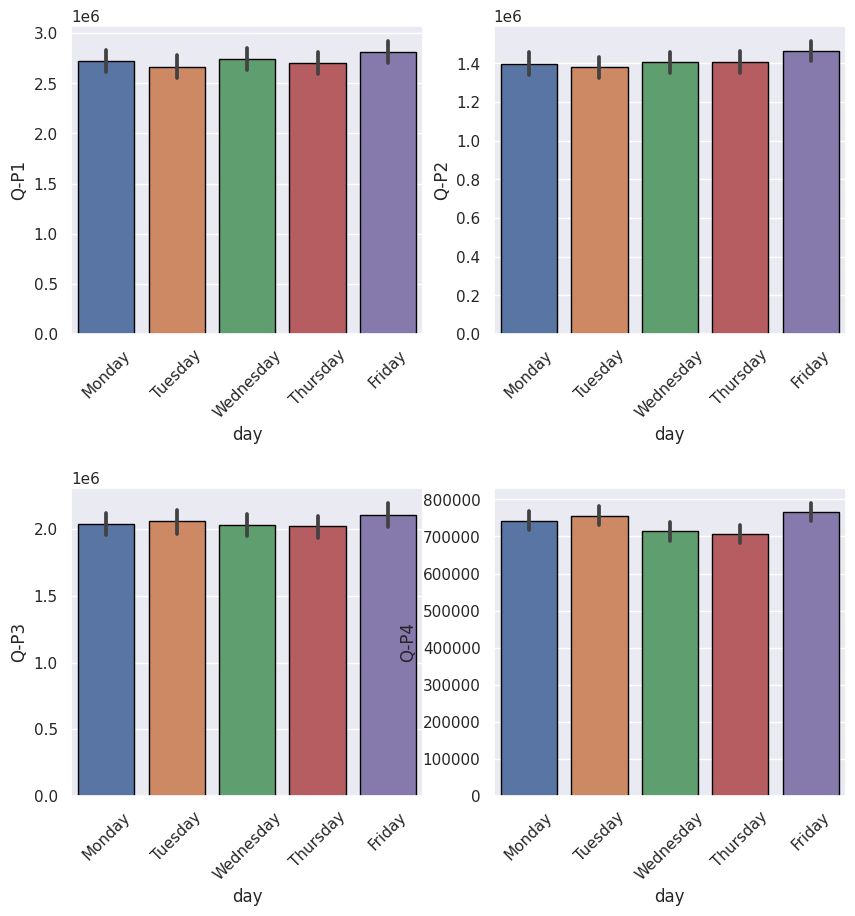
plt.subplot(2,2,4)

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

plt.xticks(rotation=45)

plt.subplots\_adjust(hspace=0.5);

**Output:**



1. **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);

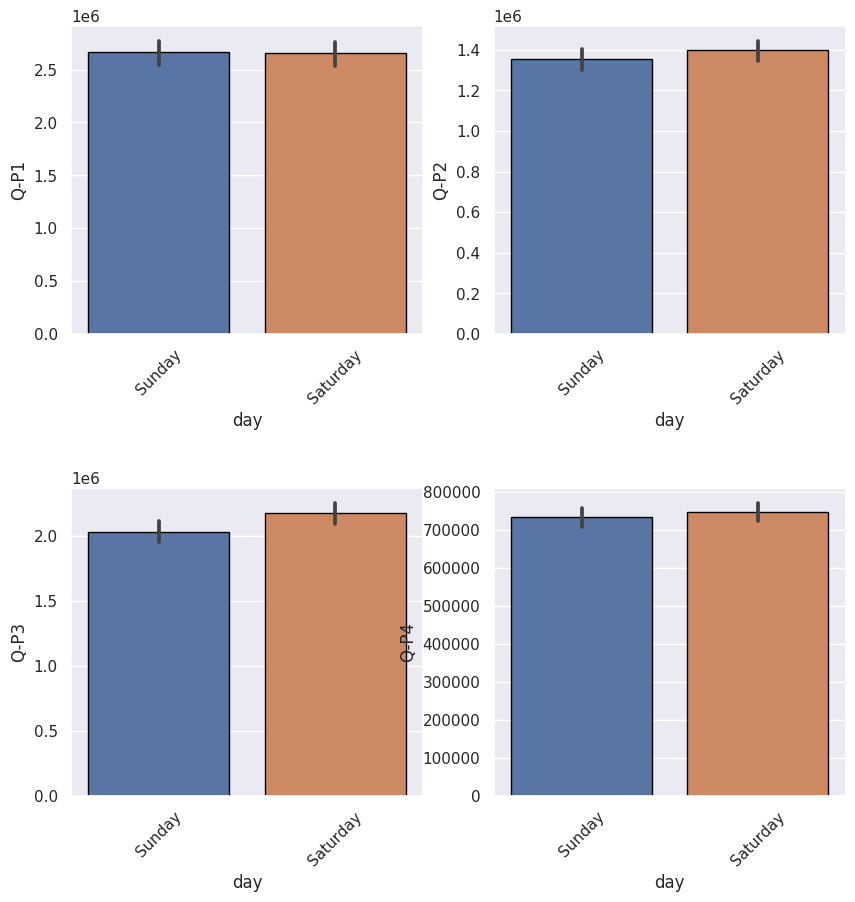
plt.subplot(2,2,4)

