INTRODUCTION

As a BI analyst, I carried out an analysis on Plato's pizza sales data for the year 2015 to enable the organization gain insight and make better driven decision for next year sales. I was provided with a dataset by the company which was needed to be cleaned. After cleaning the data, I transformed, model the data and derived some insightful analysis for Plato's pizza to prepare them ahead of next year.

Import the necessary modules

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_palette('rainbow')
sns.set_style('darkgrid')
```

Import the pizza dataset

```
order_details=pd.read_csv('C:/Users/pc/Documents/BI CHALLENGE/Pizza+Place+Sales/pizza_sales/order_details.csv')
order=pd.read_csv('C:/Users/pc/Documents/BI CHALLENGE/Pizza+Place+Sales/pizza_sales/orders.csv')
pizza_types=pd.read_csv('C:/Users/pc/Documents/BI CHALLENGE/Pizza+Place+Sales/pizza_sales/pizza_types.csv',encoding="ISO-8859-1")
pizza=pd.read_csv('C:/Users/pc/Documents/BI CHALLENGE/Pizza+Place+Sales/pizza_sales/pizzas.csv',encoding="ISO-8859-1")
```

Show the dataset columns

```
order_details.columns
Index(['order_details_id', 'order_id', 'pizza_id', 'quantity'], dtype='object')
order.columns
Index(['order_id', 'date', 'time'], dtype='object')
pizza_types.columns
Index(['pizza_type_id', 'name', 'category', 'ingredients'], dtype='object')
pizza.columns
Index(['pizza_id', 'pizza_type_id', 'size', 'price'], dtype='object')
```

Merge the dataset

```
pizza_orders=order_details.merge(order, on='order_id',how='left')\
              .merge(pizza,on='pizza_id',how='left')\
              .merge(pizza_types,on='pizza_type_id')
```

The merged dataset output

```
pizza_orders
       order_details_id  ...                                        ingredients
0                     1  ...          Sliced Ham, Pineapple, Mozzarella Cheese
1                    94  ...          Sliced Ham, Pineapple, Mozzarella Cheese
2                   110  ...          Sliced Ham, Pineapple, Mozzarella Cheese
3                   125  ...          Sliced Ham, Pineapple, Mozzarella Cheese
4                   175  ...          Sliced Ham, Pineapple, Mozzarella Cheese
...                 ...  ...                                               ...
48615             48176  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...
48616             48244  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...
48617             48311  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...
48618             48456  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...
48619             48527  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...

[48620 rows x 12 columns]
```

Display the pizza_orders columns

```
pizza_orders.columns
Index(['order_details_id', 'order_id', 'pizza_id', 'quantity', 'date', 'time',
       'pizza_type_id', 'size', 'price', 'name', 'category', 'ingredients'],
      dtype='object')
```

Drop duplicate rows from the pizza_orders

```
pizza_orders.drop_duplicates()
       order_details_id  ...                                        ingredients
0                     1  ...           Sliced Ham, Pineapple, Mozzarella Cheese
1                    94  ...           Sliced Ham, Pineapple, Mozzarella Cheese
2                   110  ...           Sliced Ham, Pineapple, Mozzarella Cheese
3                   125  ...           Sliced Ham, Pineapple, Mozzarella Cheese
4                   175  ...           Sliced Ham, Pineapple, Mozzarella Cheese
...                 ...  ...                                               ...
48615             48176  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...
48616             48244  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...
48617             48311  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...
48618             48456  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...
48619             48527  ...  Brie Carre Cheese, Prosciutto, Caramelized Oni...

[48620 rows x 12 columns]
```

Check for null value in the dataset

```
pizza_orders.isna().sum()
order_details_id    0
order_id            0
pizza_id            0
quantity            0
date                0
time                0
pizza_type_id       0
size                0
price               0
name                0
category            0
ingredients         0
Month               0
Week                0
dtype: int64
```

Print the pizza_orders info

```
pizza_orders.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 48620 entries, 0 to 48619
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   order_details_id  48620 non-null  int64
 1   order_id          48620 non-null  int64
 2   pizza_id          48620 non-null  object
 3   quantity          48620 non-null  int64
 4   date              48620 non-null  object
 5   time              48620 non-null  object
 6   pizza_type_id     48620 non-null  object
 7   size              48620 non-null  object
 8   price             48620 non-null  float64
 9   name              48620 non-null  object
 10  category          48620 non-null  category
 11  ingredients       48620 non-null  object
dtypes: category(1), float64(1), int64(3), object(7)
memory usage: 4.5+ MB
```

Statistics Summary of the pizza_orders

```
pizza_orders.describe()
       order_details_id      order_id      quantity         price
count      48620.000000  48620.000000  48620.000000  48620.000000
mean       24310.500000  10701.479761      1.019622     16.494132
std        14035.529381   6180.119770      0.143077      3.621789
min            1.000000      1.000000      1.000000      9.750000
25%        12155.750000   5337.000000      1.000000     12.750000
50%        24310.500000  10682.500000      1.000000     16.500000
75%        36465.250000  16100.000000      1.000000     20.250000
max        48620.000000  21350.000000      4.000000     35.950000
```

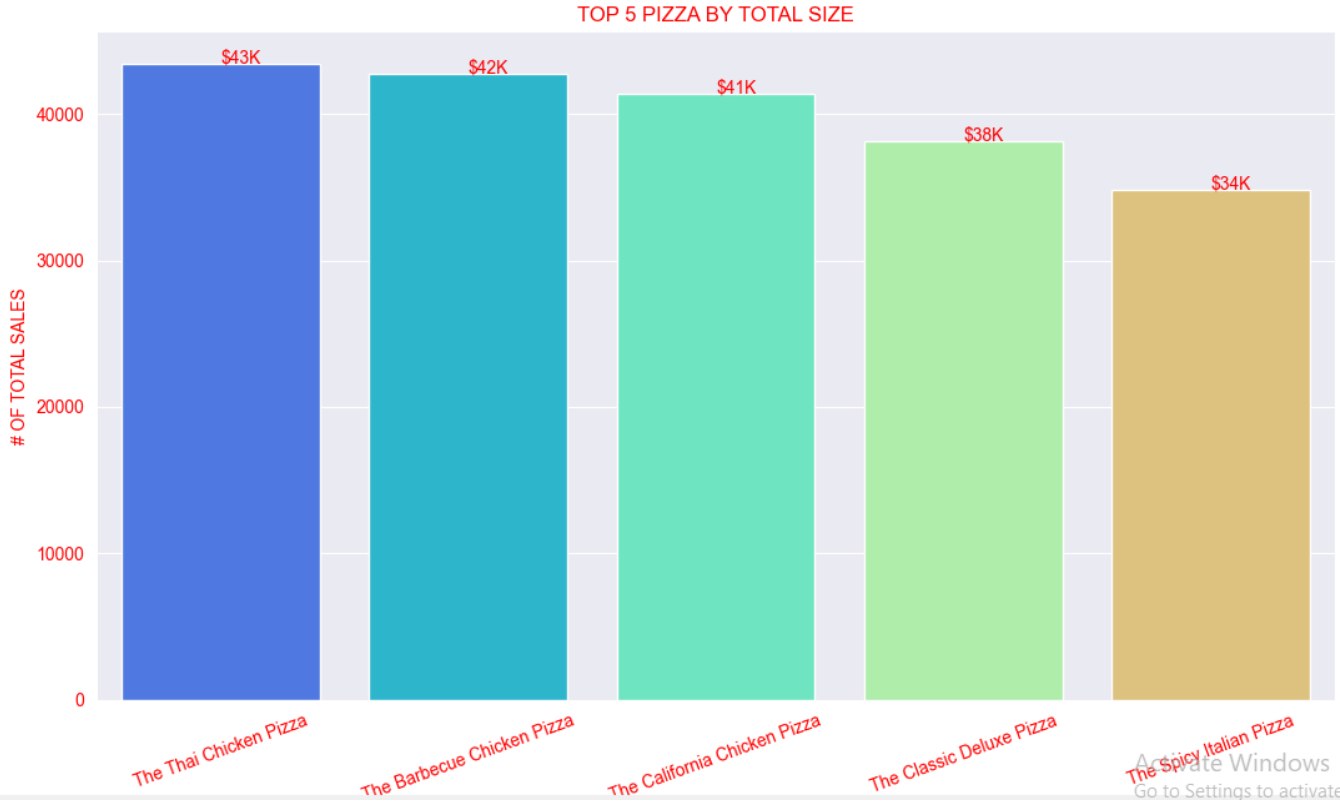
Save the pizza_orders to a csv format

```
pizza_orders.to_csv('pizza_orders.csv')
```

What are the top 5 pizza by total sales?

```
pizza_sales=pizza_orders.groupby('name')['Total_sales'].sum().sort_values(ascending=False).reset_index().iloc[:5]
pizza_sales
                         name  Total_sales
0        The Thai Chicken Pizza     43434.25
1    The Barbecue Chicken Pizza     42768.00
2  The California Chicken Pizza     41409.50
3      The Classic Deluxe Pizza     38180.50
4       The Spicy Italian Pizza     34831.25
```

```
p=sns.barplot(x='name',y='Total_sales',data=pizza_sales)
p.set_title('TOP 5 PIZZA BY TOTAL SIZE',color='r')
Text(0.5, 1.0, 'TOP 5 PIZZA BY TOTAL SIZE')
p.set_ylabel('# OF TOTAL SALES',color='r')
Text(0, 0.5, '# OF TOTAL SALES')
p.tick_params('x',colors='r')
p.tick_params('y',colors='r')
p.set_xticklabels(pizza_sales['name'],rotation=20)
[Text(0, 0, 'The Thai Chicken Pizza'), Text(1, 0, 'The Barbecue Chicken Pizza'), Text(2, 0, 'The California Chicken Pizza'), Text(3, 0, 'The Classic Del
uxe Pizza'), Text(4, 0, 'The Spicy Italian Pizza')]
p.text(0,43434.25,'$43K',color='r')
Text(0, 43434.25, '$43K')
p.text(1,42768.00,'$42K',color='r')
Text(1, 42768.0, '$42K')
p.text(2,41409.50,'$41K',color='r')
Text(2, 41409.5, '$41K')
p.text(3,38180.50,'$38K',color='r')
Text(3, 38180.5, '$38K')
p.text(4,34831.25,'$34K',color='r')
Text(4, 34831.25, '$34K')
plt.show()
```

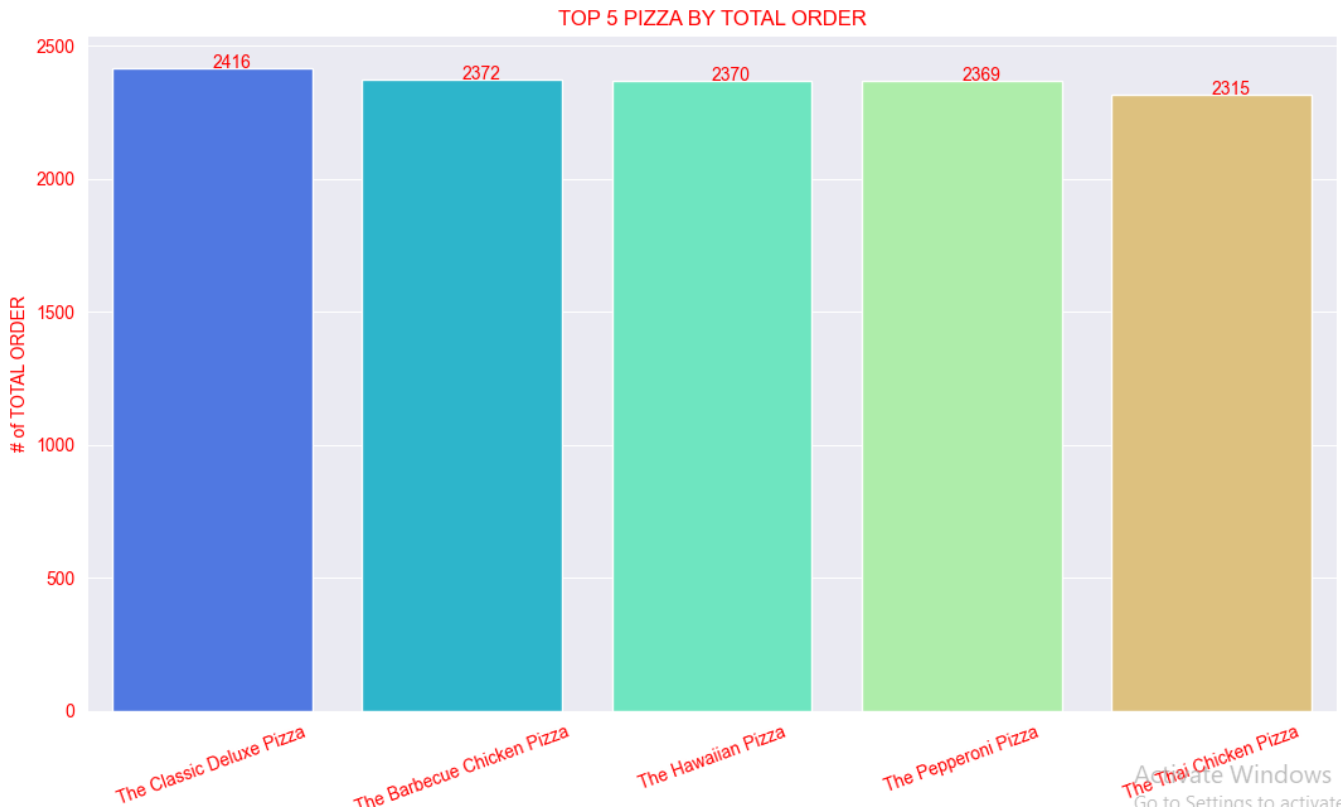



What are the top 5 most ordered pizza?

