Machine Learning - Assignment 2

Name: Idavalapati Vijay Taraka Ramarao
ID: 700742485
CRN: 13428

Question1:

Using NumPy, Generated a random vector of size 15 having only Integers in the range 1-20.
Reshape the array to 3 by 5and Printed the array shape.
Replaced the max in each row by 0

Assignment2_Question1.py | Assignment2_Question2.py

```
1   import numpy as np
2
3   # Question1
4   a=np.random.randint(1,20,15)
5
6   print("\n")
7   # 1.a Reshape the array to 3 by 5
8   a=a.reshape(3,5)
9   print(a)
10
11  print("\n")
12  # 1.b Print array shape.
13  print(a.shape)
14
15  print("\n")
16  # 1.c Replace the max in each row by 0
17  a[np.where(a==np.max(a))]=0
18  print(a)
```

Question1 Output:

Question2

2.1 Read the file from the path

```
import pandas as pd
import numpy as np

df=pd.read_csv("C:/Users/Administrator/Desktop/ML_Assignment_2/data.csv")

mean_value=df['Calories'].mean()

df['Calories'].fillna(value=mean_value,inplace=True)

print(df.head(25))
```

2.1 Output

```
C:\Users\Administrator\Documents\GitHub\ML\venv\Scripts\python.exe C:/Users/Administrator/Documents/GitHub/ML/Assignment2/Assignment2_Question2.py
    Duration  Pulse  Maxpulse    Calories
0         60    110       130  409.100000
1         60    117       145  479.000000
2         60    103       135  340.000000
3         45    109       175  282.400000
4         45    117       148  406.000000
5         60    102       127  300.000000
6         60    110       136  374.000000
7         45    104       134  253.300000
8         30    109       133  195.100000
9         60     98       124  269.000000
10        60    103       147  329.300000
11        60    100       120  250.700000
12        60    106       128  345.300000
13        60    104       132  379.300000
14        60     98       123  275.000000
15        60     98       120  215.200000
16        60    100       120  300.000000
17        45     90       112  375.790244
18        60    103       123  323.000000
19        45     97       125  243.000000
20        60    108       131  364.200000
21        45    100       119  282.000000
22        60    130       101  300.000000
23        45    105       132  246.000000
24        60    102       126  334.500000
```

2.2

describe() gives basic statistical description about the data.

```
# 2. 2 Show the basic statistical description about the data.
print(df.describe())
```

2.2 Output

```
          Duration       Pulse    Maxpulse     Calories
count   169.000000  169.000000  169.000000   169.000000
mean     63.846154  107.461538  134.047337   375.790244
std      42.299949   14.510259   16.450434   262.385991
min      15.000000   80.000000  100.000000    50.300000
25%      45.000000  100.000000  124.000000   253.300000
50%      60.000000  105.000000  131.000000   321.000000
75%      60.000000  111.000000  141.000000   384.000000
max     300.000000  159.000000  184.000000  1860.400000
```

2.3

Check if the data has null values. a. Replace the null values with the mean

```
# 2. 3 Check if the data has null values. a. Replace the null values with the mean
df.fillna(df.mean(), inplace=True)
print(df.isnull().any())
```

2.3 Output

```
Duration    False
Pulse       False
Maxpulse    False
Calories    False
dtype: bool
```

2.4

Select at least two columns and aggregate the data using: min, max, count, mean.

```
print("\n")
# 2. 4 Select at least two columns and aggregate the data using: min, max, count, mean.
print(df.agg({'Duration':['min','max','count','mean'],'Pulse':['min','max','count','mean']}))
```

2.4 Output

```
          Duration       Pulse
min      15.000000   80.000000
max     300.000000  159.000000
count   169.000000  169.000000
mean     63.846154  107.461538
```

2.5

Filter the dataframe to select the rows with calories values between 500 and 1000.

```
25    print("\n")
26    # 2. 5 Filter the dataframe to select the rows with calories values between 500 and 1000.
27    print(df.loc[(df['Calories']>500)&(df['Calories']<1000)])
```

2.5 Output

```
     Duration  Pulse  Maxpulse  Calories
51         80    123       146     643.1
62        160    109       135     853.0
65        180     90       130     800.4
66        150    105       135     873.4
67        150    107       130     816.0
72         90    100       127     700.0
73        150     97       127     953.2
75         90     98       125     563.2
78        120    100       130     500.4
90        180    101       127     600.1
99         90     93       124     604.1
103        90     90       100     500.4
106       180     90       120     800.3
108        90     90       120     500.3
```

2.6

Filter the dataframe to select the rows with calories values > 500 and pulse < 100.

```
29  print("\n")
30  # 2. 6 Filter the dataframe to select the rows with calories values > 500 and pulse < 100.
31  print(df.loc[(df['Calories']>500)&(df['Pulse']<100)])
```

2.6 Output

```
     Duration  Pulse  Maxpulse  Calories
65        180     90       130     800.4
70        150     97       129    1115.0
73        150     97       127     953.2
75         90     98       125     563.2
99         90     93       124     604.1
103        90     90       100     500.4
106       180     90       120     800.3
108        90     90       120     500.3
```

2.7

Create a new “df_modified” dataframe that contains all the columns from df except for “Maxpulse”.

```
33  print("\n")
34  # 2. 7 Create a new “df_modified” dataframe that contains all the columns from df except for “Maxpulse”.
35  df_modified = df[['Duration','Pulse','Calories']]
36  print(df_modified.head())
```

2.7 Output

```
   Duration  Pulse  Calories
0        60    110     409.1
1        60    117     479.0
2        60    103     340.0
3        45    109     282.4
4        45    117     406.0
```