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

plt.xticks(rotation=45)

plt.subplots\_adjust(hspace=0.5);

**Output:**



1. **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)

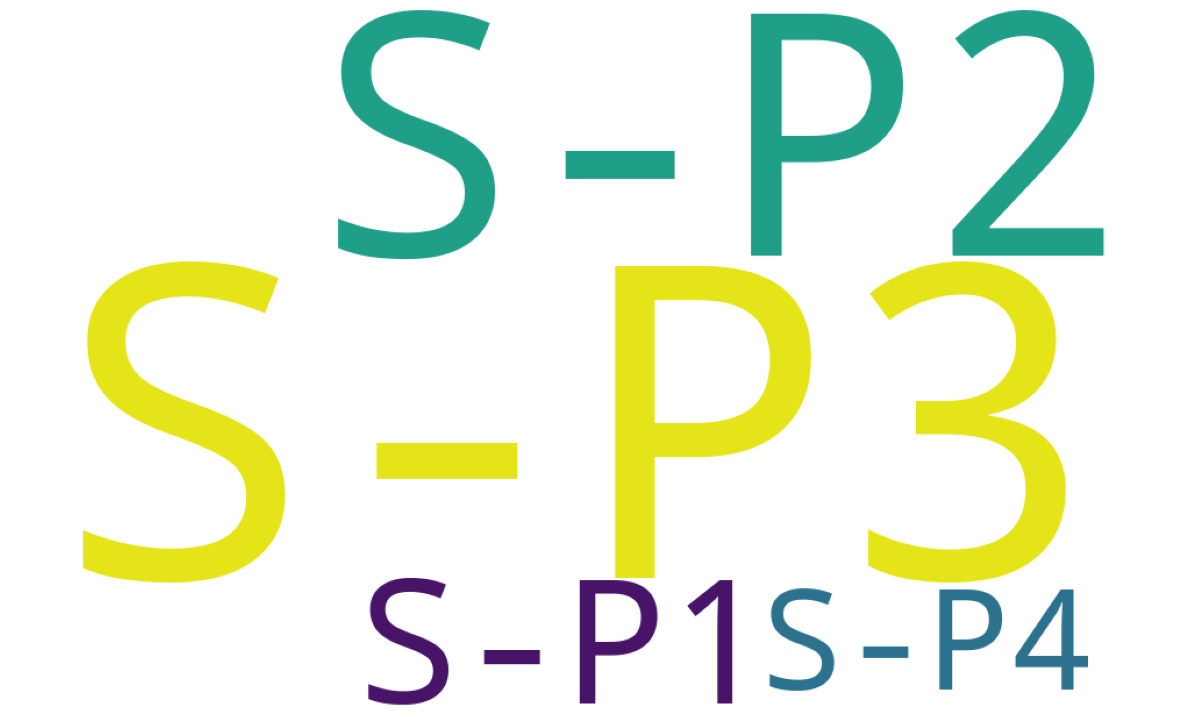
plt.figure(figsize=(15,15),dpi=100)

plt.imshow(wc)

plt.axis('off')

plt.show()

**Output:**



1. **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()

**Output:**