```
pizza_order=pizza_orders['name'].value_counts().sort_values(ascending=False).reset_index().iloc[:5]
pizza_order
```

	index	name
0	The Classic Deluxe Pizza	2416
1	The Barbecue Chicken Pizza	2372
2	The Hawaiian Pizza	2370
3	The Pepperoni Pizza	2369
4	The Thai Chicken Pizza	2315

```
o=sns.barplot(x='index',y='name',data=pizza_order)
o.set_title('TOP 5 PIZZA BY TOTAL ORDER',color='r')
Text(0.5, 1.0, 'TOP 5 PIZZA BY TOTAL ORDER')
o.set_ylabel('# of TOTAL ORDER',color='r')
Text(0, 0.5, '# of TOTAL ORDER')
o.tick_params('x',colors='r')
o.tick_params('y',colors='r')
o.set_xticklabels(pizza_order['index'],rotation=20)
[Text(0, 0, 'The Classic Deluxe Pizza'), Text(1, 0, 'The Barbecue Chicken Pizza'), Text(2, 0, 'The Hawaiian Pizza'), Text(3, 0, 'The Pepperoni Pizza'),
Text(4, 0, 'The Thai Chicken Pizza')]
o.text(0,2416,2416,color='r')
Text(0, 2416, '2416')
o.text(1,2372,2372,color='r')
Text(1, 2372, '2372')
o.text(2,2370,2370,color='r')
Text(2, 2370, '2370')
o.text(3,2369,2369,color='r')
Text(3, 2369, '2369')
o.text(4,2315,2315,color='r')
Text(4, 2315, '2315')
plt.show()
```

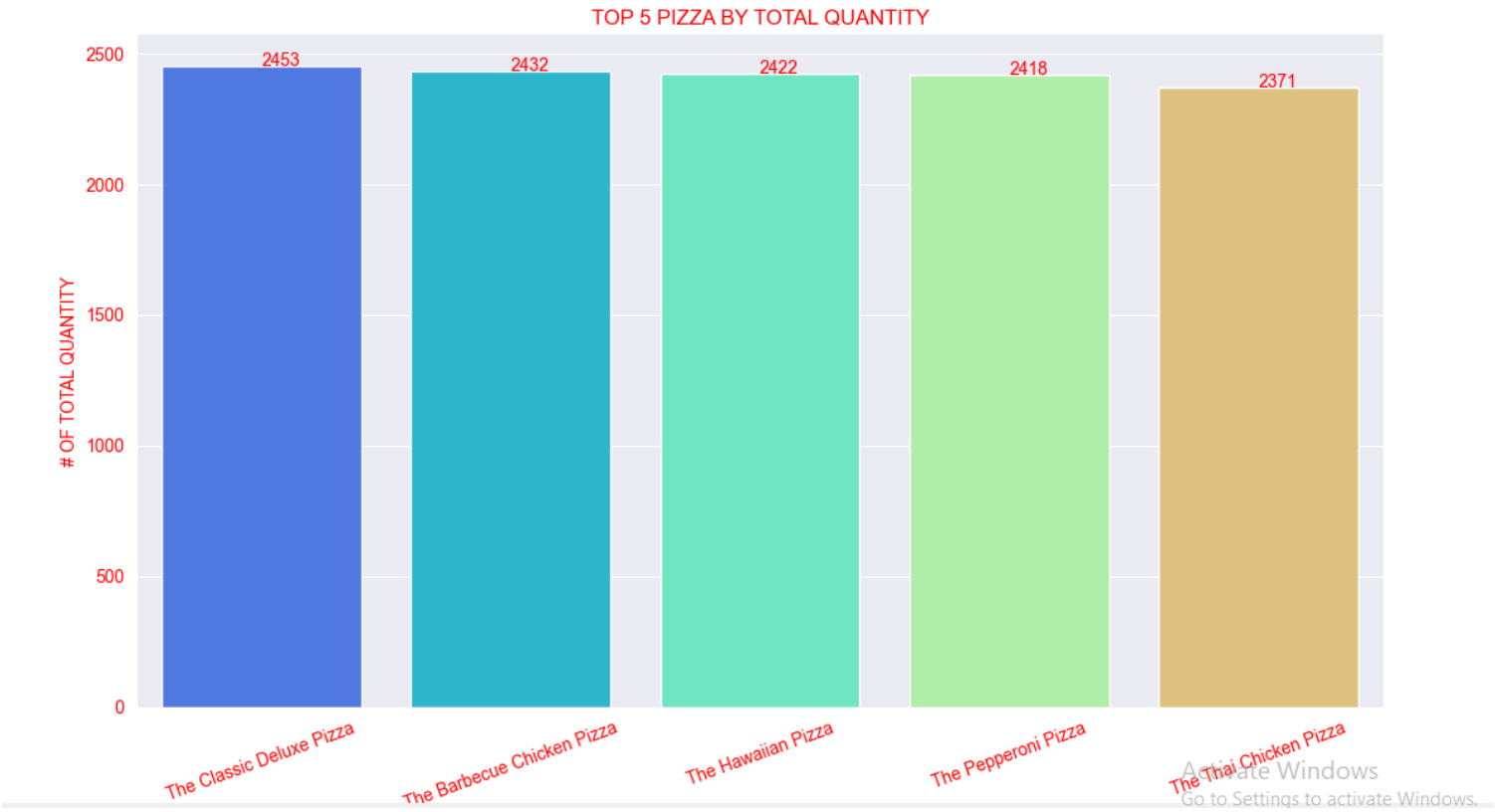



Display the top 5 pizza by quantity

```
pizza_quantity=pizza_orders.groupby('name')['quantity'].sum().sort_values(ascending=False).reset_index().iloc[:5]
pizza_quantity
```

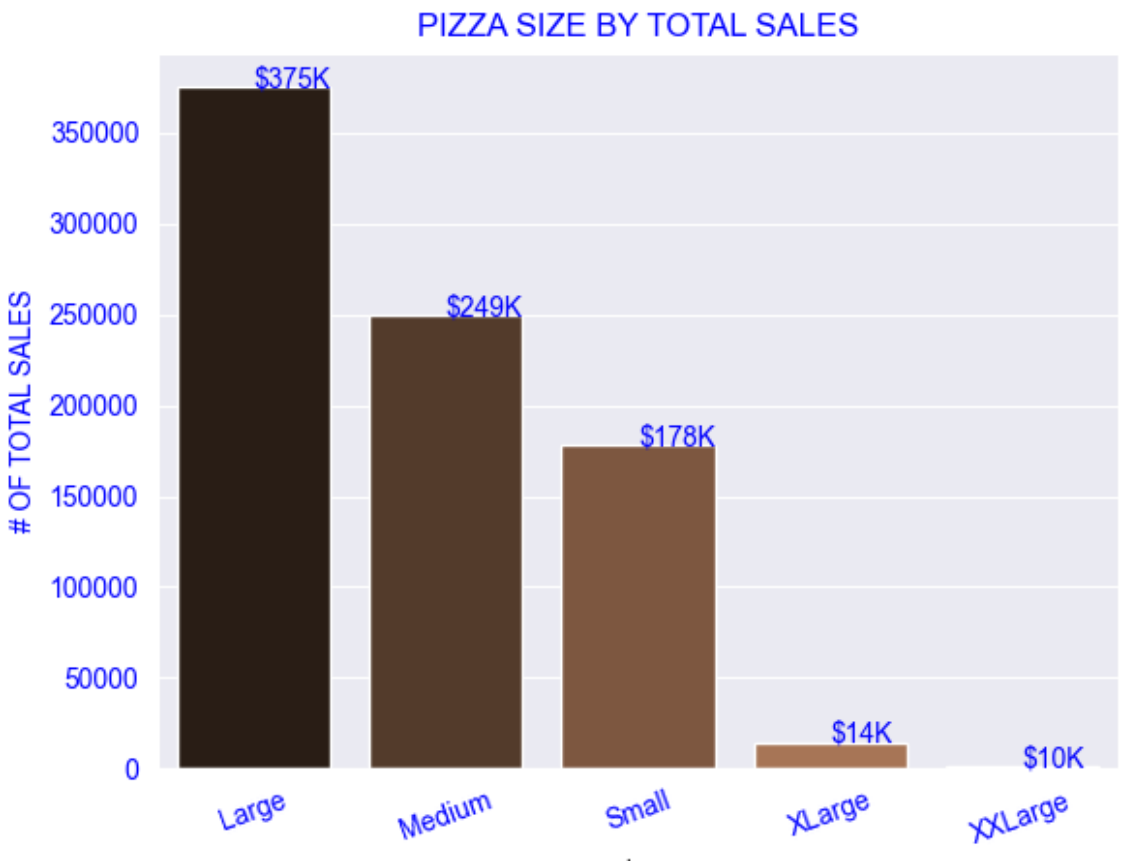
	name	quantity
0	The Classic Deluxe Pizza	2453
1	The Barbecue Chicken Pizza	2432
2	The Hawaiian Pizza	2422
3	The Pepperoni Pizza	2418
4	The Thai Chicken Pizza	2371

