Code & Output

EX 6.3

Code:

```
# Julia Cuellar
# DSC 550
# Final project

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns


# Display pizza place data
def read_file():
    pizza = pd.read_csv('pizzaplace.csv')
    pizza.rename(columns={'Unnamed: 0': 'num'}, inplace=True)
    pizza.drop(['num', 'id', 'date'], axis=1, inplace=True)
    print('Pizza data:\n', pizza.head(5))


# Display described, summarized, and length of pizza place data
def des_sum_len():
    pizza = pd.read_csv('pizzaplace.csv')
    pizza.rename(columns={'Unnamed: 0': 'num'}, inplace=True)
    pizza.drop(['num', 'id', 'date'], axis=1, inplace=True)
    print('Described pizza data:\n', pizza.describe())
    print('Summarized pizza data:\n', pizza.describe(include=['O']))
    print('Length of pizza data:\n', len(pizza))


# Display bar chart of pizza name
def showBar_Pname():
    pizza = pd.read_csv('pizzaplace.csv')
    pizza.rename(columns={'Unnamed: 0': 'num'}, inplace=True)
    pizza.drop(['num', 'id', 'date'], axis=1, inplace=True)
    pizza['name'].value_counts().plot(kind='barh').invert_yaxis()
    plt.title('Pizza name')
    plt.show()


# Display bar chart of pizza size
def showBar_Psize():
    pizza = pd.read_csv('pizzaplace.csv')
    pizza.rename(columns={'Unnamed: 0': 'num'}, inplace=True)
    pizza.drop(['num', 'id', 'date'], axis=1, inplace=True)
    pizza['size'].value_counts().plot(kind='barh')
    plt.title('Pizza size')
    plt.show()


# Display pie chart of pizza type
def showPie_Ptype():
    pizza = pd.read_csv('pizzaplace.csv')
```

```
    pizza.rename(columns={'Unnamed: 0': 'num'}, inplace=True)
    pizza.drop(['num', 'id', 'date'], axis=1, inplace=True)
    plt.pie(pizza['type'].value_counts(), autopct=lambda p: f'{p:.2f}%',
labels=['classic', 'supreme', 'veggie',

'chicken'])
    plt.title('Pizza type')
    plt.show()


# Display boxplot of pizza price
def showBoxplot_Pprice():
    pizza = pd.read_csv('pizzaplace.csv')
    pizza.rename(columns={'Unnamed: 0': 'num'}, inplace=True)
    pizza.drop(['num', 'id', 'date'], axis=1, inplace=True)
    sns.boxplot(pizza['price'])
    plt.title('Pizza price')
    plt.show()


if __name__ == "__main__":
    read_file()
    des_sum_len()
    showBar_Pname()
    showBar_Psize()
    showPie_Ptype()
    showBoxplot_Pprice()
```

Output:

Pizza data:

```
       time         name size     type  price
0  11:38:36     hawaiian    M  classic  13.25
1  11:57:40  classic_dlx    M  classic  16.00
2  11:57:40     mexicana    M   veggie  16.00
3  11:57:40     thai_ckn    L  chicken  20.75
4  11:57:40  five_cheese    L   veggie  18.50
```

Described pizza data:

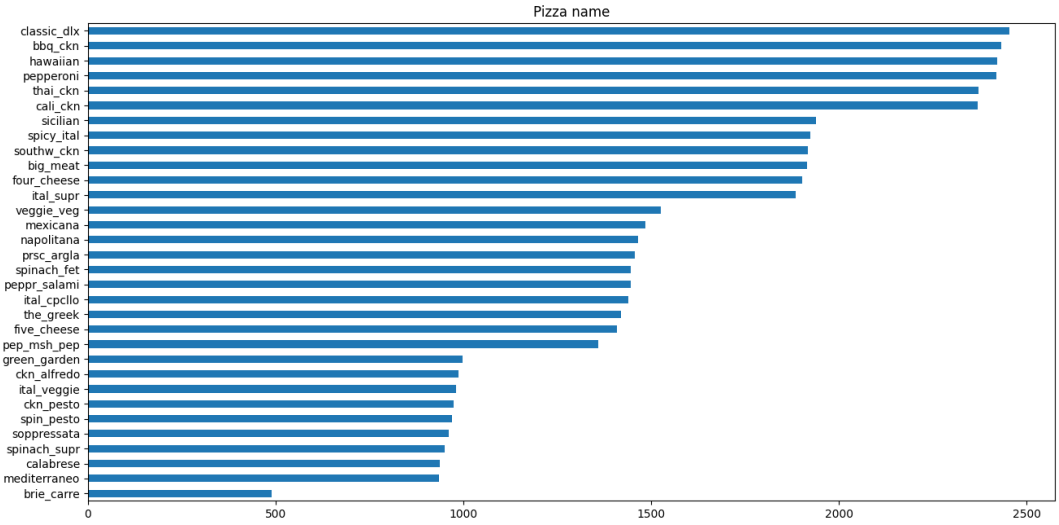
```
              price
count  49574.000000
mean      16.497762
std        3.621954
min        9.750000
25%       12.750000
50%       16.500000
75%       20.250000
max       35.950000
```

