Name: Rohini Patturaja

Subject: Big Data Analytics

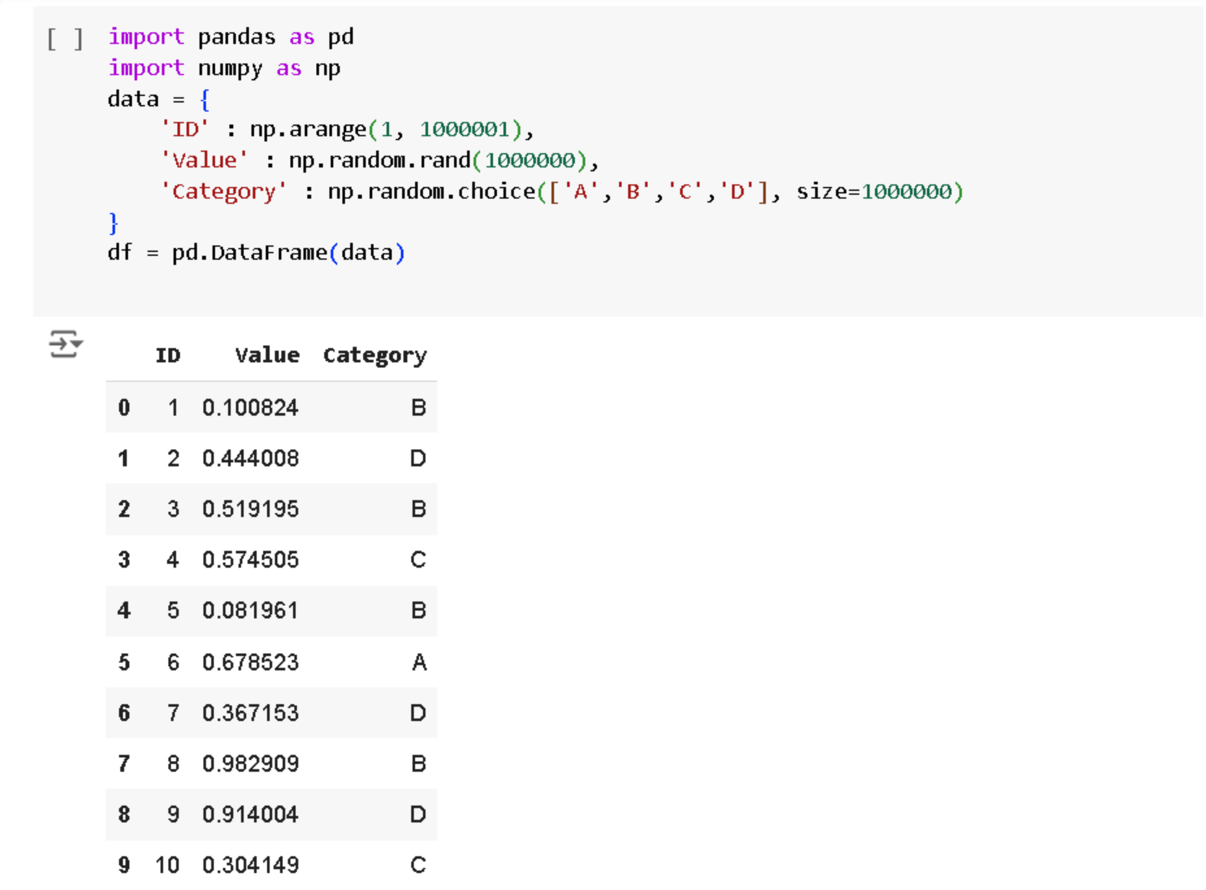
Course ID: DSA 5620

Email: rxp36770@ucmo.edu

Github link: Github ICP3 link

Video link: Rohini_Patturaja_ICP3.mp4

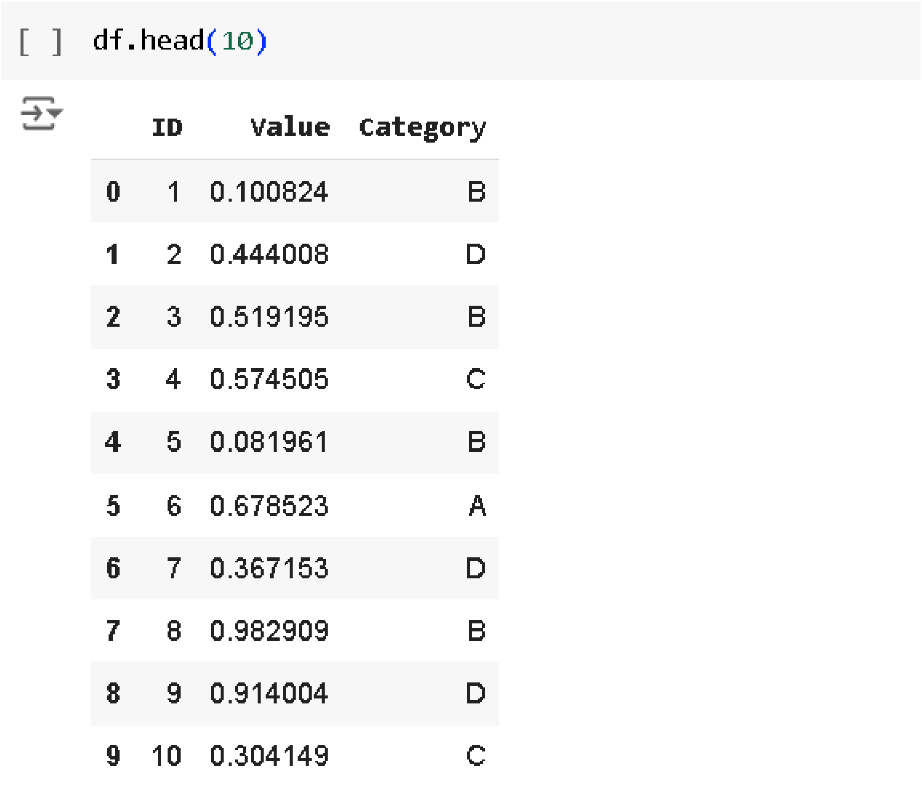
1. Creating a DataFrame from a given dictionary



```
[ ]  import pandas as pd
     import numpy as np
     data = {
         'ID' : np.arange(1, 1000001),
         'Value' : np.random.rand(1000000),
         'Category' : np.random.choice(['A','B','C','D'], size=1000000)