```
q=sns.barplot(x='name',y='quantity',data=pizza_quantity)
q.set_title('TOP 5 PIZZA BY TOTAL QUANTITY',color='r')
Text(0.5, 1.0, 'TOP 5 PIZZA BY TOTAL QUANTITY')
q.set_ylabel('# OF TOTAL QUANTITY',color='r')
Text(0, 0.5, '# OF TOTAL QUANTITY')
q.tick_params('x',colors='r')
q.tick_params('y',colors='r')
q.set_xticklabels(pizza_quantity['name'],rotation=20)
[Text(0, 0, 'The Classic Deluxe Pizza'), Text(1, 0, 'The Barbecue Chicken Pizza'), Text(2, 0, 'The Hawaiian Pizza'), Text(3, 0, 'The Pepperoni Pizza'),
Text(4, 0, 'The Thai Chicken Pizza')]
q.text(0,2453,2453,color='r')
Text(0, 2453, '2453')
q.text(1,2432,2432,color='r')
Text(1, 2432, '2432')
q.text(2,2422,2422,color='r')
Text(2, 2422, '2422')
q.text(3,2418,2418,color='r')
Text(3, 2418, '2418')
q.text(4,2371,2371,color='r')
Text(4, 2371, '2371')
plt.show()
```

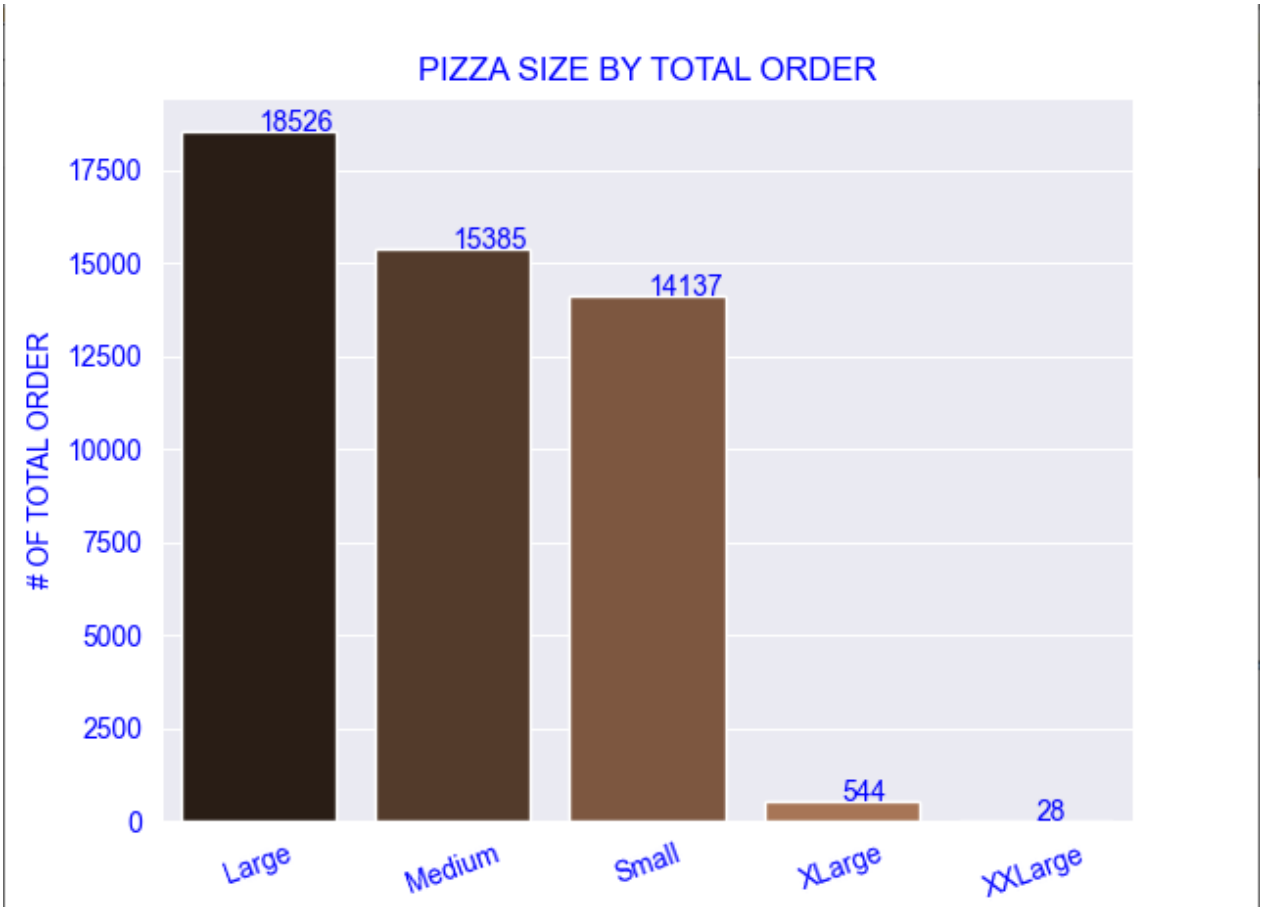
Which of the pizza size generated more sales?

```
size_sales
      size  Total_sales
0    Large    375318.70
1   Medium    249382.25
2    Small    178076.50
3   XLarge     14076.00
4  XXLarge      1006.60
sns.set_palette('copper')
s=sns.barplot(x='size',y='Total_sales',data=size_sales)
s.set_title('PIZZA SIZE BY TOTAL SALES',color='b')
Text(0.5, 1.0, 'PIZZA SIZE BY TOTAL SALES')
s.set_ylabel('# OF TOTAL SALES',color='b')
Text(0, 0.5, '# OF TOTAL SALES')
s.tick_params('x',colors='b')
s.tick_params('y',colors='b')
s.set_xticklabels(size_sales['size'],rotation=20)
[Text(0, 0, 'Large'), Text(1, 0, 'Medium'), Text(2, 0, 'Small'), Text(3, 0, 'XLarge'), Text(4, 0, 'XXLarge')]
s.text(0,375318.70,'$375K',color='b')
Text(0, 375318.7, '$375K')
s.text(1,249382.25,'$249K',color='b')
Text(1, 249382.25, '$249K')
s.text(2,178076.50,'$178K',color='b')
Text(2, 178076.5, '$178K')
s.text(3,14076.00,'$14K',color='b')
Text(3, 14076.0, '$14K')
s.text(4,1006.60,'$10K',color='b')
Text(4, 1006.6, '$10K')
plt.show()
```



Which of the pizza size was the most ordered?

```
size_order=pizza_orders['size'].value_counts().sort_values(ascending=False).reset_index()
size_order
     index   size
0    Large  18526
1   Medium  15385
2    Small  14137
3   XLarge    544
4  XXLarge     28
o=sns.barplot(x='index',y='size',data=size_order)
o.set_title('PIZZA SIZE BY TOTAL ORDER',color='b')
Text(0.5, 1.0, 'PIZZA SIZE BY TOTAL ORDER')
o.set_ylabel('# OF TOTAL ORDER',color='b')
Text(0, 0.5, '# OF TOTAL ORDER')
o.tick_params('x',colors='b')
o.tick_params('y',colors='b')
o.set_xticklabels(size_order['index'],rotation=20)
[Text(0, 0, 'Large'), Text(1, 0, 'Medium'), Text(2, 0, 'Small'), Text(3, 0, 'XLarge'), Text(4, 0, 'XXLarge')]
o.text(0,18526,18526,color='b')
Text(0, 18526, '18526')
o.text(1,15385,15385,color='b')
Text(1, 15385, '15385')
o.text(2,14137,14137,color='b')
Text(2, 14137, '14137')
o.text(3,544,544,color='b')
Text(3, 544, '544')
o.text(4,28,28,color='b')
Text(4, 28, '28')
plt.show()
```

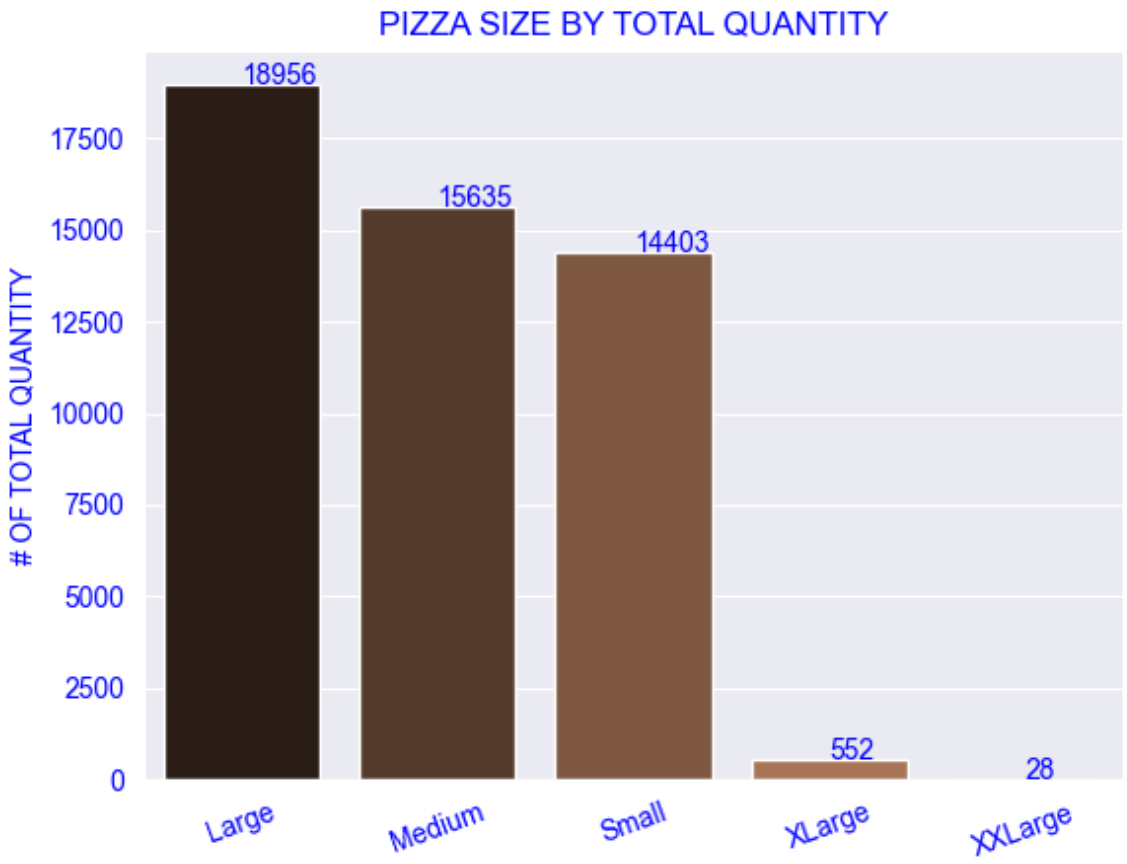


Pizza size by quantity

```
size_quantity=pizza_orders.groupby('size')['quantity'].sum().sort_values(ascending=False).reset_index()
size_quantity
```

	size	quantity
0	Large	18956
1	Medium	15635
2	Small	14403
3	XLarge	552
4	XXLarge	28

```
q=sns.barplot(x='size',y='quantity',data=size_quantity)
q.set_title('PIZZA SIZE BY TOTAL QUANTITY',color='b')
Text(0.5, 1.0, 'PIZZA SIZE BY TOTAL QUANTITY')
q.set_ylabel('# OF TOTAL QUANTITY',color='b')
Text(0, 0.5, '# OF TOTAL QUANTITY')
q.tick_params('x',colors='b')
q.tick_params('y',colors='b')
q.set_xticklabels(size_quantity['size'],rotation=20)
[Text(0, 0, 'Large'), Text(1, 0, 'Medium'), Text(2, 0, 'Small'), Text(3, 0, 'XLarge'), Text(4, 0, 'XXLarge')]
q.text(0,18956,18956,color='b')
Text(0, 18956, '18956')
q.text(1,15635,15635,color='b')
Text(1, 15635, '15635')
q.text(2,14403,14403,color='b')
Text(2, 14403, '14403')
q.text(3,552,552,color='b')
Text(3, 552, '552')
q.text(4,28,28,color='b')
Text(4, 28, '28')
plt.show()
```

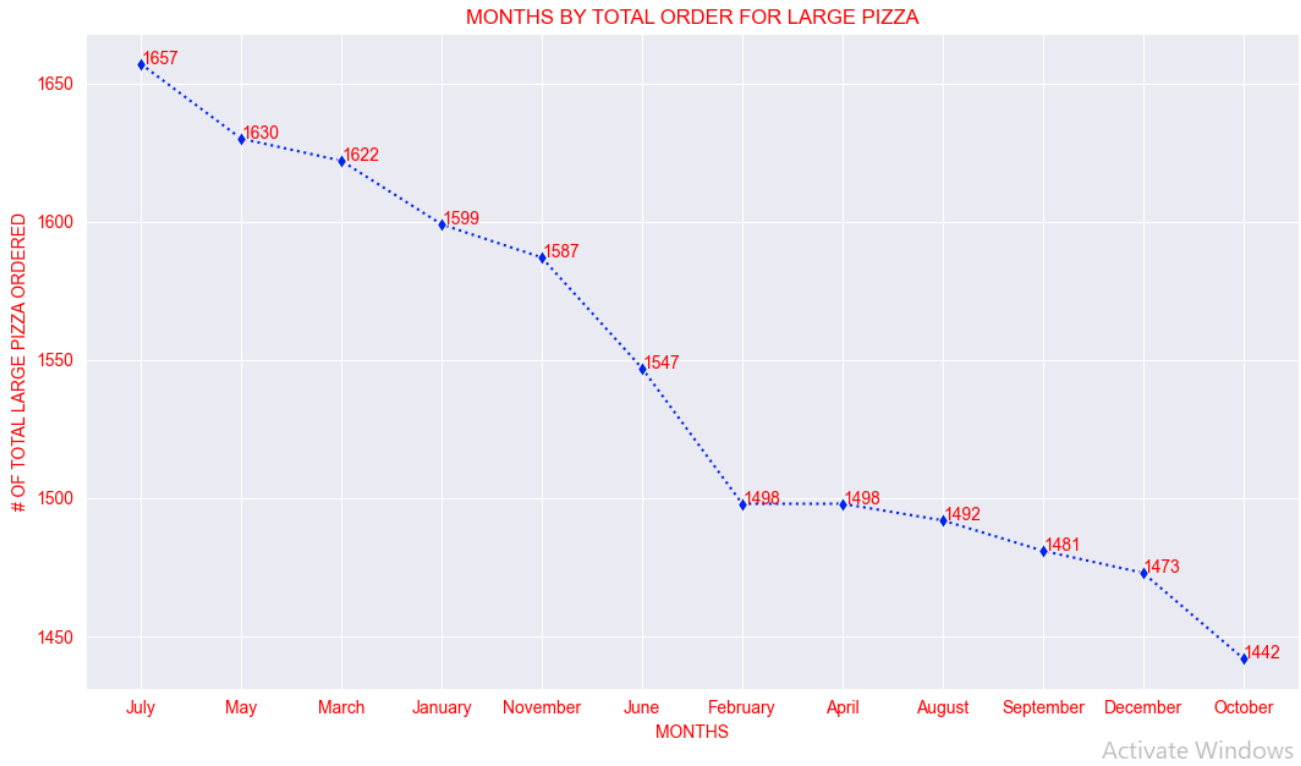


How many large pizzas were ordered in each month?

