Problem statement:

New-Wheels sales have been dipping steadily in the past year, and due to the critical customer feedback and ratings online, there has been a drop in new customers every quarter, which is concerning to the business. The CEO of the company now wants a quarterly report with all the key metrics sent to him so he can assess the health of the business and make the necessary decisions.

Objective:

As a data scientist, you see that there is an array of questions that are being asked at the leadership level that needs to be answered using data. But this data is not organized and is just being dumped as flat files to be used only when there are concerning problems! You need to first create a pipeline to organize and maintain this data using a SQL database so that it becomes easy to answer questions in the future. Once this is done, use the data to answer the questions posed and create a quarterly business report for the CEO.

Importing the necessary packages

In [1]:

```
# Import the libraries

import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
%matplotlib inline
pd.set_option('display.max_columns',None)
```

Question 1

What is the distribution of customers across states?

Hint: For each state, count the number of customers.

In [2]:

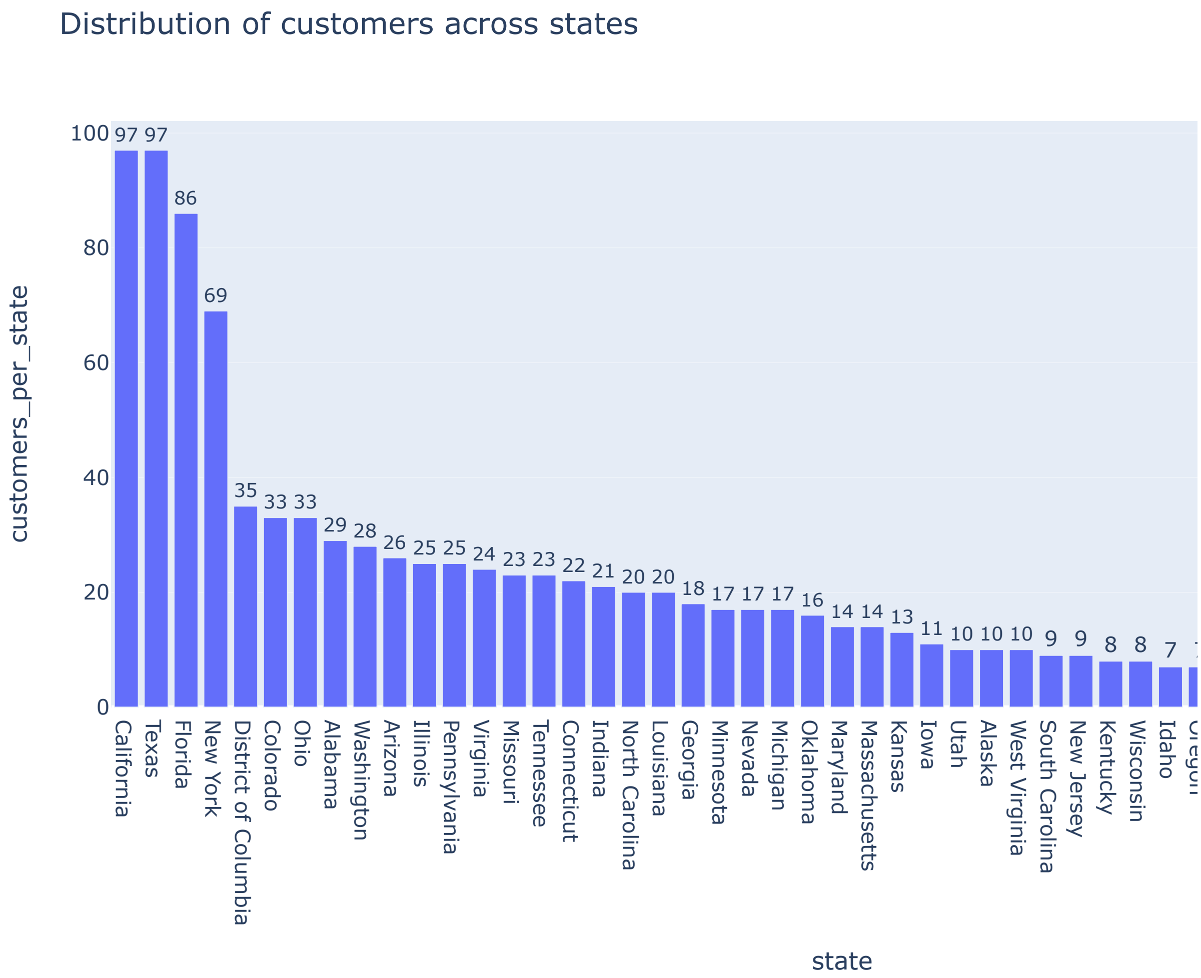
```
q1 = pd.read_csv("q1.csv")
q1.head()
```

Out[2]:

	state	customers_per_state
0	California	97
1	Texas	97
2	Florida	86
3	New York	69
4	District of Columbia	35

In [3]:

```
fig = px.bar(q1, x="state", y="customers_per_state", title="Distribution of customers across state
fig.update_layout(xaxis=dict(tickmode='array'), autosize=False,
    width=1000,
    height=600)
fig.update_traces(textfont_size=12, textangle=0, textposition="outside", cliponaxis=False)
fig.show()
```



Question 2

What is the average rating in each quarter? Very Bad is 1, Bad is 2, Okay is 3, Good is 4, Very Good is 5.