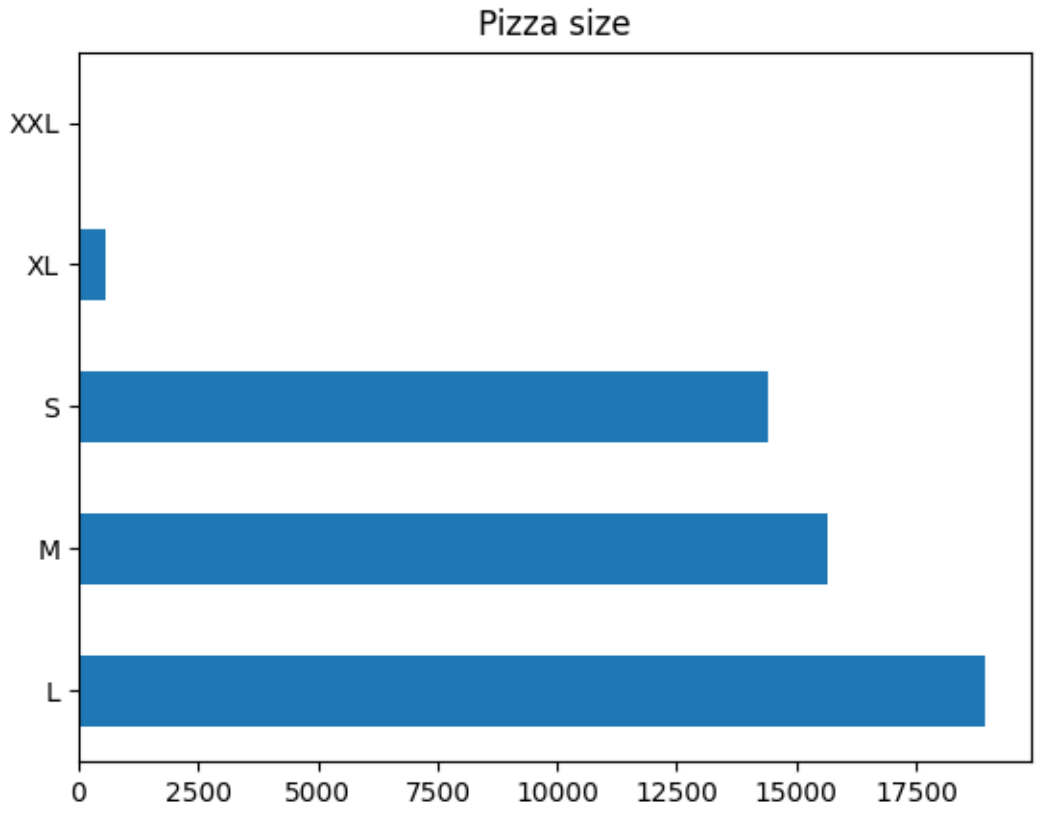
Summarized pizza data:

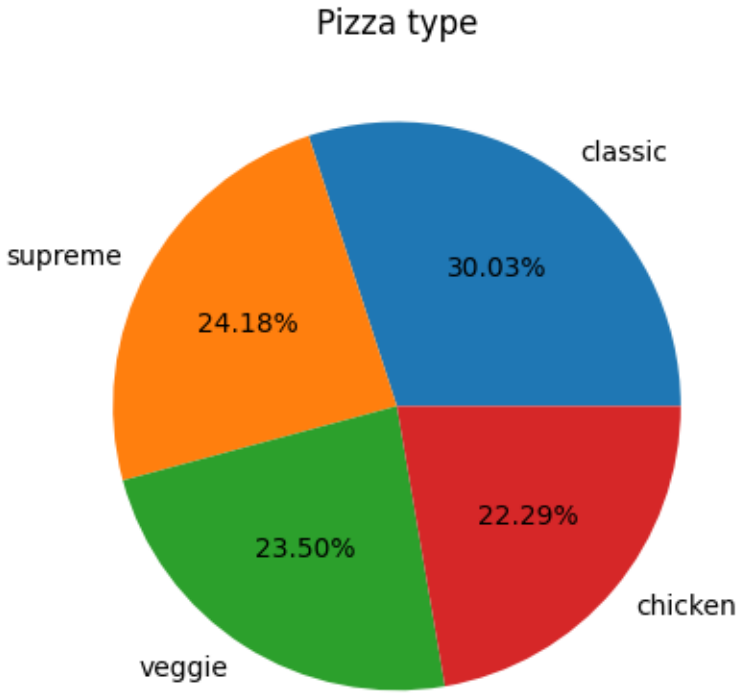
```
           time         name   size     type
count      49574        49574  49574    49574
unique     16382          32      5        4
top     12:25:12  classic_dlx      L  classic
freq          28         2453  18956    14888
```