```
large_pizza=pizza_orders.query('size=="Large"')
large_pizza
       order_details_id  order_id  ...        Day  Total_sales
3                   125        52  ...   Thursday         16.5
4                   175        76  ...     Friday         16.5
6                   202        86  ...     Friday         16.5
7                   234       100  ...     Friday         16.5
9                   256       108  ...     Friday         16.5
...                 ...       ...  ...        ...          ...
48131             48300     21219  ...    Tuesday         17.5
48132             48333     21228  ...    Tuesday         17.5
48133             48408     21252  ...  Wednesday         17.5
48134             48434     21269  ...  Wednesday         17.5
48136             48512     21302  ...   Thursday         17.5

[18526 rows x 15 columns]
month_large=large_pizza['Month'].value_counts().reset_index()
month_large
        index  Month
0        July   1657
1         May   1630
2       March   1622
3     January   1599
4    November   1587
5        June   1547
6    February   1498
7       April   1498
8      August   1492
9   September   1481
10   December   1473
11    October   1442
l=sns.lineplot(x='index',y='Month',data=month_large,linestyle=':',marker='d')
l.set_title('MONTHS BY TOTAL ORDER FOR LARGE PIZZA',color='r')
Text(0.5, 1.0, 'MONTHS BY TOTAL ORDER FOR LARGE PIZZA')
l.set_xlabel('MONTHS',color='r')
Text(0.5, 0, 'MONTHS')
l.set_ylabel('# OF TOTAL LARGE PIZZA ORDERED',color='r')
Text(0, 0.5, '# OF TOTAL LARGE PIZZA ORDERED')
l.tick_params('x',colors='r')
l.tick_params('y',colors='r')
l.text(0,1657,1657,color='r')
Text(0, 1657, '1657')
l.text(1,1630,1630,color='r')
```

```
Text(1, 1630, '1630')
l.text(3,1599,1599,color='r')
Text(3, 1599, '1599')
l.text(2,1622,1622,color='r')
Text(2, 1622, '1622')
l.text(4,1587,1587,color='r')
Text(4, 1587, '1587')
l.text(5,1547,1547,color='r')
Text(5, 1547, '1547')
l.text(6,1498,1498,color='r')
Text(6, 1498, '1498')
l.text(7,1498,1498,color='r')
Text(7, 1498, '1498')
l.text(8,1492,1492,color='r')
Text(8, 1492, '1492')
l.text(9,1481,1481,color='r')
Text(9, 1481, '1481')
l.text(10,1473,1473,color='r')
Text(10, 1473, '1473')
l.text(11,1442,1442,color='r')
Text(11, 1442, '1442')
plt.show()
```

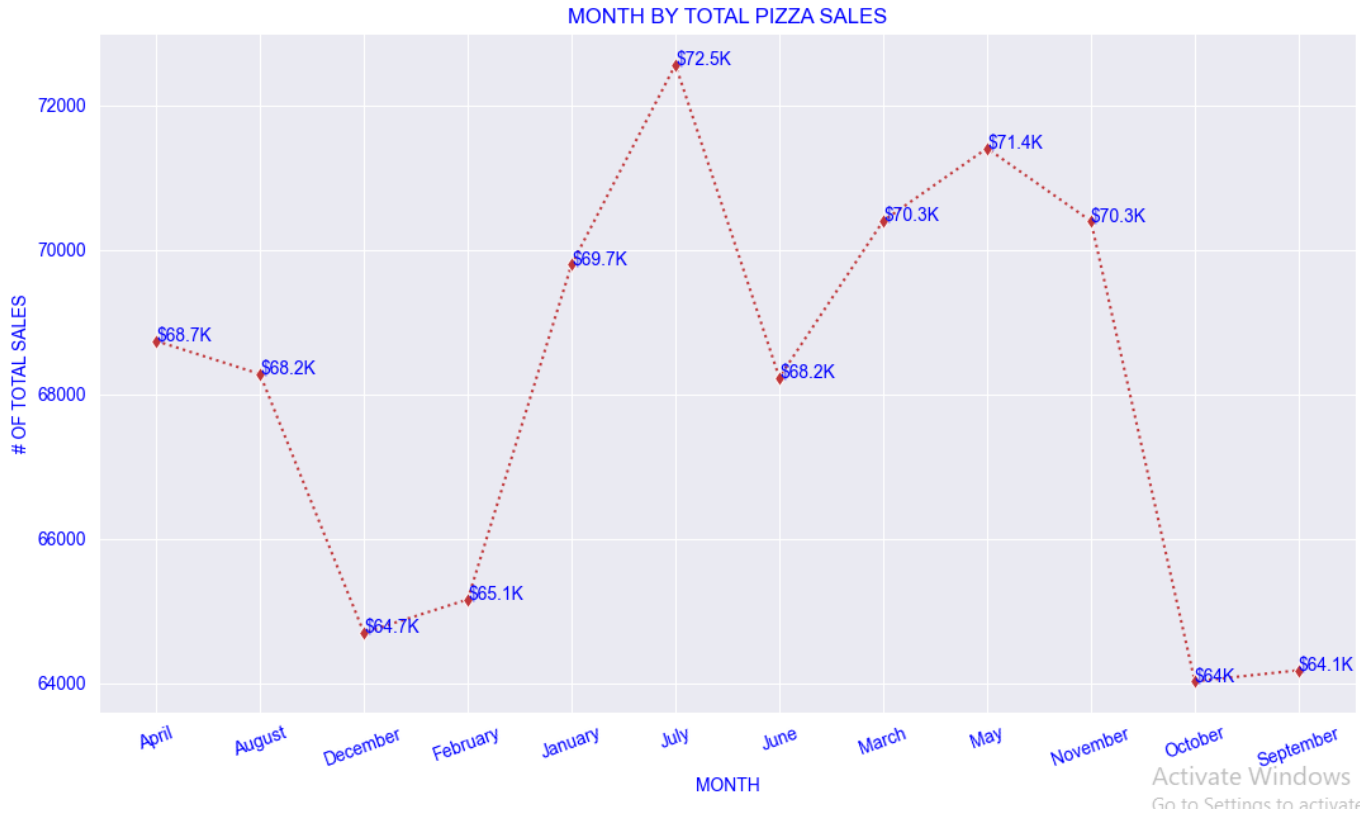


Which month has the highest total sales?

```
month_sales=pizza_orders.groupby('Month')['Total_sales'].sum().reset_index()
month_sales
        Month  Total_sales
0       April     68736.80
1      August     68278.25
2    December     64701.15
3    February     65159.60
4     January     69793.30
5        July     72557.90
6        June     68230.20
7       March     70397.10
8         May     71402.75
9    November     70395.35
10    October     64027.60
11  September     64180.05
s=sns.lineplot(x='Month',y='Total_sales',data=month_sales,linestyle=':',marker='d')
s.set_title('MONTH BY TOTAL PIZZA SALES',color='b')
Text(0.5, 1.0, 'MONTH BY TOTAL PIZZA SALES')
s.set_xlabel('MONTH',color='b')
Text(0.5, 0, 'MONTH')
s.set_ylabel('# OF TOTAL SALES',color='b')
Text(0, 0.5, '# OF TOTAL SALES')
s.tick_params('x',colors='b')
s.tick_params('y',colors='b')
s.set_xticklabels(month_sales['Month'],rotation=20)

Warning (from warnings module):
  File "<pyshell#16>", line 1
UserWarning: FixedFormatter should only be used together with FixedLocator
[Text(0, 0, 'April'), Text(1, 0, 'August'), Text(2, 0, 'December'), Text(3, 0, 'February'), Text(4, 0, 'January'), Text(5, 0, 'July'), Text(6, 0, 'June'
), Text(7, 0, 'March'), Text(8, 0, 'May'), Text(9, 0, 'November'), Text(10, 0, 'October'), Text(11, 0, 'September')]
s.text(0,68736.80,'$68.7K',color='b')
Text(0, 68736.8, '$68.7K')
s.text(1,68278.25,'$68.2K',color='b')
Text(1, 68278.25, '$68.2K')
s.text(2,64701.15,'$64.7K',color='b')
Text(2, 64701.15, '$64.7K')
s.text(3,65159.60,'$65.1K',color='b')
Text(3, 65159.6, '$65.1K')
s.text(4,69793.30,'$69.7K',color='b')
Text(4, 69793.3, '$69.7K')
s.text(5,72557.90,'$72.5K',color='b')
Text(5, 72557.9, '$72.5K')
```

```
s.text(6,68230.20,'$68.2K',color='b')
Text(6, 68230.2, '$68.2K')
s.text(7,70397.10,'$70.3K',color='b')
Text(7, 70397.1, '$70.3K')
s.text(8,71402.75,'$71.4K',color='b')
Text(8, 71402.75, '$71.4K')
s.text(9,70395.35,'$70.3K',color='b')
Text(9, 70395.35, '$70.3K')
s.text(10,64027.60,'$64K',color='b')
Text(10, 64027.6, '$64K')
s.text(11,64180.05,'$64.1K',color='b')
Text(11, 64180.05, '$64.1K')
plt.show()
```

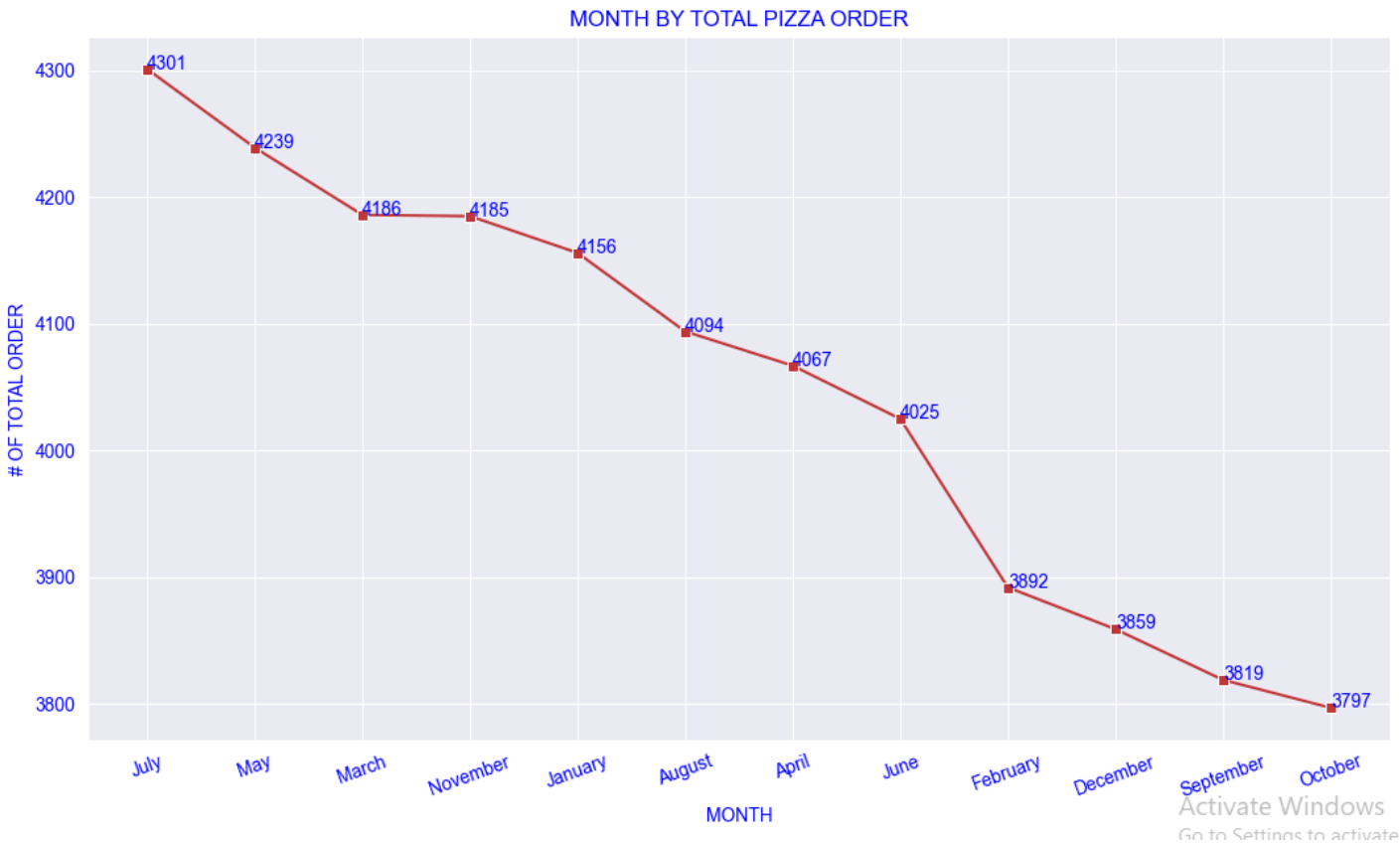
Display the pizza total order by month