Hint: Use a common table expression and in that CTE, assign numbers to the different customer ratings. Now average the feedback for each quarter.

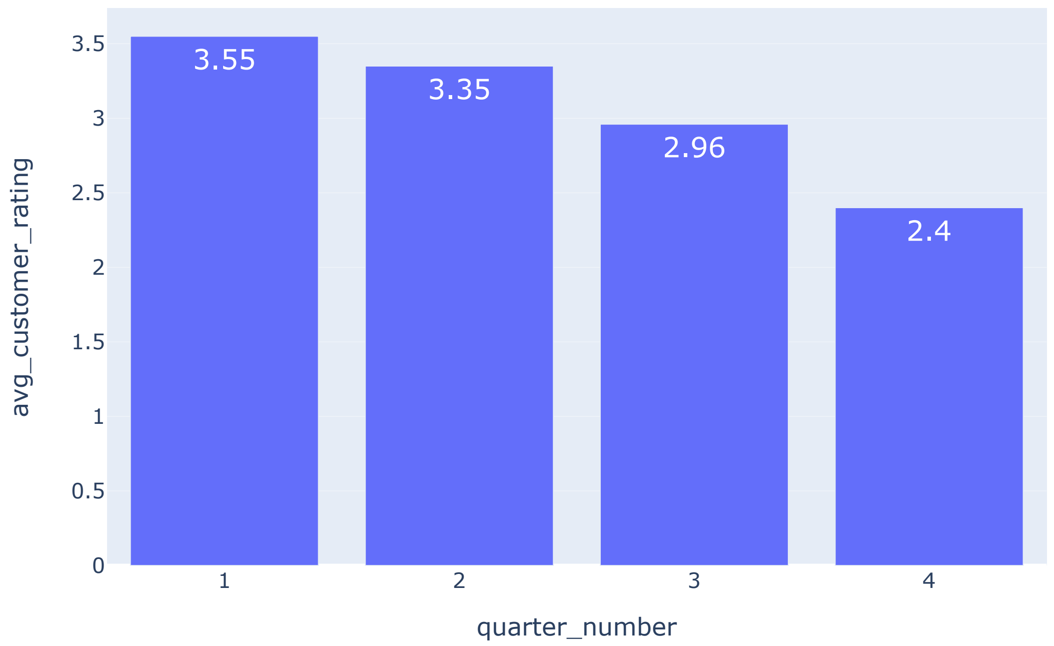
In [4]:
```
q2 = pd.read_csv("q2.csv")
q2.head()
```

Out[4]:

	quarter_number	avg_customer_rating
0	1	3.55
1	2	3.35
2	3	2.96
3	4	2.40

In [5]:
```
fig = px.bar(q2, x="quarter_number", y="avg_customer_rating", title="Average rating in each quarte
fig.update_layout(xaxis=dict(tickmode='array'), autosize=False,
    width=700,
    height=500)
fig.update_traces(textfont_size=16, textangle=0, textposition="inside", cliponaxis=False)
fig.show()
```

Average rating in each quarter



Question 3

Are customers getting more dissatisfied over time?

Hint: Need the percentage of different types of customer feedback in each quarter. Use a common table expression and determine the number of customer feedback in each category as well as the total number of customer feedback in each quarter.

Now use that common table expression to find out the percentage of different types of customer feedback in each quarter. Eg: (total number of very good feedback/total customer feedback)* 100 gives you the percentage of very good feedback.

In [6]:

```
q3 = pd.read_csv("q3.csv")
q3.head()
```

Out[6]:

	quarter_number	orders_count	very_bad_rate_%	bad_rate_%	okay_rate_%	good_rate_%	very_good_rate_%
0	1	310	11.0	11.3	19.0	28.7	30.0
1	2	262	14.9	14.1	20.2	22.1	28.6
2	3	229	17.9	22.7	21.8	21.0	16.6
3	4	199	30.7	29.1	20.1	10.1	10.1

In [7]:

```
q3 = q3.set_index("quarter_number")
q3.head()
```

Out[7]:

quarter_number	orders_count	very_bad_rate_%	bad_rate_%	okay_rate_%	good_rate_%	very_good_rate_%
1	310	11.0	11.3	19.0	28.7	30.0
2	262	14.9	14.1	20.2	22.1	28.6
3	229	17.9	22.7	21.8	21.0	16.6
4	199	30.7	29.1	20.1	10.1	10.1

In [8]:

```
q3_drop=q3.drop(['orders_count'], axis=1)
q3_drop.head()
```

Out[8]:

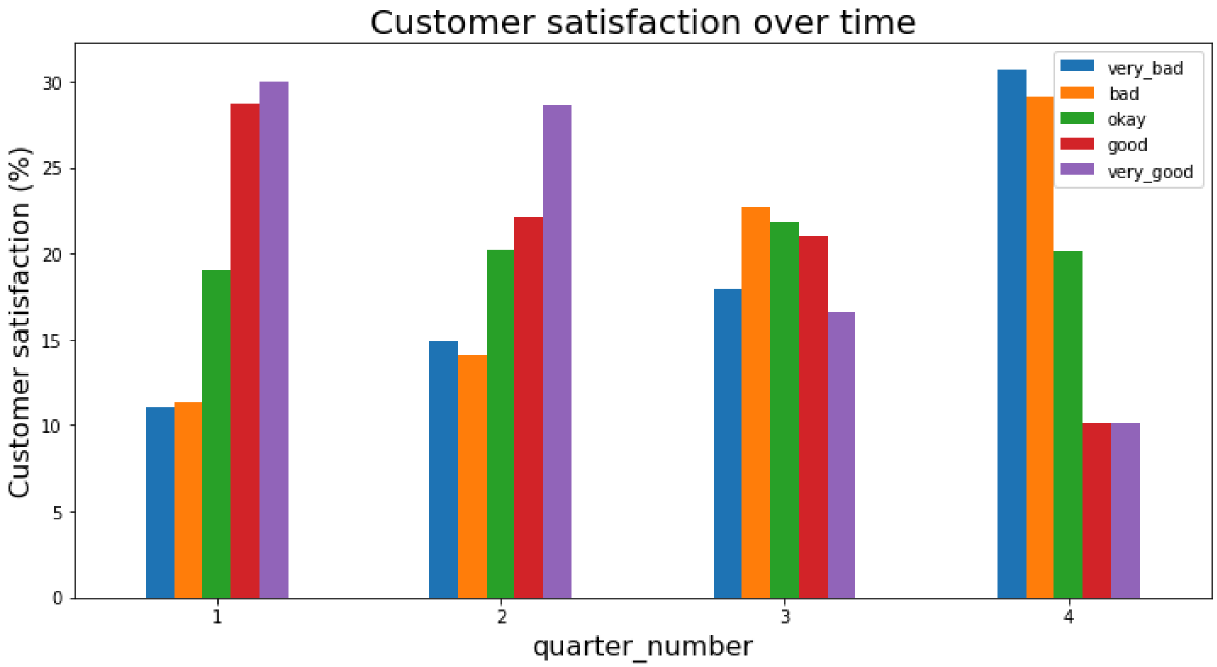
quarter_number	very_bad_rate_%	bad_rate_%	okay_rate_%	good_rate_%	very_good_rate_%
1	11.0	11.3	19.0	28.7	30.0
2	14.9	14.1	20.2	22.1	28.6
3	17.9	22.7	21.8	21.0	16.6
4	30.7	29.1	20.1	10.1	10.1

In [9]:

```
q3_drop.plot.bar(figsize=(12,6))
plt.title('Customer satisfaction over time', fontsize=20)
plt.xlabel('quarter_number', fontsize=16)
plt.ylabel('Customer satisfaction (%)', fontsize=16)
plt.xticks(rotation=0)
location = 0
legend_drawn_flag = True
plt.legend(["very_bad", "bad","okay","good","very_good"], loc=0, frameon=legend_drawn_flag)
```

Out[9]:

```
<matplotlib.legend.Legend at 0x159afe0a9d0>
```



Question 4

Which are the top 5 vehicle makers preferred by the customer?

Hint: For each vehicle make what is the count of the customers.

In [10]:

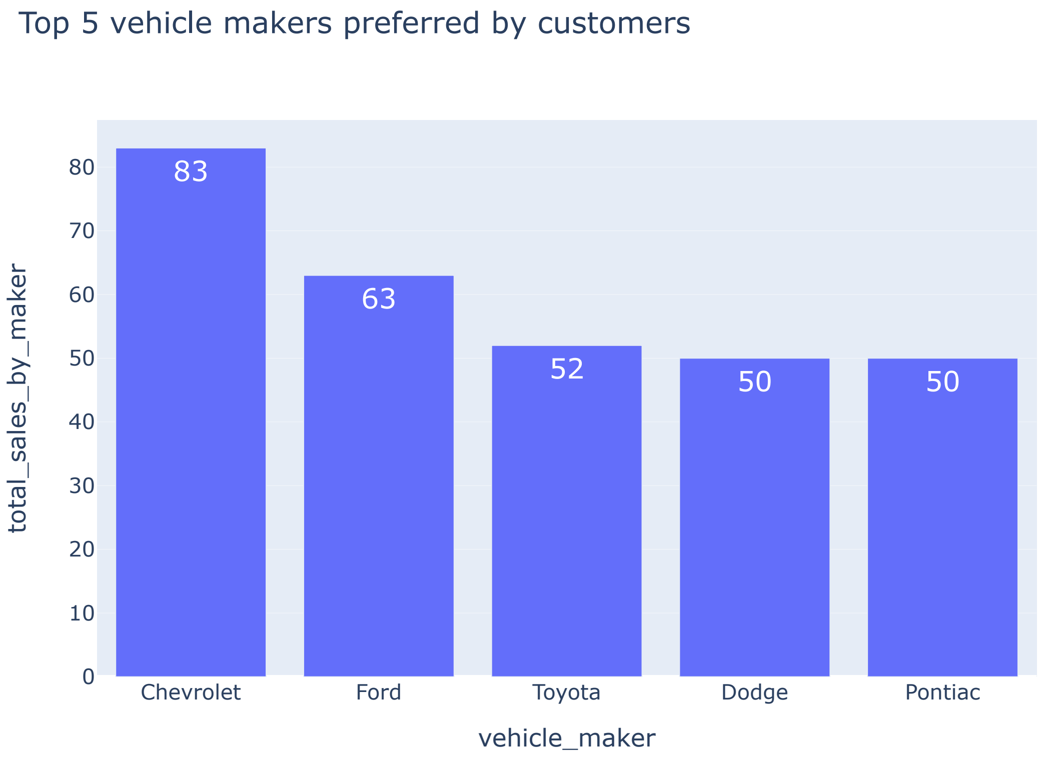
```
q4 = pd.read_csv("q4.csv")
q4.head()
```