2.8

Delete the “Maxpulse” column from the main df dataframe

```
39  # 2. 8 Delete the “Maxpulse” column from the main df dataframe
40  del df['Maxpulse']
41  print(df.head())
```

2.8 Output

```
   Duration  Pulse  Calories
0        60    110     409.1
1        60    117     479.0
2        60    103     340.0
3        45    109     282.4
4        45    117     406.0
```

2.9

Convert the datatype of Calories column to int datatype.

```
# 2. 9 Convert the datatype of Calories column to int datatype.
print(df.dtypes)
print("\n")
df['Calories'] = df['Calories'].astype(np.int64)
print(df.dtypes)
```

2.9 Output

```
Duration      int64
Pulse         int64
Calories    float64
dtype: object


Duration    int64
Pulse       int64
Calories    int64
dtype: object
```

2.10

Using pandas create a scatter plot for the two columns (Duration and Calories

```
print("\n")
# 2. 10 Using pandas create a scatter plot for the two columns (Duration and Calories).
print(df.plot.scatter(x='Duration',y='Calories',c='DarkBlue'))
```

2.10 Output

```
AxesSubplot(0.125,0.11;0.775x0.77)
```

Question3:

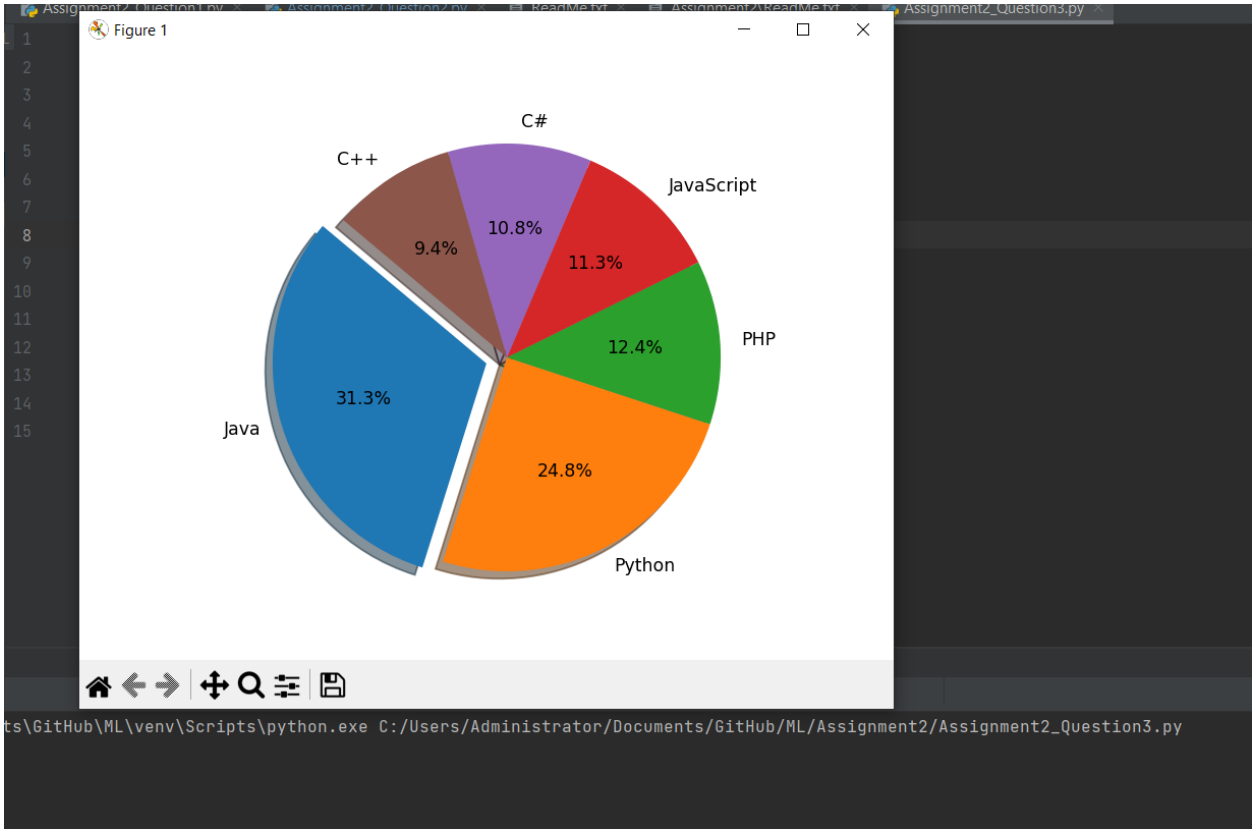
Python program to display the pue chart with different programming languages

```
import matplotlib.pyplot as plt

# Data to plot
prog_languages = 'Java', 'Python', 'PHP', 'JavaScript', 'C#', 'C++'
popularity = [22.2, 17.6, 8.8, 8, 7.7, 6.7]
colors = ["#1f77b4", "#ff7f0e", "#2ca02c", "#d62728", "#9467bd", "#8c564b"]
# explode 1st slice
explode = (0.1, 0, 0, 0, 0, 0)
# Plot
plt.pie(popularity, explode=explode, labels=prog_languages, colors=colors,
        autopct='%1.1f%%', shadow=True, startangle=140)

plt.axis('equal')
plt.show()
```

Question3 Output



Related Links:

Source Code:
https://github.com/VijayTarakaRamarao/ML/tree/main/Assignment2

Video Recording:
https://github.com/VijayTarakaRamarao/ML/blob/main/Assignment2/ML_Assignment2_Recording.mp4