```
month_orders=pizza_orders['Month'].value_counts().reset_index()
month_orders
        index  Month
0        July   4301
1         May   4239
2       March   4186
3    November   4185
4     January   4156
5      August   4094
6       April   4067
7        June   4025
8    February   3892
9    December   3859
10  September   3819
11    October   3797
o=sns.lineplot(x='index',y='Month',data=month_orders,linestyle='-',marker='s')
o.set_title('MONTH BY TOTAL PIZZA ORDER',color='b')
Text(0.5, 1.0, 'MONTH BY TOTAL PIZZA ORDER')
o.set_xlabel('MONTH',color='b')
Text(0.5, 0, 'MONTH')
o.set_ylabel('# OF TOTAL ORDER',color='b')
Text(0, 0.5, '# OF TOTAL ORDER')
o.tick_params('x',colors='b')
o.tick_params('y',colors='b')
o.set_xticklabels(month_orders['index'],rotation=20)

Warning (from warnings module):
  File "<pyshell#38>", line 1
UserWarning: FixedFormatter should only be used together with FixedLocator
[Text(0, 0, 'July'), Text(1, 0, 'May'), Text(2, 0, 'March'), Text(3, 0, 'November'), Text(4, 0, 'January'), Text(5, 0, 'August'), Text(6, 0, 'April'), T
ext(7, 0, 'June'), Text(8, 0, 'February'), Text(9, 0, 'December'), Text(10, 0, 'September'), Text(11, 0, 'October')]
o.text(0,4301,4301,color='b')
Text(0, 4301, '4301')
o.text(1,4239,4239,color='b')
Text(1, 4239, '4239')
o.text(2,4186,4186,color='b')
Text(2, 4186, '4186')
o.text(3,4185,4185,color='b')
Text(3, 4185, '4185')
o.text(4,4156,4156,color='b')
Text(4, 4156, '4156')
o.text(5,4094,4094,color='b')
Text(5, 4094, '4094')
```



```
o.text(6,4067,4067,color='b')
Text(6, 4067, '4067')
o.text(7,4025,4025,color='b')
Text(7, 4025, '4025')
o.text(8,3892,3892,color='b')
Text(8, 3892, '3892')
o.text(9,3859,3859,color='b')
Text(9, 3859, '3859')
o.text(10,3819,3819,color='b')
Text(10, 3819, '3819')
o.text(11,3797,3797,color='b')
Text(11, 3797, '3797')
plt.show()
```



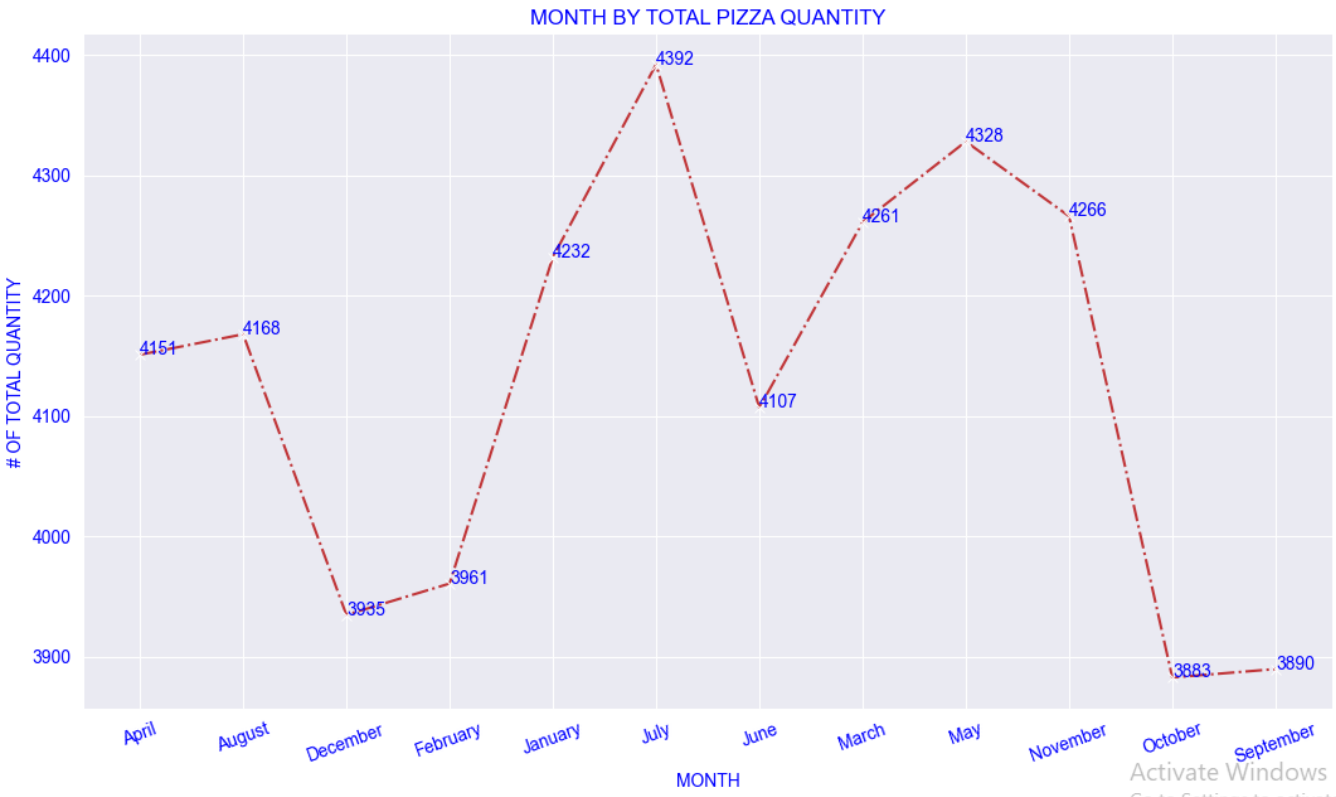
Display the pizza total quantity by month

```
month_quantity=pizza_orders.groupby('Month')['quantity'].sum().reset_index()
month_quantity
        Month  quantity
0       April      4151
1      August      4168
2    December      3935
3    February      3961
4     January      4232
5        July      4392
6        June      4107
7       March      4261
8         May      4328
9    November      4266
10    October      3883
11  September      3890
q=sns.lineplot(x='Month',y='quantity',data=month_quantity,linestyle='-.',marker='x')
q.set_title('MONTH BY TOTAL PIZZA QUANTITY',color='b')
Text(0.5, 1.0, 'MONTH BY TOTAL PIZZA QUANTITY')
q.set_xlabel('MONTH',color='b')
Text(0.5, 0, 'MONTH')
q.set_ylabel('# OF TOTAL QUANTITY',color='b')
Text(0, 0.5, '# OF TOTAL QUANTITY')
q.tick_params('x',colors='b')
q.tick_params('y',colors='b')
q.set_xticklabels(month_quantity['Month'],rotation=20)

Warning (from warnings module):
  File "<pyshell#15>", line 1
UserWarning: FixedFormatter should only be used together with FixedLocator
[Text(0, 0, 'April'), Text(1, 0, 'August'), Text(2, 0, 'December'), Text(3, 0, 'February'), Text(4, 0, 'January'), Text(5, 0, 'July'), Text(6, 0, 'June'
), Text(7, 0, 'March'), Text(8, 0, 'May'), Text(9, 0, 'November'), Text(10, 0, 'October'), Text(11, 0, 'September')]
q.text(0,4151,4151,color='b')
Text(0, 4151, '4151')
q.text(1,4168,4168,color='b')
Text(1, 4168, '4168')
q.text(2,3935,3935,color='b')
Text(2, 3935, '3935')
q.text(3,3961,3961,color='b')
Text(3, 3961, '3961')
q.text(4,4232,4232,color='b')
Text(4, 4232, '4232')
q.text(5,4392,4392,color='b')
Text(5, 4392, '4392')
```

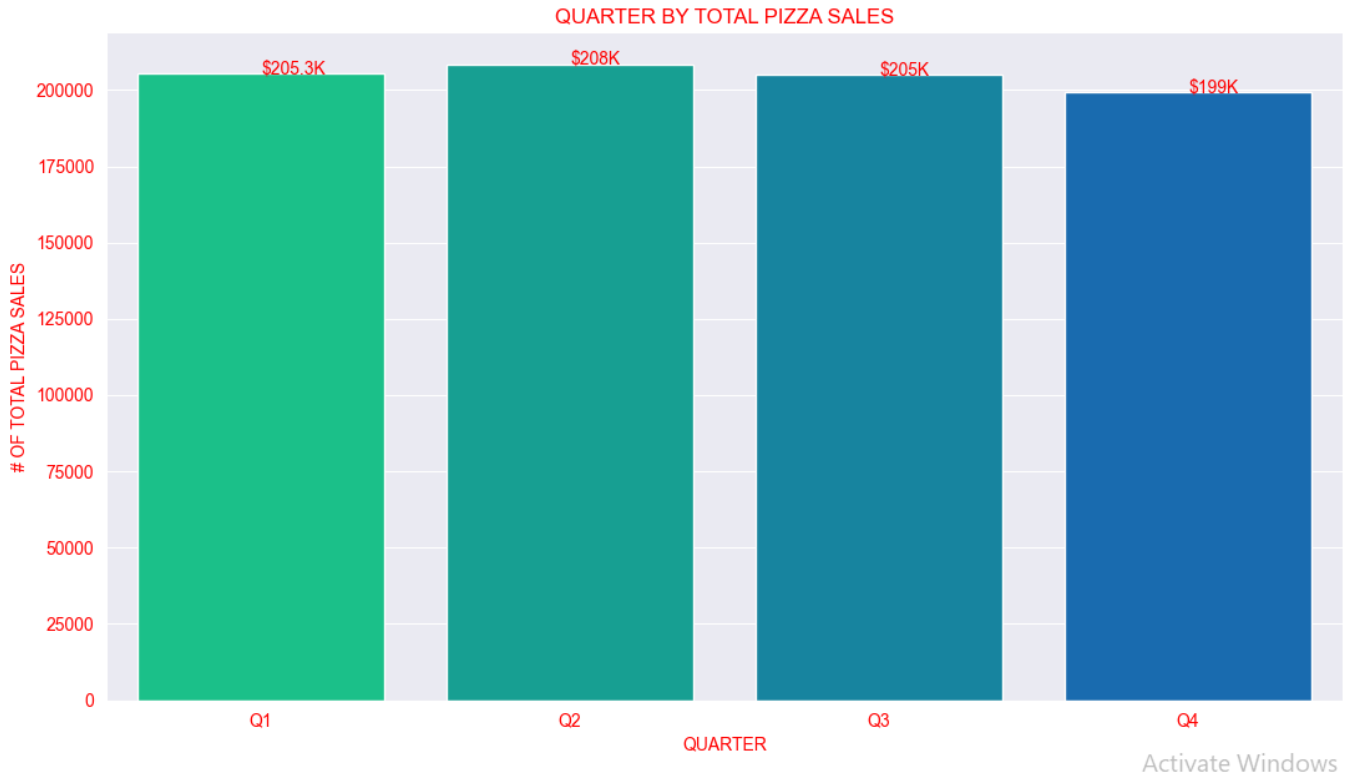