Out[10]:

	vehicle_maker	total_sales_by_maker
0	Chevrolet	83
1	Ford	63
2	Toyota	52
3	Dodge	50
4	Pontiac	50

In [11]:

```
fig = px.bar(q4, x="vehicle_maker", y="total_sales_by_maker", title="Top 5 vehicle makers preferre
fig.update_layout(xaxis=dict(tickmode='array'), autosize=False,
    width=700,
    height=500)
fig.update_traces(textfont_size=16, textangle=0, textposition="inside", cliponaxis=False)
fig.show()
```



Question 5

What is the most preferred vehicle make in each state?

Hint: Use the window function RANK() to rank based on the count of customers for each state and vehicle maker. After ranking, take the vehicle maker whose rank is 1.

In [12]:

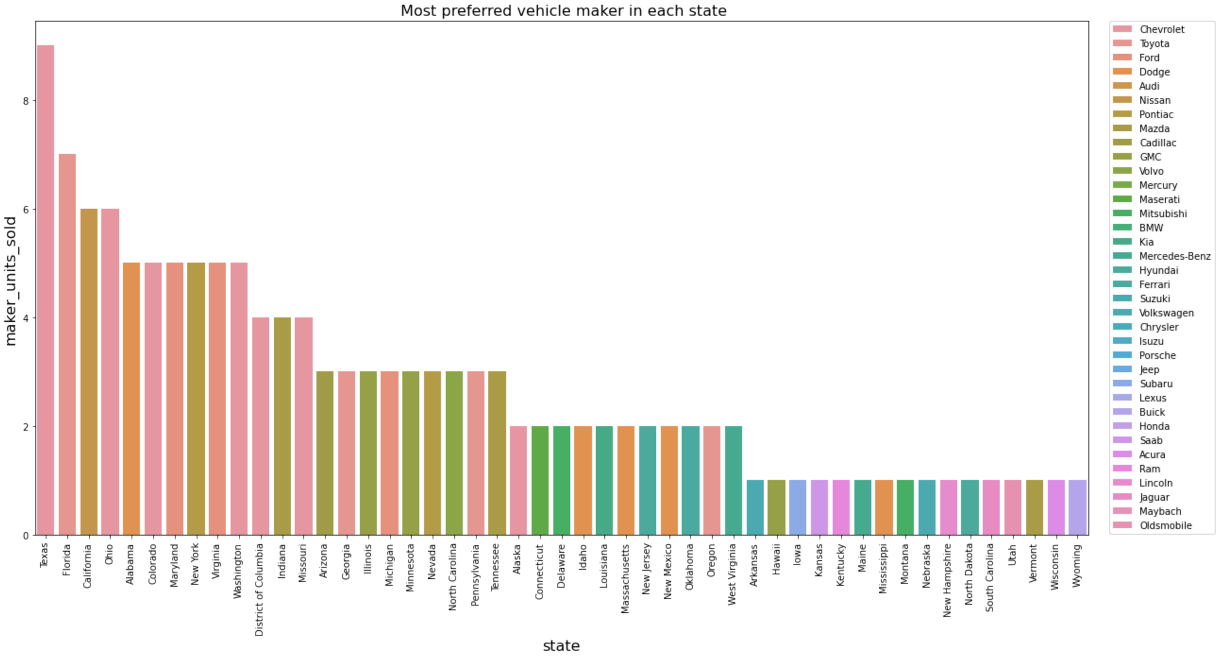
```
q5 = pd.read_csv("q5.csv")
q5.head()
```

Out[12]:

	state	vehicle_maker	maker_units_sold
0	Texas	Chevrolet	9
1	Florida	Toyota	7
2	California	Ford	6
3	California	Dodge	6
4	California	Audi	6

In [13]:

```python
plt.figure(figsize=(20,10))
sns.barplot(x='state', y='maker_units_sold', hue='vehicle_maker', data=q5, dodge=False)
plt.title(' Most preferred vehicle maker in each state', fontsize=16)
plt.xlabel('state', fontsize=16)
plt.ylabel('maker_units_sold', fontsize=16)
# plt.legend(loc='upper right')
plt.legend(bbox_to_anchor=(1.02, 1), loc='upper left', borderaxespad=0)
sns.color_palette("rocket")
plt.xticks(rotation=90)
plt.show()
```



Question 6

What is the trend of number of orders by quarters?

Hint: Count the number of orders for each quarter.