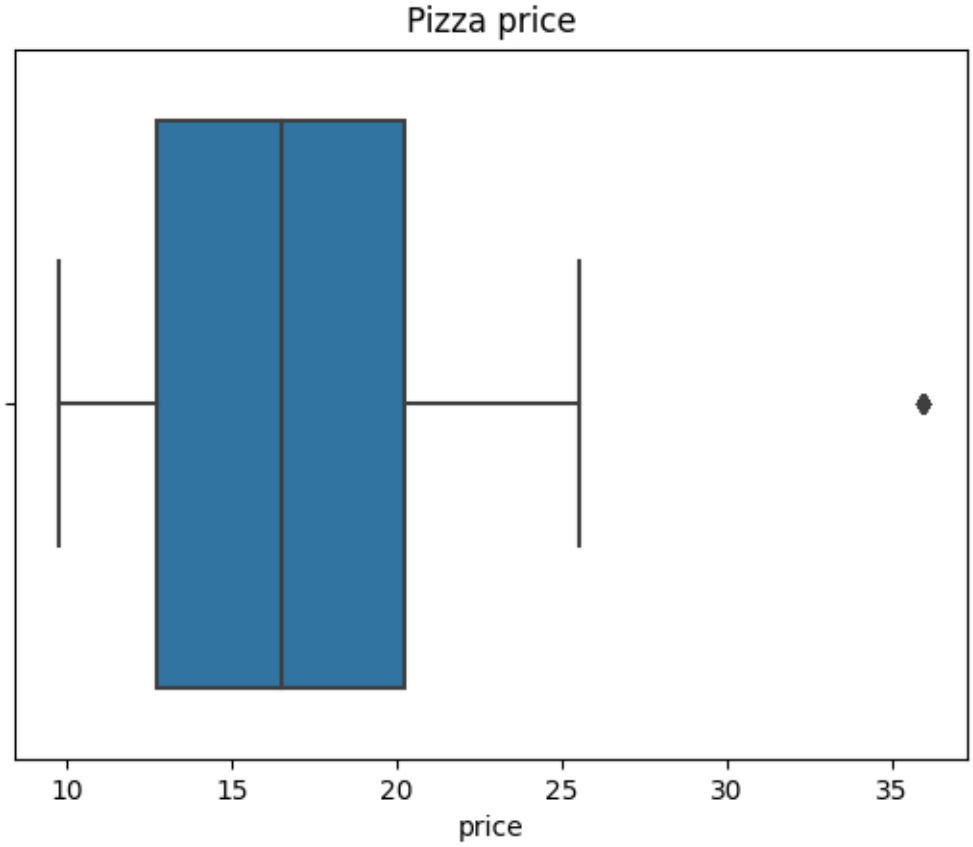
Length of pizza data:

```
 49574
```


Pizza name
classic_dlx
bbq_ckn
hawaiian
pepperoni
thai_ckn
cali_ckn
sicilian
spicy_ital
southw_ckn
big_meat
four_cheese
ital_supr
veggie_veg
mexicana
napolitana
prsc_argla
spinach_fet
peppr_salami
ital_cpcllo
the_greek
five_cheese
pep_msh_pep
green_garden
ckn_alfredo
ital_veggie
ckn_pesto
spin_pesto
soppressata
spinach_supr
calabrese
mediterraneo
brie_carre
0
500
1000
1500
2000
2500


Pizza size
XXL
XL
S
M
L
0
2500
5000
7500
10000
12500
15000
17500


Pizza type
classic
supreme
30.03%
24.18%
22.29%
23.50%
chicken
veggie


Pizza price
10
15
20
25
30
35
price