```
q.text(6,4107,4107,color='b')
Text(6, 4107, '4107')
q.text(7,4261,4261,color='b')
Text(7, 4261, '4261')
q.text(8,4328,4328,color='b')
Text(8, 4328, '4328')
q.text(9,4266,4266,color='b')
Text(9, 4266, '4266')
q.text(10,3883,3883,color='b')
Text(10, 3883, '3883')
q.text(11,3890,3890,color='b')
Text(11, 3890, '3890')
plt.show()
```



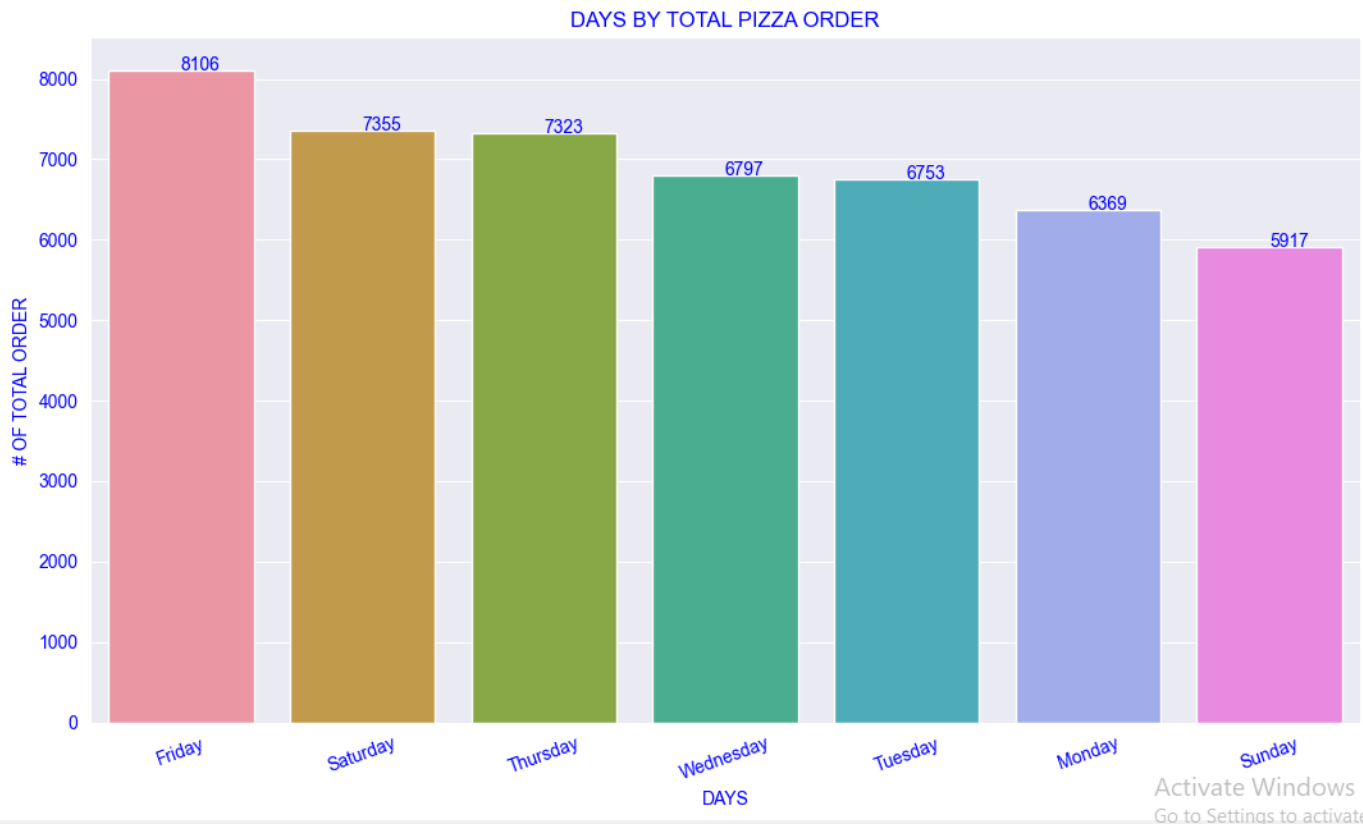
What are the total sales per quarter?

```
quarter=pizza_orders.groupby('Quarter')['Total_sales'].sum().reset_index()
quarter
  Quarter  Total_sales
0      Q1    205350.00
1      Q2    208369.75
2      Q3    205016.20
3      Q4    199124.10
q=sns.barplot(x='Quarter',y='Total_sales',data=quarter)
q.set_title('QUARTER BY TOTAL PIZZA SALES',color='r')
Text(0.5, 1.0, 'QUARTER BY TOTAL PIZZA SALES')
q.set_xlabel('QUARTER',color='r')
Text(0.5, 0, 'QUARTER')
q.set_ylabel('# OF TOTAL PIZZA SALES',color='r')
Text(0, 0.5, '# OF TOTAL PIZZA SALES')
q.tick_params('x',colors='r')
q.tick_params('y',colors='r')
q.text(0,205350.00,'$205.3K',color='r')
Text(0, 205350.0, '$205.3K')
q.text(1,208369.75,'$208K',color='r')
Text(1, 208369.75, '$208K')
q.text(2,205016.20,'$205K',color='r')
Text(2, 205016.2, '$205K')
q.text(3,199124.10,'$199K',color='r')
Text(3, 199124.1, '$199K')
plt.show()
```



Which day is the busiest day of the year?

```
day_order=pizza_orders['Day'].value_counts().sort_values(ascending=False).reset_index()
day_order
       index   Day
0     Friday  8106
1   Saturday  7355
2   Thursday  7323
3  Wednesday  6797
4    Tuesday  6753
5     Monday  6369
6     Sunday  5917
sns.set_palette('OrRd')
sns.set_style('darkgrid')
o=sns.barplot(x='index',y='Day',data=day_order)
o.set_title('DAYS BY TOTAL PIZZA ORDER',color='b')
Text(0.5, 1.0, 'DAYS BY TOTAL PIZZA ORDER')
o.set_xlabel('DAYS',color='b')
Text(0.5, 0, 'DAYS')
o.set_ylabel('# OF TOTAL ORDER',color='b')
Text(0, 0.5, '# OF TOTAL ORDER')
o.tick_params('x',colors='b')
o.tick_params('y',colors='b')
o.set_xticklabels(day_order['index'],rotation=20)
[Text(0, 0, 'Friday'), Text(1, 0, 'Saturday'), Text(2, 0, 'Thursday'), Text(3, 0, 'Wednesday'), Text(4, 0, 'Tuesday'), Text(5, 0, 'Monday'), Text(6, 0,
'Sunday')]
o.text(0,8106,8106,color='b')
Text(0, 8106, '8106')
o.text(1,7355,7355,color='b')
Text(1, 7355, '7355')
o.text(2,7323,7323,color='b')
Text(2, 7323, '7323')
o.text(3,6797,6797,color='b')
Text(3, 6797, '6797')
o.text(4,6753,6753,color='b')
Text(4, 6753, '6753')
o.text(5,6369,6369,color='b')
Text(5, 6369, '6369')
o.text(6,5917,5917,color='b')
Text(6, 5917, '5917')
plt.show()
```

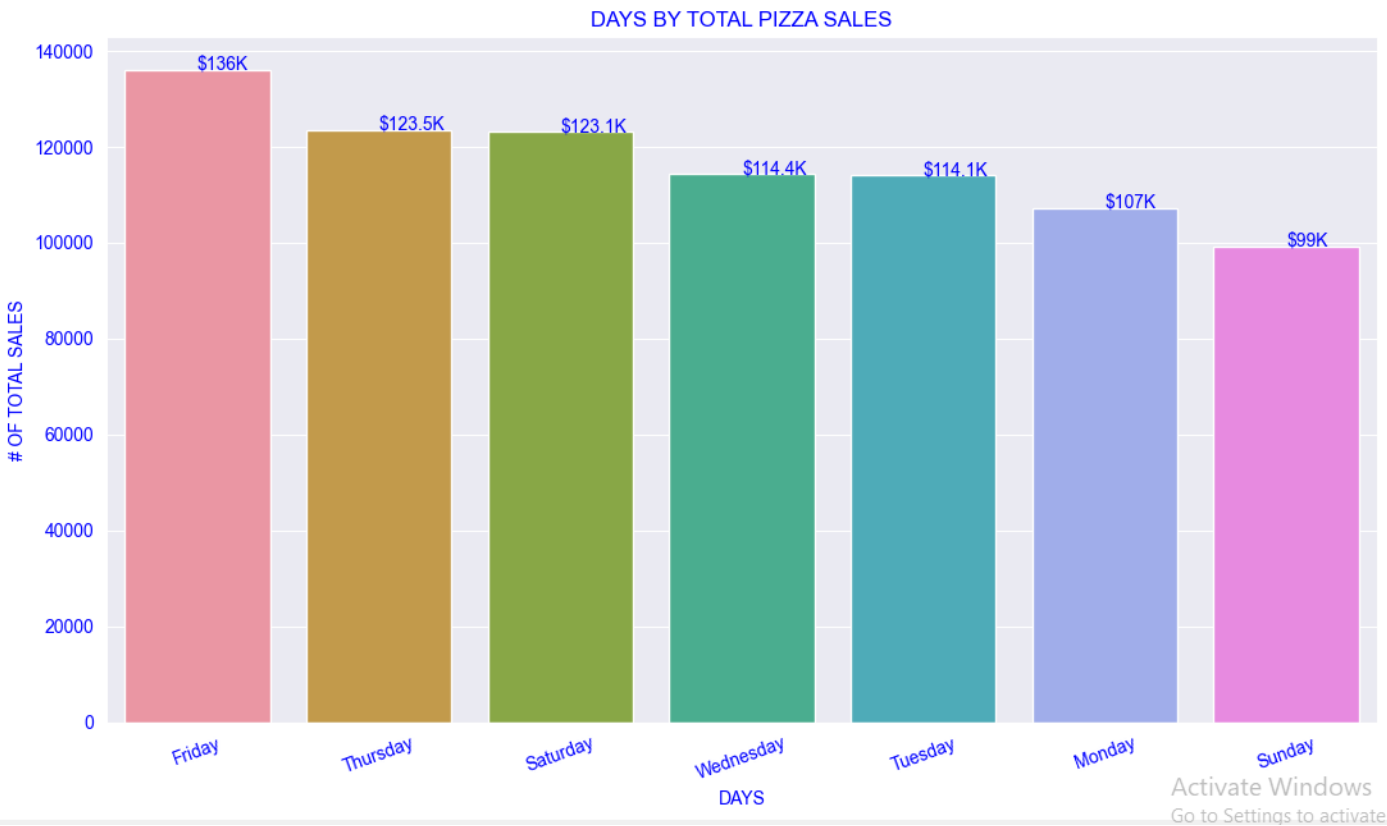


What is the total sales made in days of the year?

```
day_sales=pizza_orders.groupby('Day')['Total_sales'].sum().sort_values(ascending=False).reset_index()
day_sales
```

	Day	Total_sales
0	Friday	136073.90
1	Thursday	123528.50
2	Saturday	123182.40
3	Wednesday	114408.40
4	Tuesday	114133.80
5	Monday	107329.55
6	Sunday	99203.50