     }
     df = pd.DataFrame(data)
```

	ID	Value	Category
0	1	0.100824	B
1	2	0.444008	D
2	3	0.519195	B
3	4	0.574505	C
4	5	0.081961	B
5	6	0.678523	A
6	7	0.367153	D
7	8	0.982909	B
8	9	0.914004	D
9	10	0.304149	C

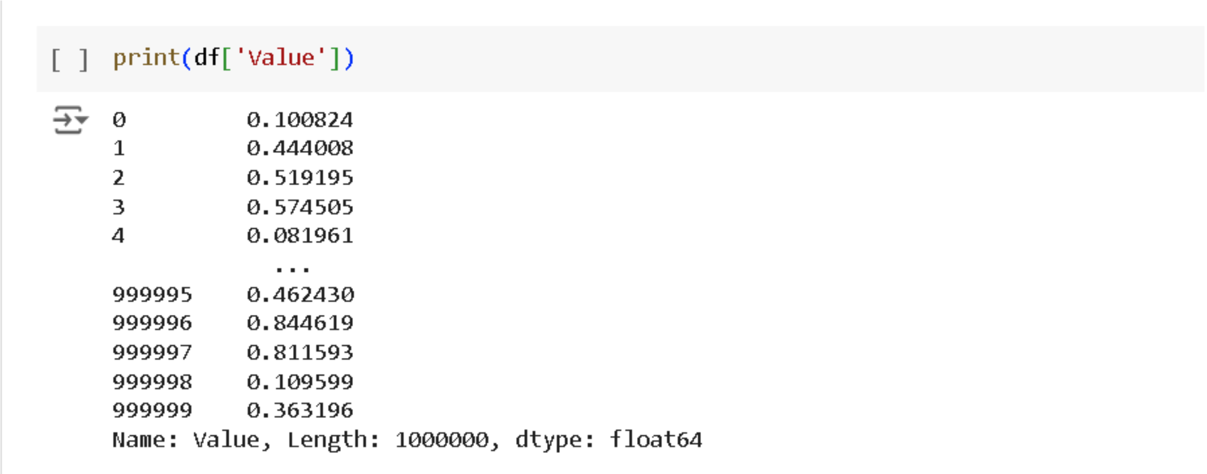
2. Output first 10 rows.



```
[ ] df.head(10)
```

	ID	Value	Category
0	1	0.100824	B
1	2	0.444008	D
2	3	0.519195	B
3	4	0.574505	C
4	5	0.081961	B
5	6	0.678523	A
6	7	0.367153	D
7	8	0.982909	B
8	9	0.914004	D
9	10	0.304149	C

