

```
In [1]: pip install mysql-connector-python
Requirement already satisfied: mysql-connector-python in c:\users\paka\anaconda3\lib\site-packages (8.0.31)
Requirement already satisfied: protobuf<=3.20.1,>=3.11.0 in c:\users\paka\anaconda3\lib\site-packages (from mysql-connector-python) (3.20.1)
Note: you may need to restart the kernel to use updated packages.
```

```
In [2]: import mysql.connector
from mysql.connector import Error
import pandas as pd
import numpy as np
import seaborn as sns
import plotly.express as px
import matplotlib.pyplot as plt
```

```
In [3]: connection = mysql.connector.connect(host='localhost',
                                             database='store_management_system',
                                             user='root',
                                             password='root')
```

```
In [4]: select_sql_query="show tables"
cursor = connection.cursor()
cursor.execute(select_sql_query)
records = cursor.fetchall()
for x in records:
    print(x, "\n")

('category',)

('customer',)

('customer_product',)

('employee',)

('invoice',)

('order_items',)

('orders',)

('orders1',)

('orders2',)

('plan',)

('plan_promotions',)

('product',)

('promotions',)

('store',)

('store_supplier',)

('supplier',)
```

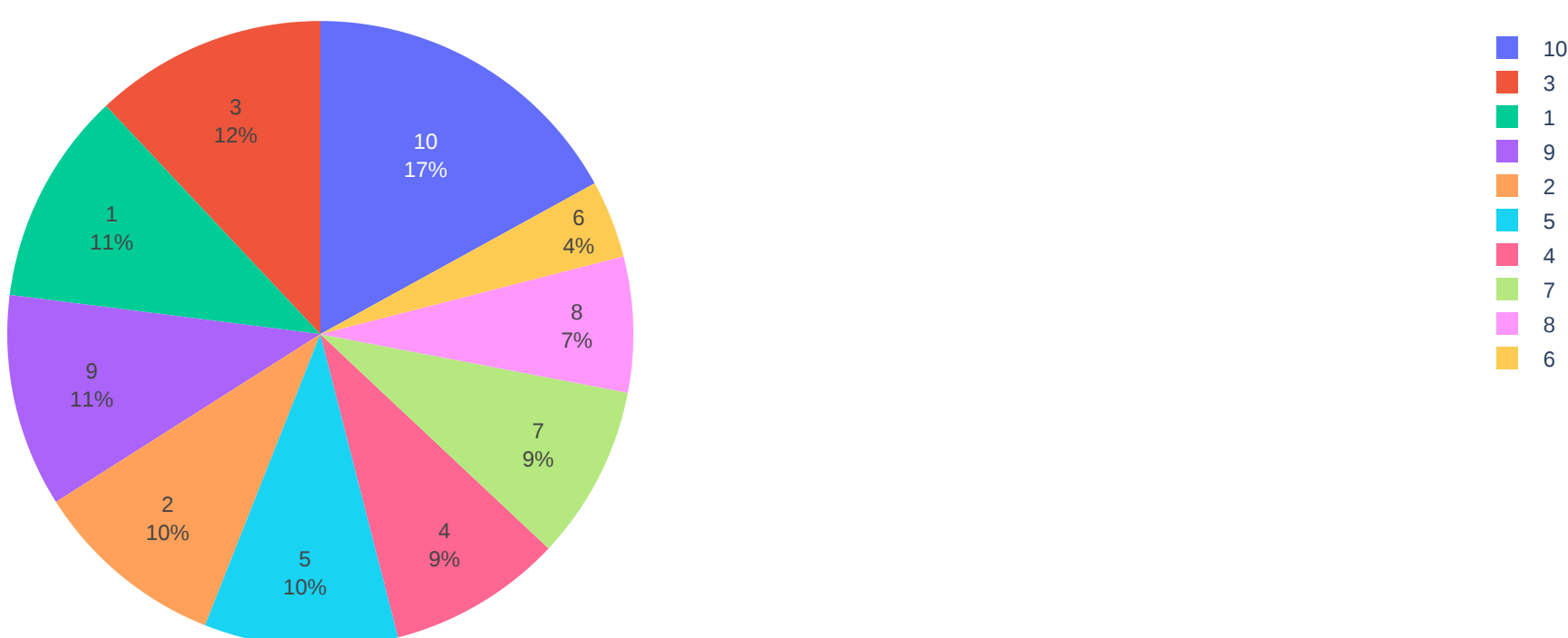
```
In [5]: sql_select_Query = 'SELECT store_id,count(store_id) as Number_of_employees from employee group by store_id order by Number_of_employees desc'
cursor = connection.cursor()
cursor.execute(sql_select_Query)
records = cursor.fetchall()
df_no_of_employees = pd.DataFrame(records, columns = ['store_id', 'Number_of_employees'])
df_no_of_employees
```

	store_id	Number_of_employees
0	10	17
1	3	12
2	1	11
3	9	11
4	2	10
5	5	10
6	4	9
7	7	9
8	8	7
9	6	4

```
In [6]: fig = px.pie(df_no_of_employees, values = 'Number_of_employees', names = 'store_id', title ='Percentage distribution of Number of employees working in stores')
fig.update_traces(textposition = 'inside', textinfo='percent+label')
fig.show()
```