```
s=sns.barplot(x='Day',y='Total_sales',data=day_sales)
s.set_title('DAYS BY TOTAL PIZZA SALES',color='b')
Text(0.5, 1.0, 'DAYS BY TOTAL PIZZA SALES')
s.set_xlabel('DAYS',color='b')
Text(0.5, 0, 'DAYS')
s.set_ylabel('# OF TOTAL SALES',color='b')
Text(0, 0.5, '# OF TOTAL SALES')
s.tick_params('x',colors='b')
s.tick_params('y',colors='b')
s.set_xticklabels(day_sales['Day'],rotation=20)
[Text(0, 0, 'Friday'), Text(1, 0, 'Thursday'), Text(2, 0, 'Saturday'), Text(3, 0, 'Wednesday'), Text(4, 0, 'Tuesday'), Text(5, 0, 'Monday'), Text(6, 0,
'Sunday')]
s.text(0,136073.90,'$136K',color='b')
Text(0, 136073.9, '$136K')
s.text(1,123528.50,'$123.5K',color='b')
Text(1, 123528.5, '$123.5K')
s.text(2,123182,'$123.1K',color='b')
Text(2, 123182, '$123.1K')
s.text(3,114408.40,'$114.4K',color='b')
Text(3, 114408.4, '$114.4K')
s.text(4,114133.80,'$114.1K',color='b')
Text(4, 114133.8, '$114.1K')
s.text(5,107329.55,'$107K',color='b')
Text(5, 107329.55, '$107K')
s.text(6,99203.50,'$99K',color='b')
Text(6, 99203.5, '$99K')
plt.show()
```

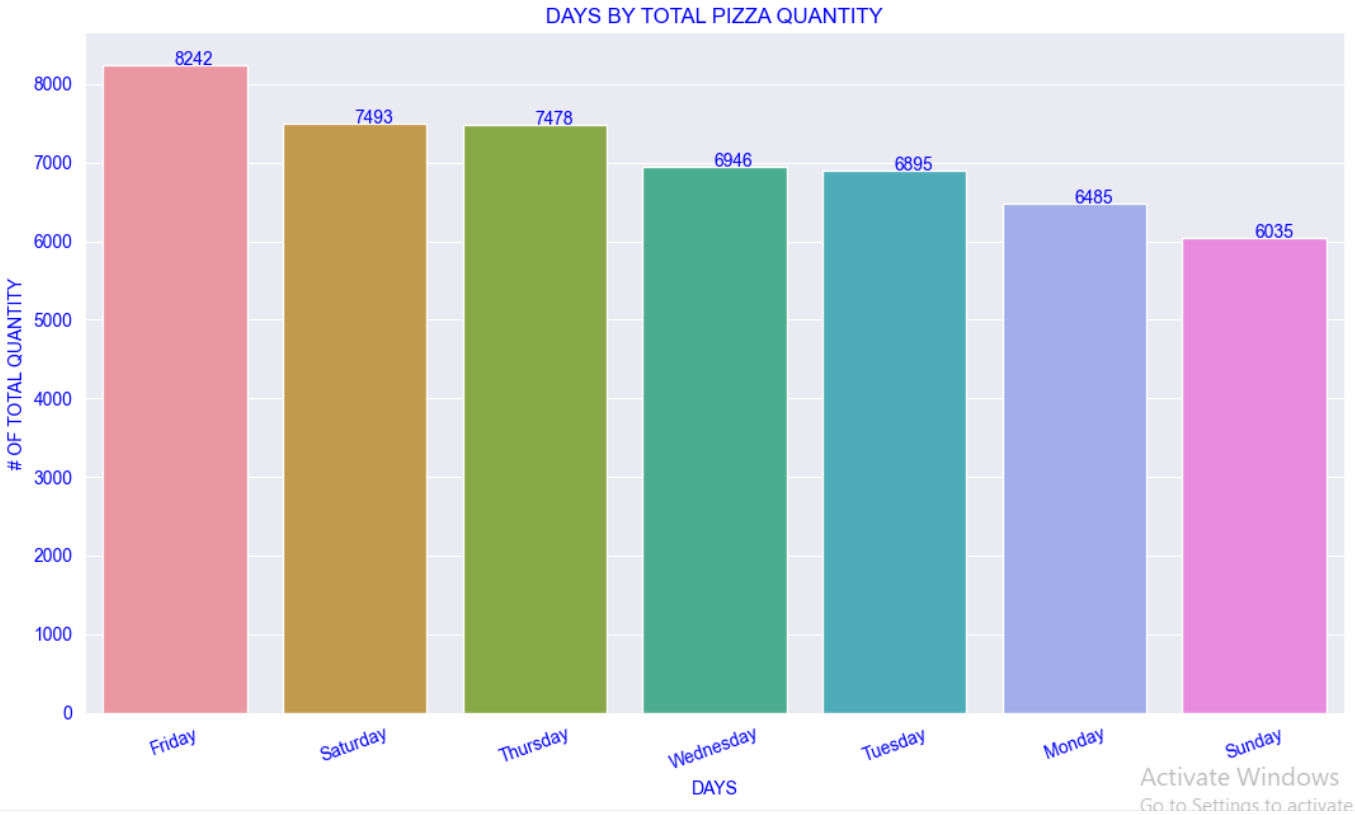


What is the total quantity purchased in days of the year?

```
day_quantity=pizza_orders.groupby('Day')['quantity'].sum().sort_values(ascending=False).reset_index()
day_quantity
```

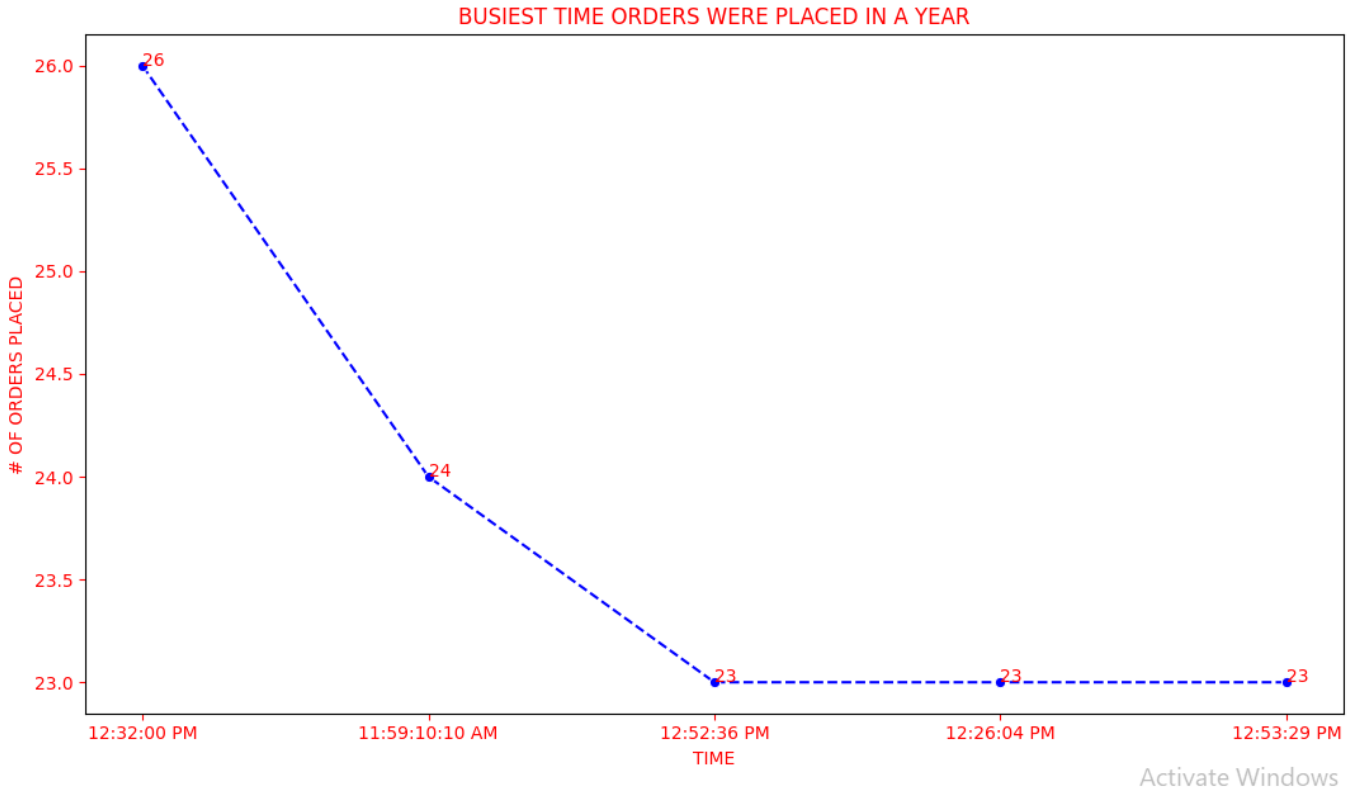
	Day	quantity
0	Friday	8242
1	Saturday	7493
2	Thursday	7478
3	Wednesday	6946
4	Tuesday	6895
5	Monday	6485
6	Sunday	6035

```
q=sns.barplot(x='Day',y='quantity',data=day_quantity)
q.set_title('DAYS BY TOTAL PIZZA QUANTITY',color='b')
Text(0.5, 1.0, 'DAYS BY TOTAL PIZZA QUANTITY')
q.set_xlabel('DAYS',color='b')
Text(0.5, 0, 'DAYS')
q.set_ylabel('# OF TOTAL QUANTITY',color='b')
Text(0, 0.5, '# OF TOTAL QUANTITY')
q.tick_params('x',colors='b')
q.tick_params('y',colors='b')
q.set_xticklabels(day_quantity['Day'],rotation=20)
[Text(0, 0, 'Friday'), Text(1, 0, 'Saturday'), Text(2, 0, 'Thursday'), Text(3, 0, 'Wednesday'), Text(4, 0, 'Tuesday'), Text(5, 0, 'Monday'), Text(6, 0,
'Sunday')]
q.text(0,8242,8242,color='b')
Text(0, 8242, '8242')
q.text(1,7493,7493,color='b')
Text(1, 7493, '7493')
q.text(2,7478,7478,color='b')
Text(2, 7478, '7478')
q.text(3,6946,6946,color='b')
Text(3, 6946, '6946')
q.text(4,6895,6895,color='b')
Text(4, 6895, '6895')
q.text(5,6485,6485,color='b')
Text(5, 6485, '6485')
q.text(6,6035,6035,color='b')
Text(6, 6035, '6035')
plt.show()
```



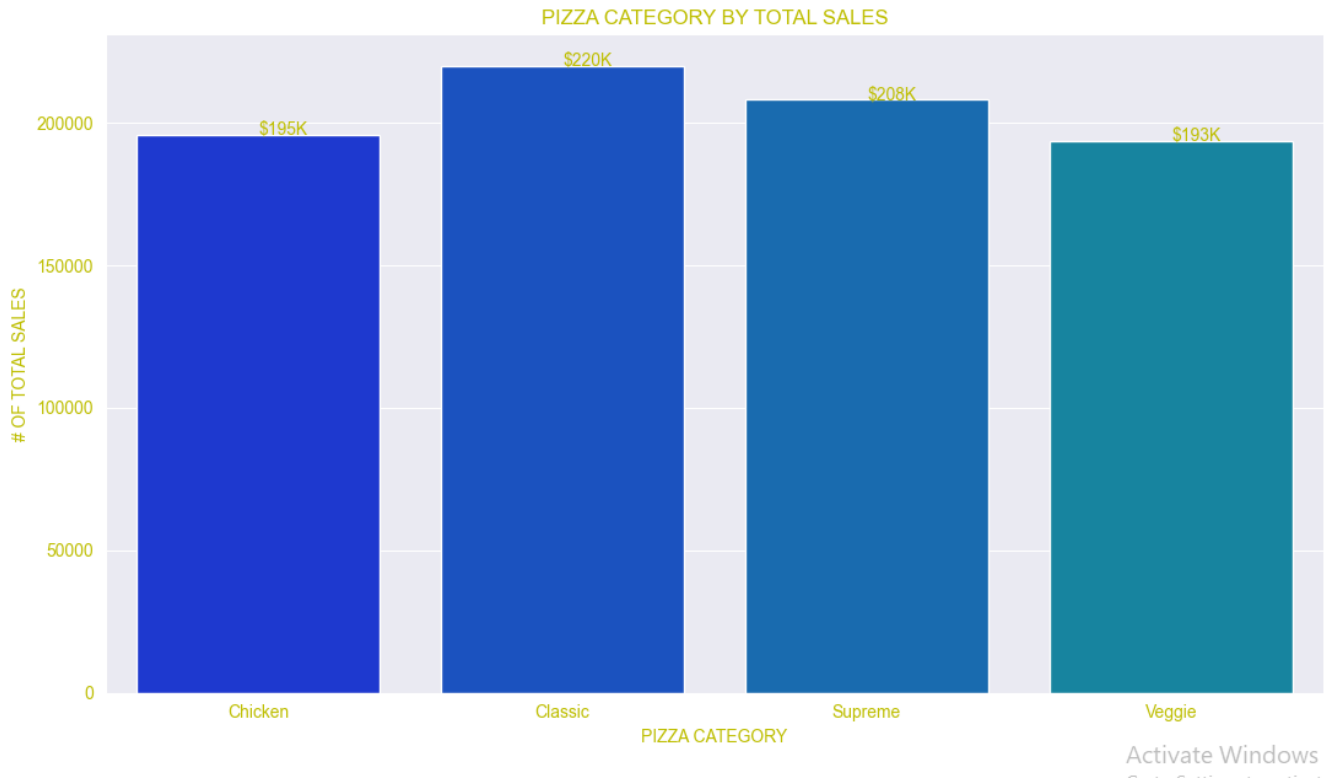
What are the top 5 busiest time in which orders were placed in the year?