In [14]:

```python
q6 = pd.read_csv("q6.csv")
q6.head()
```

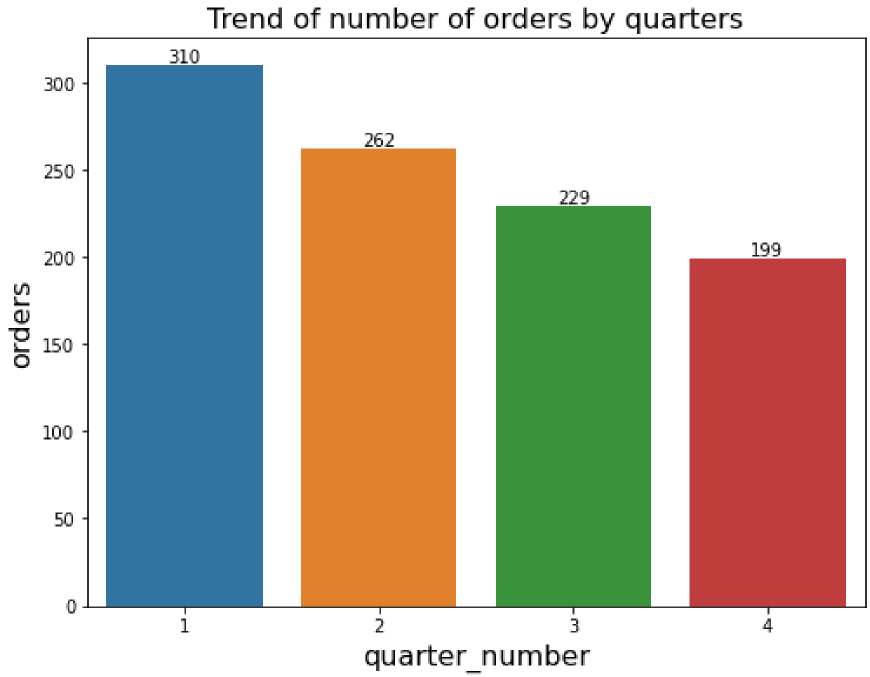
Out[14]:

	quarter_number	orders
0	1	310
1	2	262
2	3	229
3	4	199

In [15]:

```python
plt.figure(figsize=(8,6))
ax=sns.barplot(x='quarter_number', y='orders', data=q6)
plt.title('Trend of number of orders by quarters', fontsize=16)
plt.xlabel('quarter_number', fontsize=16)
plt.ylabel('orders', fontsize=16)
```

```
for i in ax.containers:
    ax.bar_label(i,)
plt.xticks(rotation=0)
plt.show()
```



Question 7

What is the quarter over quarter % change in revenue?

Hint: Quarter over Quarter percentage change in revenue means what is the change in revenue from the subsequent quarter to the previous quarter in percentage.

To calculate you need to use the common table expression to find out the sum of revenue for each quarter.

Then use that CTE along with the LAG function to calculate the QoQ percentage change in revenue.

In [16]:

```
q7 = pd.read_csv("q7.csv")
q7.head()
```

Out[16]:

	quarter_number	revenue	previous_quarter_revenue	QUARTER OVER QUARTER REVENUE(%)
0	1	21605081.10	NaN	NaN
1	2	19790742.08	21605081.10	-8.4
2	3	20553128.72	19790742.08	3.9
3	4	14922858.95	20553128.72	-27.4

In [17]:

```
q7_drop=q7.drop(['revenue','previous_quarter_revenue'], axis=1)
q7_drop.head()
```

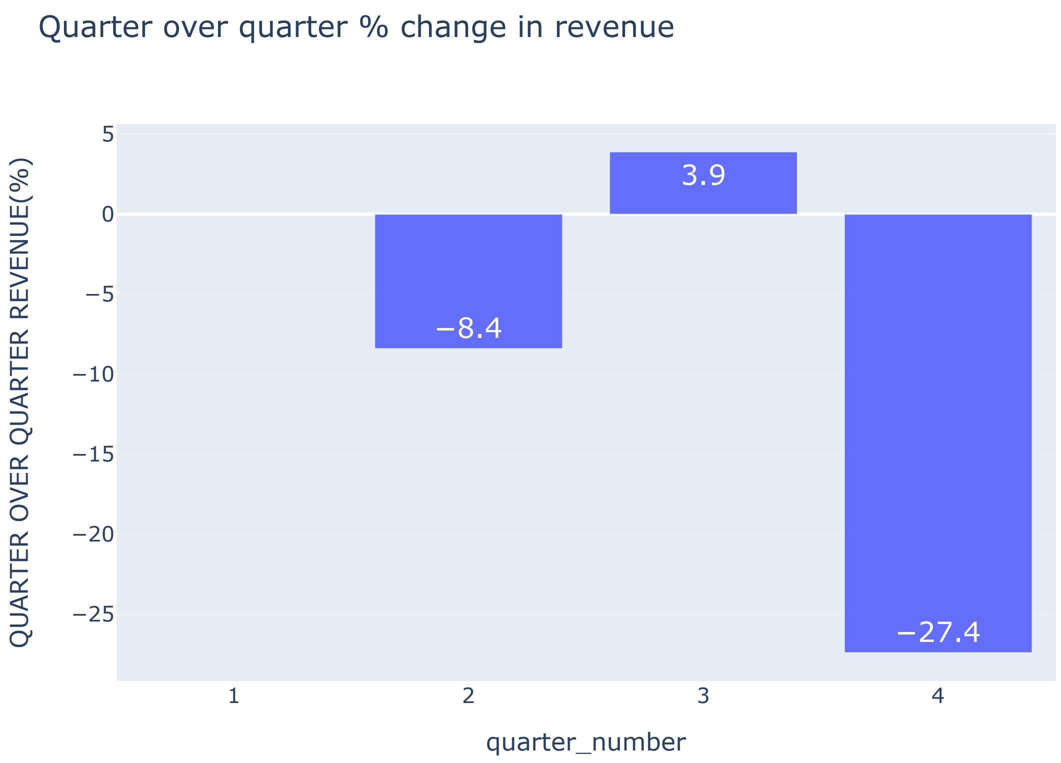
Out[17]:

	quarter_number	QUARTER OVER QUARTER REVENUE(%)
0	1	NaN
1	2	-8.4
2	3	3.9
3	4	-27.4

In [18]:

```
fig = px.bar(q7_drop, x="quarter_number", y="QUARTER OVER QUARTER REVENUE(%)", title="Quarter over
fig.update_layout(xaxis=dict(tickmode='array'), autosize=False,
    width=700,
    height=500)
```

```
fig.update_traces(textfont_size=16, textangle=0, textposition="inside", cliponaxis=False)
fig.show()
```



Question 8

What is the trend of revenue and orders by quarters?

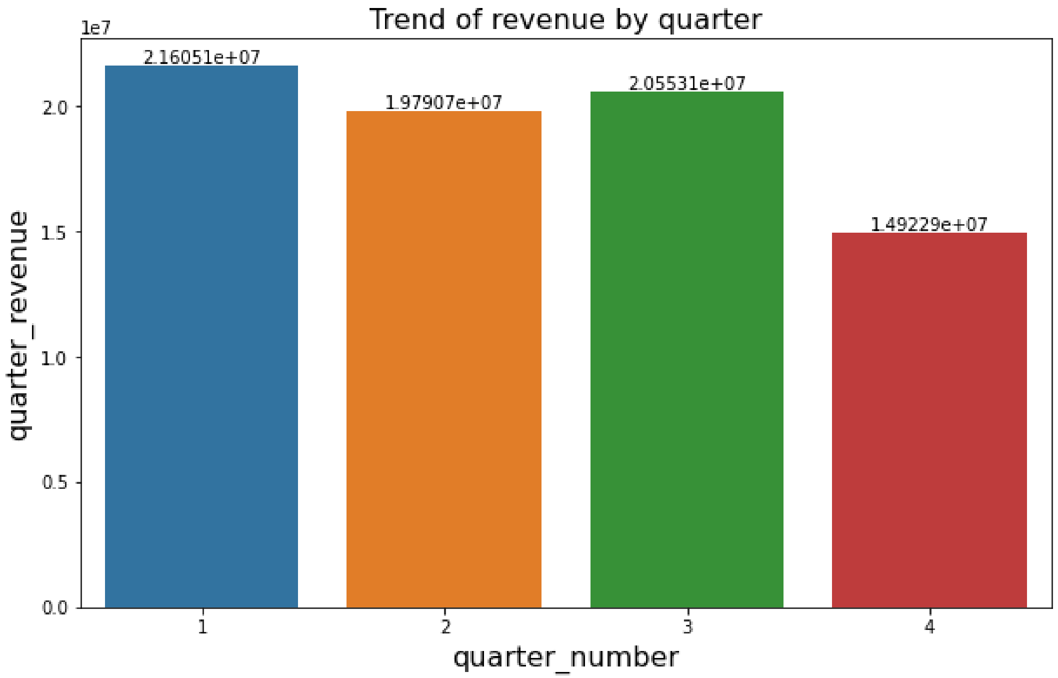
Hint: Find out the sum of revenue and count the number of orders for each quarter.