Percentage distribution of Number of employees working in stores



```
In [7]: sql_select_Query = 'select month(order_date) month,count(order_id) c from orders2 where year(ORDER_DATE)=2021 group by month(order_date) order by month'
cursor = connection.cursor()
cursor.execute(sql_select_Query)
records = cursor.fetchall()
df_orders_2021 = pd.DataFrame(records, columns = ['Month', 'Number_of_orders'])
sql_select_Query1 = 'select month(order_date) month,count(order_id) c from orders1 where year(ORDER_DATE)=2022 group by month(order_date) order by month'
cursor = connection.cursor()
cursor.execute(sql_select_Query1)
records = cursor.fetchall()
df_orders_2022 = pd.DataFrame(records, columns = ['Month', 'Number_of_orders'])
```

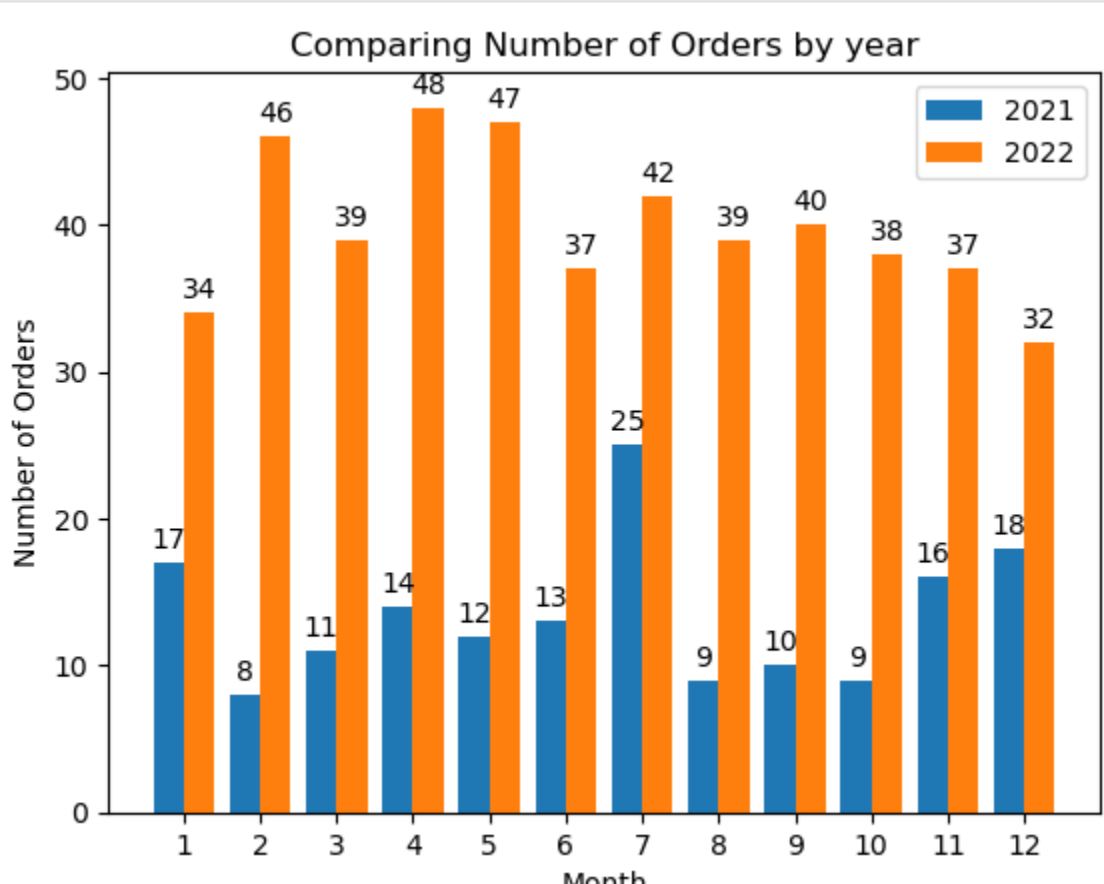
```
In [8]: df_orders_2021
```

	Month	Number_of_orders
0	1	17
1	2	8
2	3	11
3	4	14
4	5	12
5	6	13
6	7	25
7	8	9
8	9	10
9	10	9
10	11	16
11	12	18

```
In [9]: df_orders_2022
```