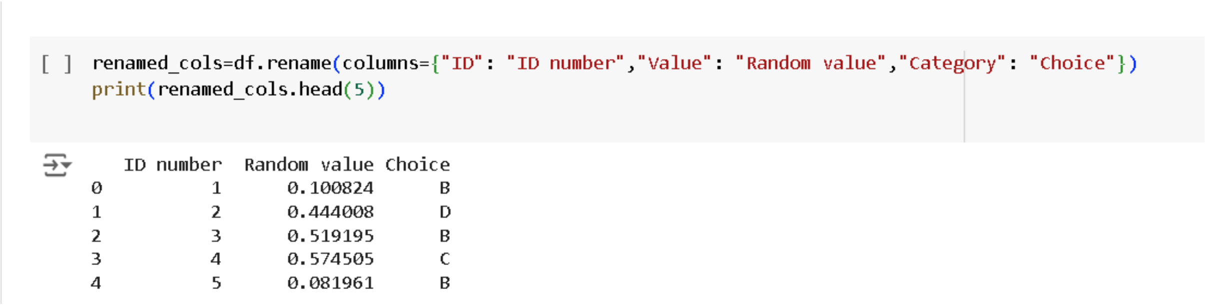
3. Access a column “Value”



```
[ ] print(df['Value'])
```

```
0          0.100824
1          0.444008
2          0.519195
3          0.574505
4          0.081961
             ...
999995     0.462430
999996     0.844619
999997     0.811593
999998     0.109599
999999     0.363196
Name: Value, Length: 1000000, dtype: float64
```

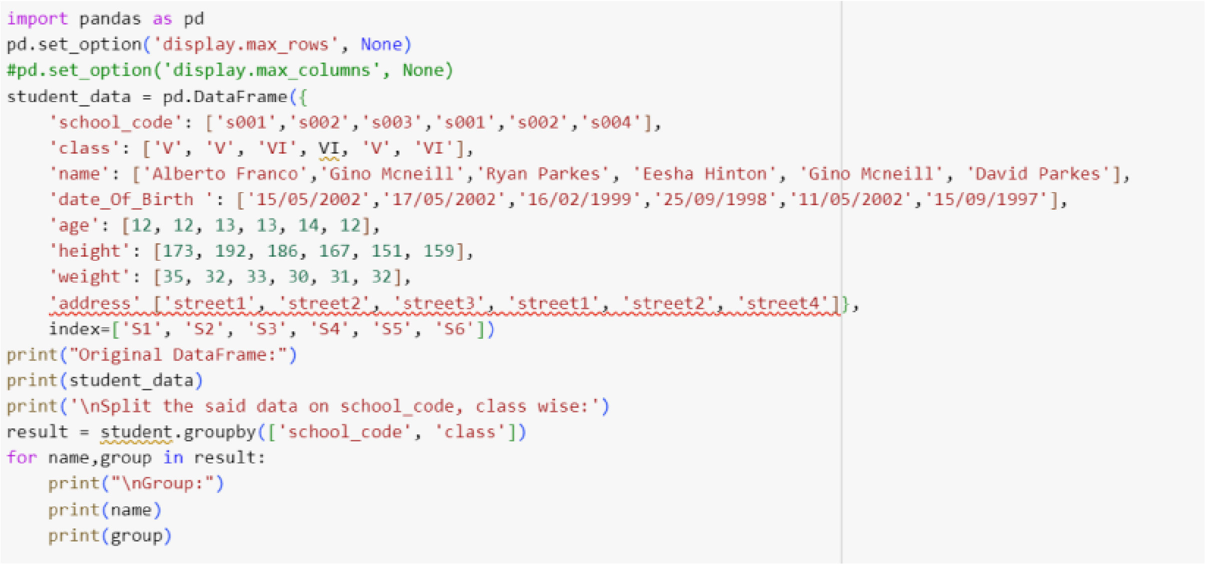
4. Modify columns in the DataFrame with names (ID number, Random value, Choice) and show output for first five rows.



```
[ ]  renamed_cols=df.rename(columns={"ID": "ID number","Value": "Random value","Category": "Choice"})
     print(renamed_cols.head(5))
```