```
time_orders=pizza_orders['time'].value_counts().sort_values(ascending=False).reset_index().iloc[:5]
time_orders
      index  time
0  12:32:00    26
1  11:59:10    24
2  12:52:36    23
3  12:26:04    23
4  12:53:29    23
t=sns.lineplot(x='index',y='time',data=time_orders,linestyle='--',marker='o',color='b')
t.set_title('BUSIEST TIME ORDERS WERE PLACED IN A YEAR',color='r')
Text(0.5, 1.0, 'BUSIEST TIME ORDERS WERE PLACED IN A YEAR')
t.set_xlabel('TIME',color='r')
Text(0.5, 0, 'TIME')
t.set_ylabel('# OF ORDERS PLACED',color='r')
Text(0, 0.5, '# OF ORDERS PLACED')
t.tick_params('x',colors='r')
t.tick_params('y',colors='r')
t.set_xticks(['12:32:00','11:59:10','12:52:36','12:26:04','12:53:29'],['12:32:00 PM','11:59:10:10 AM','12:52:36 PM','12:26:04 PM','12:53:29 PM'])
[<matplotlib.axis.XTick object at 0x0000014672060820>, <matplotlib.axis.XTick object at 0x00000146720607F0>, <matplotlib.axis.XTick object at 0x00000146
7208D4E0>, <matplotlib.axis.XTick object at 0x000001467208DFC0>, <matplotlib.axis.XTick object at 0x000001467208E650>]
t.text(0,26,26,color='r')
Text(0, 26, '26')
t.text(1,24,24,color='r')
Text(1, 24, '24')
t.text(2,23,23,color='r')
Text(2, 23, '23')
t.text(3,23,23,color='r')
Text(3, 23, '23')
t.text(4,23,23,color='r')
Text(4, 23, '23')
plt.show()
```



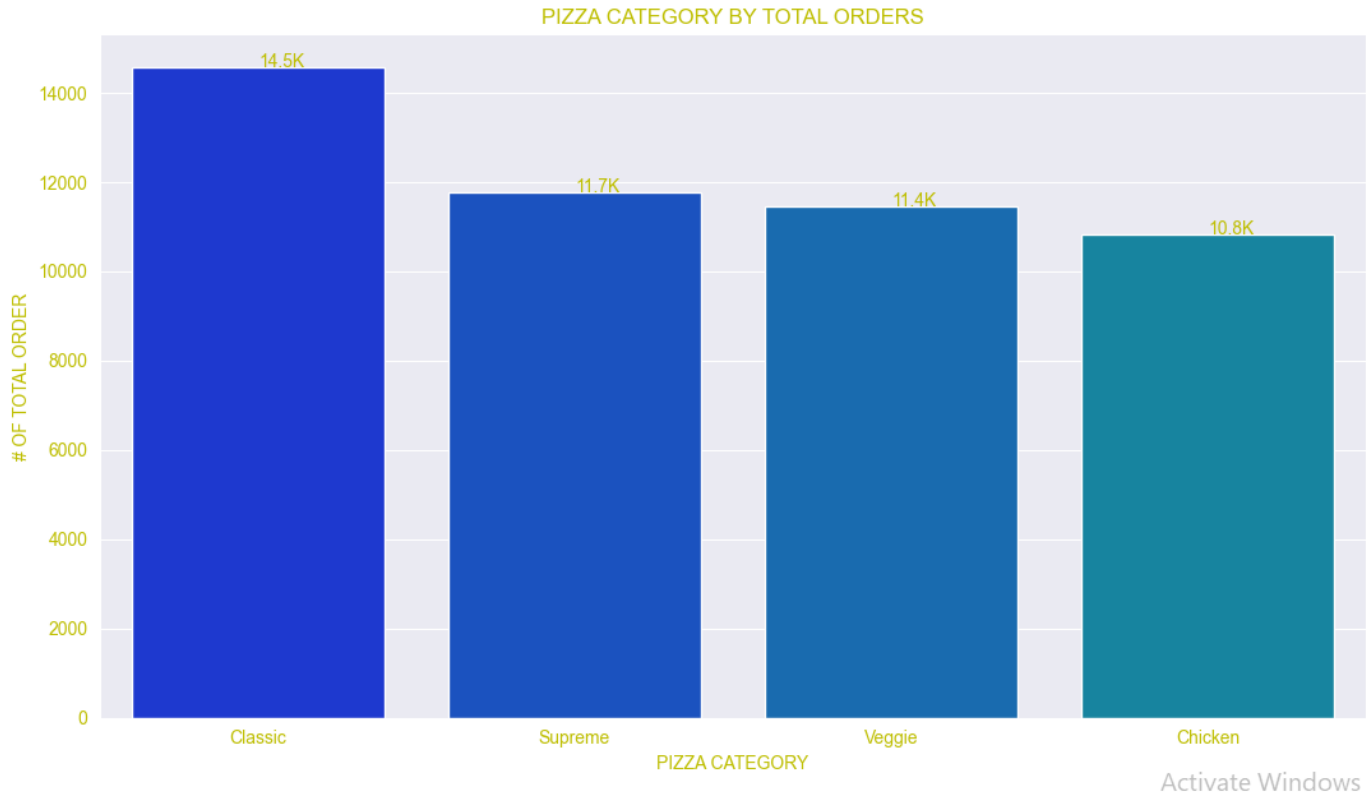
How much was made from each pizza category?

```
category_sales=pizza_orders.groupby('category')['Total_sales'].sum().reset_index()
category_sales
  category  Total_sales
0  Chicken    195919.50
1  Classic    220053.10
2  Supreme    208197.00
3   Veggie    193690.45
sns.set_palette('winter')
sns.set_style('darkgrid')
c=sns.barplot(x='category',y='Total_sales',data=category_sales)
c.set_title('PIZZA CATEGORY BY TOTAL SALES',color='y')
Text(0.5, 1.0, 'PIZZA CATEGORY BY TOTAL SALES')
c.set_xlabel('PIZZA CATEGORY',color='y')
Text(0.5, 0, 'PIZZA CATEGORY')
c.set_ylabel('# OF TOTAL SALES',color='y')
Text(0, 0.5, '# OF TOTAL SALES')
c.tick_params('x',colors='y')
c.tick_params('y',colors='y')
c.text(0,195919.50,'$195K',color='y')
Text(0, 195919.5, '$195K')
c.text(1,220053.10,'$220K',color='y')
Text(1, 220053.1, '$220K')
c.text(2,208197.00,'$208K',color='y')
Text(2, 208197.0, '$208K')
c.text(3,193690.45,'$193K',color='y')
Text(3, 193690.45, '$193K')
plt.show()
```



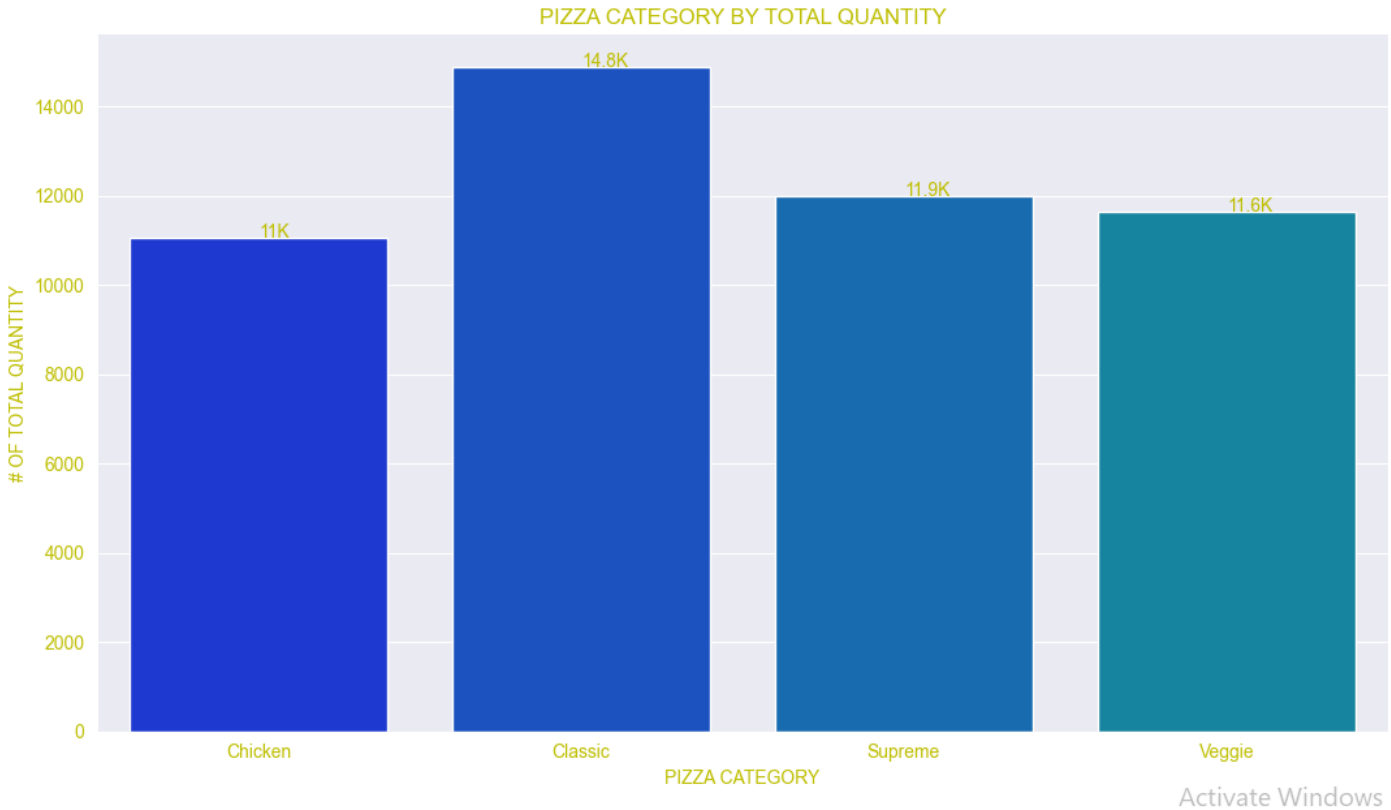
How many pizza orders were placed for each pizza category?

```
category_orders=pizza_orders['category'].value_counts().reset_index()
category_orders
     index  category
0  Classic     14579
1  Supreme     11777
2   Veggie     11449
3  Chicken     10815
o=sns.barplot(x='index',y='category',data=category_orders)
o.set_title('PIZZA CATEGORY BY TOTAL ORDERS',color='y')
Text(0.5, 1.0, 'PIZZA CATEGORY BY TOTAL ORDERS')
o.set_xlabel('PIZZA CATEGORY',color='y')
Text(0.5, 0, 'PIZZA CATEGORY')
o.set_ylabel('# OF TOTAL ORDER',color='y')
Text(0, 0.5, '# OF TOTAL ORDER')
o.tick_params('x',colors='y')
o.tick_params('y',colors='y')
o.text(0,14579,'14.5K',color='y')
Text(0, 14579, '14.5K')
o.text(1,11777,'11.7K',color='y')
Text(1, 11777, '11.7K')
o.text(2,11449,'11.4K',color='y')
Text(2, 11449, '11.4K')
o.text(3,10815,'10.8K',color='y')
Text(3, 10815, '10.8K')
plt.show()
```



How many pizza quantities were ordered for each pizza category?

```
category_quantity=pizza_orders.groupby('category')['quantity'].sum().reset_index()
category_quantity
  category  quantity
0  Chicken     11050
1  Classic     14888
2  Supreme     11987
3   Veggie     11649
q=sns.barplot(x='category',y='quantity',data=category_quantity)
q.set_title('PIZZA CATEGORY BY TOTAL QUANTITY',color='y')
Text(0.5, 1.0, 'PIZZA CATEGORY BY TOTAL QUANTITY')
q.set_xlabel('PIZZA CATEGORY',color='y')
Text(0.5, 0, 'PIZZA CATEGORY')
q.set_ylabel('# OF TOTAL QUANTITY',color='y')
Text(0, 0.5, '# OF TOTAL QUANTITY')
q.tick_params('x',colors='y')
q.tick_params('y',colors='y')
q.text(0,11050,'11K',color='y')
Text(0, 11050, '11K')
q.text(1,14888,'14.8K',color='y')
Text(1, 14888, '14.8K')
q.text(2,11987,'11.9K',color='y')
Text(2, 11987, '11.9K')
q.text(3,11649,'11.6K',color='y')
Text(3, 11649, '11.6K')
plt.show()
```



CONCLUSION

Plato's pizza made over **$817k** from **32** pizza type in year **2015**, **48.62K** orders and **49,574k** quantities of pizzas. Over **$43k** was made from The Thai Chicken Pizza which was their highest sales from the pizza type, **2416** total order for The Classic Deluxe pizza and a total of **2453** Classic Deluxe pizza were ordered.

July was the month with highest sales of about **$72.5K**, while October has the lowest with **$64K**. More sales were generated in Quarter 2(April – June) of the year with **$208K**. Large, Medium and Small pizza are the most ordered pizza type and the production for the size should be increased to avoid unavailability of the pizza while XXlarge pizza quantity should be reduced because it has a low turnout ordered rate. Low turnout rate **might** be due to the high price rate of the XXlarge pizza.

Classic pizza was the most ordered pizza by category with over **14.5K** orders. Thursdays to Saturdays are the busiest days of the year which pizza were ordered while **26** pizzas were ordered around **12:32 PM** which made it the highest ordered time, the organization website and attendants should be available and on standby during those days and time to hasten customer requests.