	Month	Number_of_orders
0	1	34
1	2	46
2	3	39
3	4	48
4	5	47
5	6	37
6	7	42
7	8	39
8	9	40
9	10	38
10	11	37
11	12	32

```
In [10]: x_axis = np.arange(len(df_orders_2022['Month']))
plt1=plt.bar(x_axis-0.2, df_orders_2021['Number_of_orders'],0.4, label='2021')
plt2=plt.bar(x_axis+0.2, df_orders_2022['Number_of_orders'],0.4, label='2022')
plt.xticks(x_axis,df_orders_2022['Month'])
plt.xlabel("Month")
plt.ylabel("Number of Orders")
plt.title("Comparing Number of Orders by year")
plt.legend()
plt.bar_label(plt1, padding=3)
plt.bar_label(plt2, padding=3)
plt.show()
```

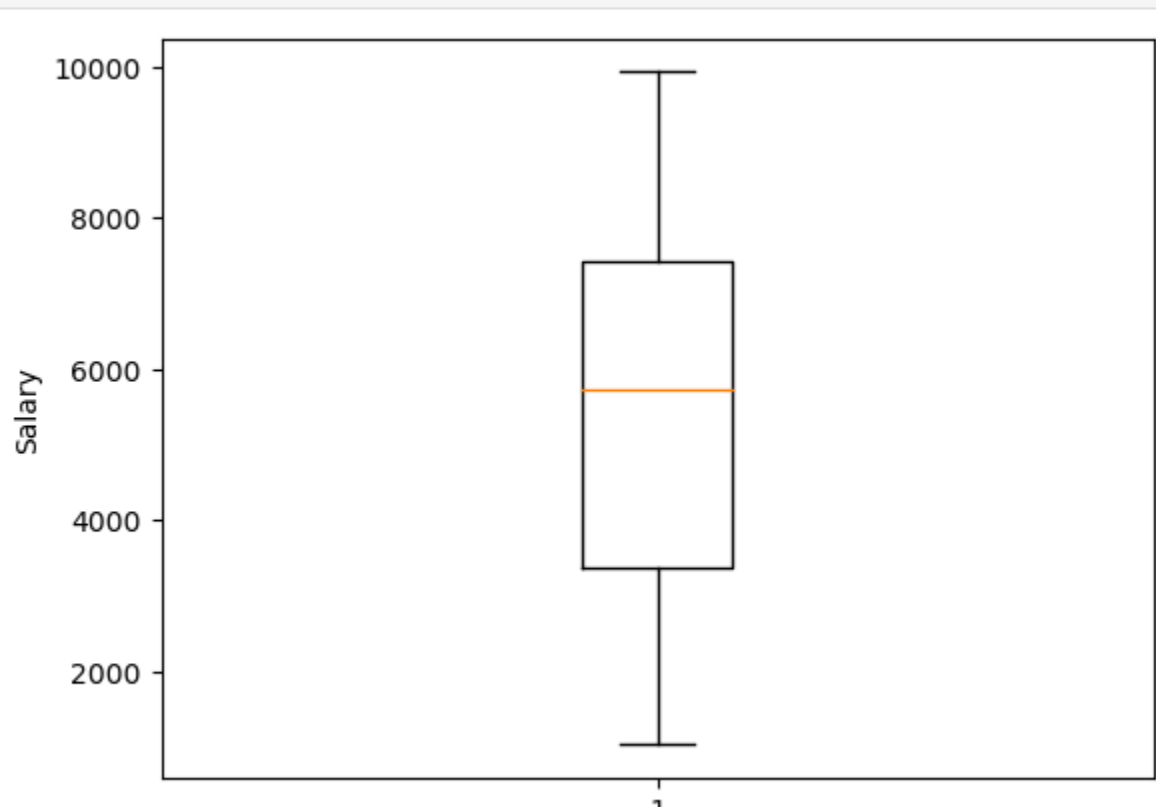


```
In [11]: sql_select_Query = 'SELECT SALARY from employee'
cursor = connection.cursor()
cursor.execute(sql_select_Query)
records = cursor.fetchall()
df_salary = pd.DataFrame(records, columns = ['SALARY'])
df_salary
```

	SALARY
0	6874
1	3871
2	4645
3	1044
4	4346
...	...
95	1030
96	9356
97	3453
98	5581
99	4864

100 rows × 1 columns

```
In [12]: plt.boxplot(df_salary)
plt.ylabel('Salary')
plt.show()
```



```
In [ ]:
```

```
In [ ]:
```