In [19]:
```
q8 = pd.read_csv("q8.csv")
q8.head()
```

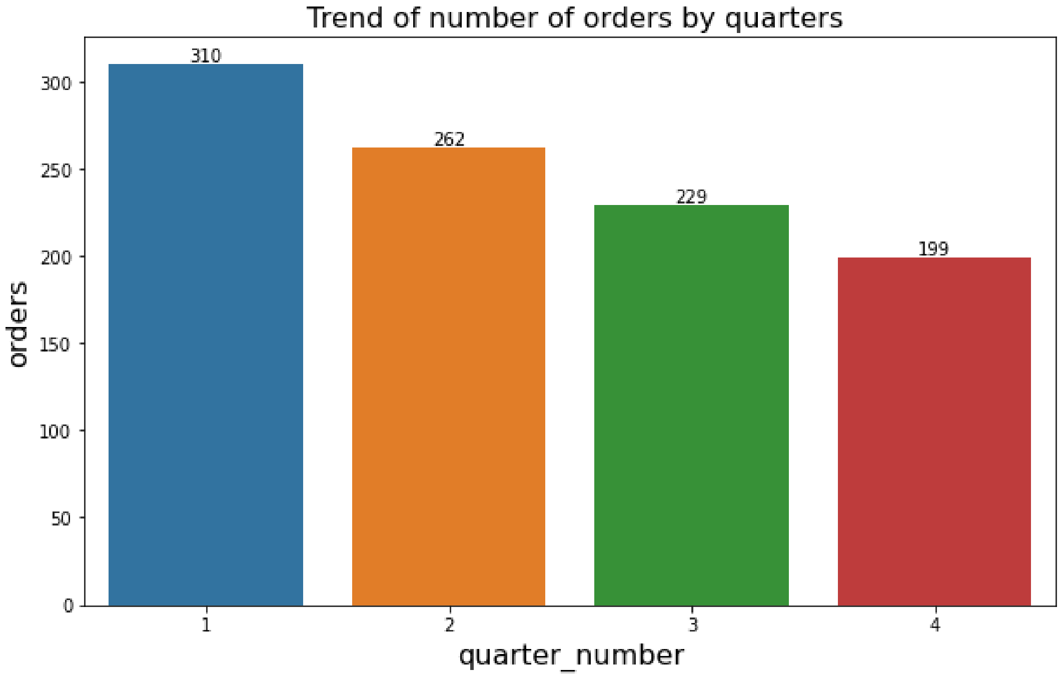
Out[19]:

	quarter_number	quarter_revenue	quarter_orders
0	1	21605081	310
1	2	19790742	262
2	3	20553129	229
3	4	14922859	199

In [20]:
```
plt.figure(figsize=(10,6))
ax=sns.barplot(x='quarter_number', y='quarter_revenue', data=q8)
plt.title('Trend of revenue by quarter', fontsize=16)
plt.xlabel('quarter_number', fontsize=16)
plt.ylabel('quarter_revenue', fontsize=16)
for i in ax.containers:
    ax.bar_label(i,)
plt.xticks(rotation=0)
plt.show()
```



In [21]:
```
plt.figure(figsize=(10,6))
ax=sns.barplot(x='quarter_number', y='quarter_orders', data=q8)
plt.title(' Trend of number of orders by quarters', fontsize=16)
plt.xlabel('quarter_number', fontsize=16)
plt.ylabel('orders', fontsize=16)
for i in ax.containers:
    ax.bar_label(i,)
plt.xticks(rotation=0)
plt.show()
```



Question 9

What is the average discount offered for different types of credit cards?

Hint: Find out the average of discount for each credit card type.

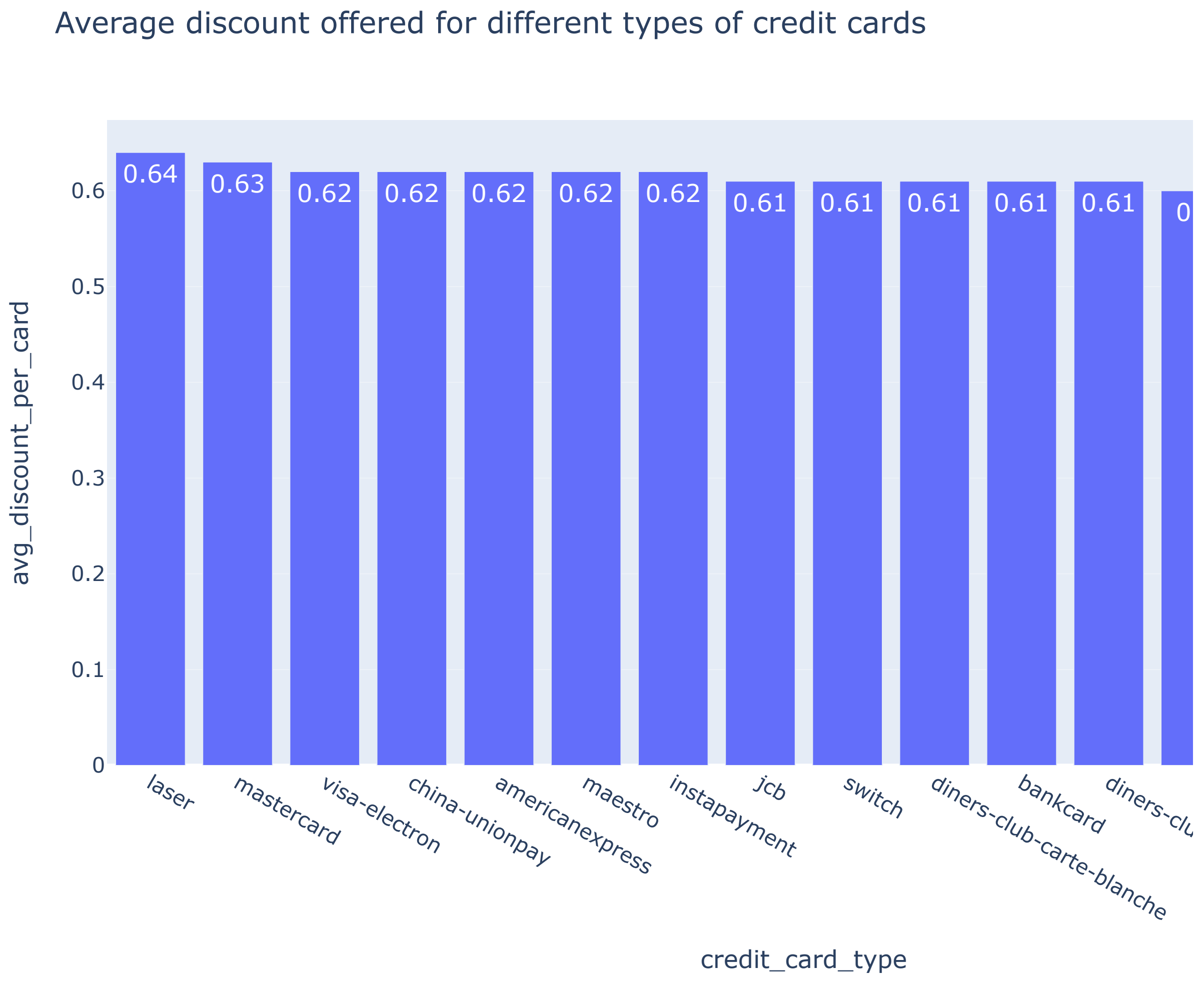
In [22]:
```
q9 = pd.read_csv("q9.csv")
q9.head()
```

Out[22]:

	credit_card_type	avg_discount_per_card
0	laser	0.64
1	mastercard	0.63
2	visa-electron	0.62
3	china-unionpay	0.62
4	americanexpress	0.62

In [23]:

```
fig = px.bar(q9, x="credit_card_type", y="avg_discount_per_card", title="Average discount offered
fig.update_layout(xaxis=dict(tickmode='array'), autosize=False,
    width=1000,
    height=600)
fig.update_traces(textfont_size=14, textangle=0, textposition="inside", cliponaxis=False)
fig.show()
```


Average discount offered for different types of credit cards

Question 10

What is the average time taken to ship the placed orders for each quarters?

Hint: For each quarter, find out the average of the function that you created to calculate the difference between the ship date and the order date.

In [24]:

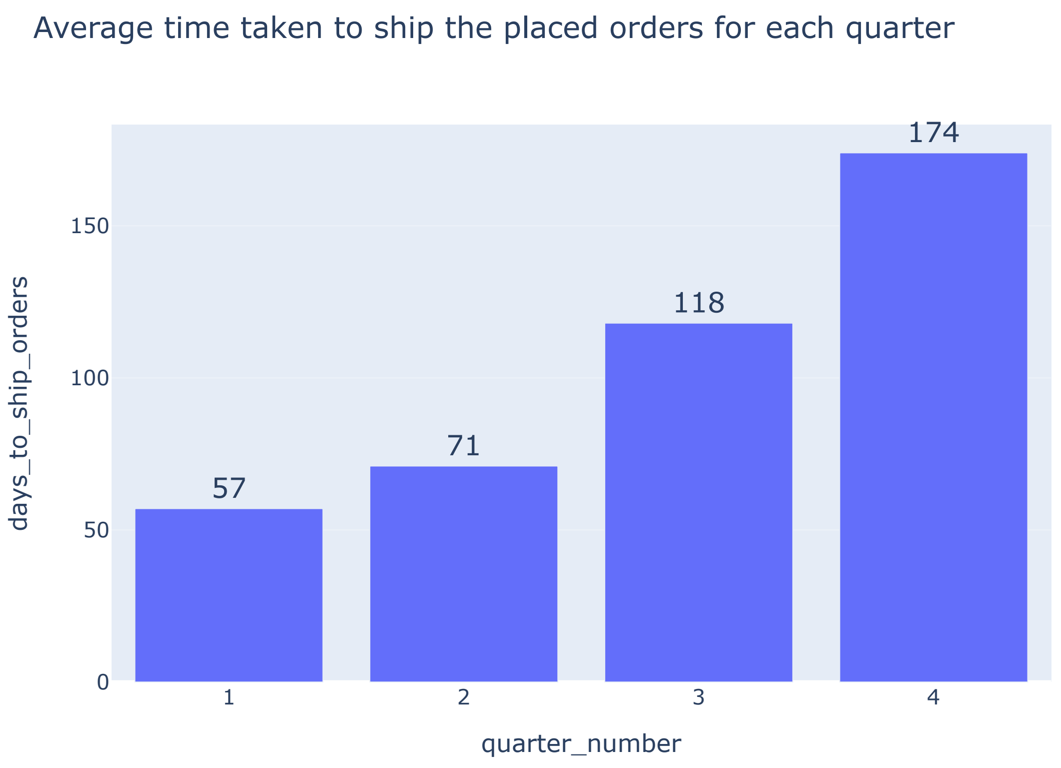
```
q10 = pd.read_csv("q10.csv")
q10.head()
```

Out[24]:

	quarter_number	days_to_ship_orders
0	1	57
1	2	71
2	3	118
3	4	174

In [25]:

```
fig = px.bar(q10, x="quarter_number", y="days_to_ship_orders", title="Average time taken to ship t
fig.update_layout(xaxis=dict(tickmode='array'), autosize=False,
    width=700,
    height=500)
fig.update_traces(textfont_size=16, textangle=0, textposition="outside", cliponaxis=False)
fig.show()
```


Average time taken to ship the placed orders for each quarter
days_to_ship_orders
quarter_number