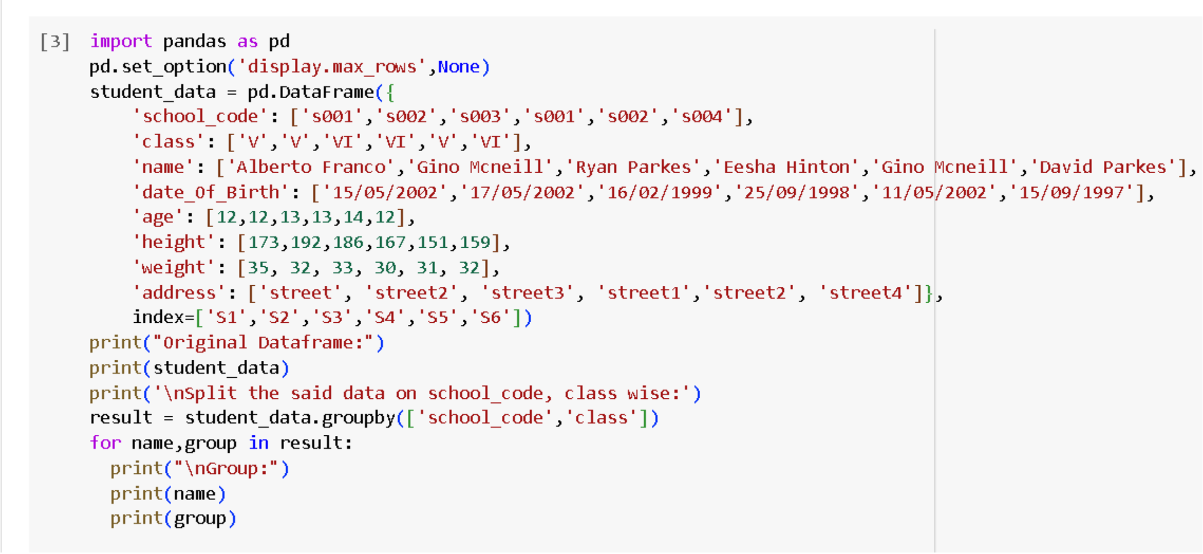
```
      ID number  Random value Choice
   0          1      0.100824      B
   1          2      0.444008      D
   2          3      0.519195      B
   3          4      0.574505      C
   4          5      0.081961      B
```

5. Run the below given code by removing bugs and errors

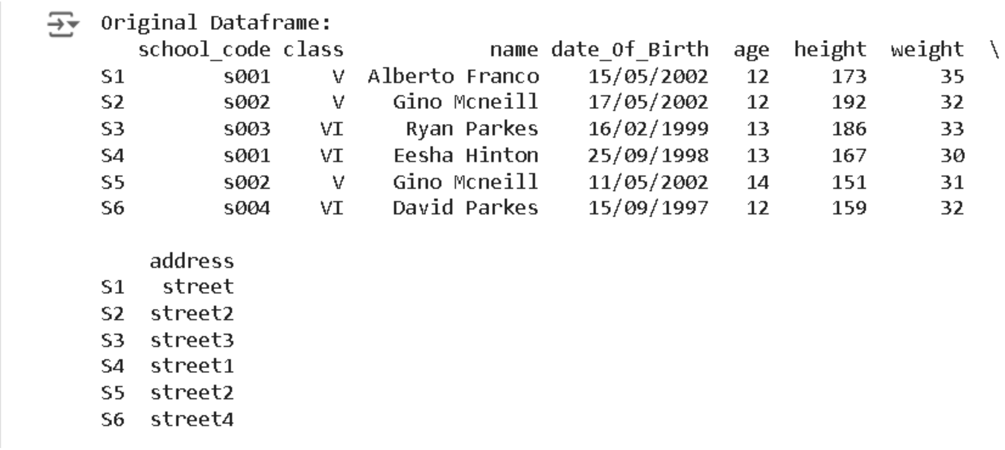


```
import pandas as pd
pd.set_option('display.max_rows', None)
#pd.set_option('display.max_columns', None)
student_data = pd.DataFrame({
    'school_code': ['s001','s002','s003','s001','s002','s004'],
    'class': ['V', 'V', 'VI', VI, 'V', 'VI'],
    'name': ['Alberto Franco','Gino Mcneill','Ryan Parkes', 'Eesha Hinton', 'Gino Mcneill', 'David Parkes'],
    'date_Of_Birth ': ['15/05/2002','17/05/2002','16/02/1999','25/09/1998','11/05/2002','15/09/1997'],
    'age': [12, 12, 13, 13, 14, 12],
    'height': [173, 192, 186, 167, 151, 159],
    'weight': [35, 32, 33, 30, 31, 32],
    'address' ['street1', 'street2', 'street3', 'street1', 'street2', 'street4']},
    index=['S1', 'S2', 'S3', 'S4', 'S5', 'S6'])
print("Original DataFrame:")
print(student_data)
print('\nSplit the said data on school_code, class wise:')
result = student.groupby(['school_code', 'class'])
for name,group in result:
    print("\nGroup:")
    print(name)
    print(group)
```

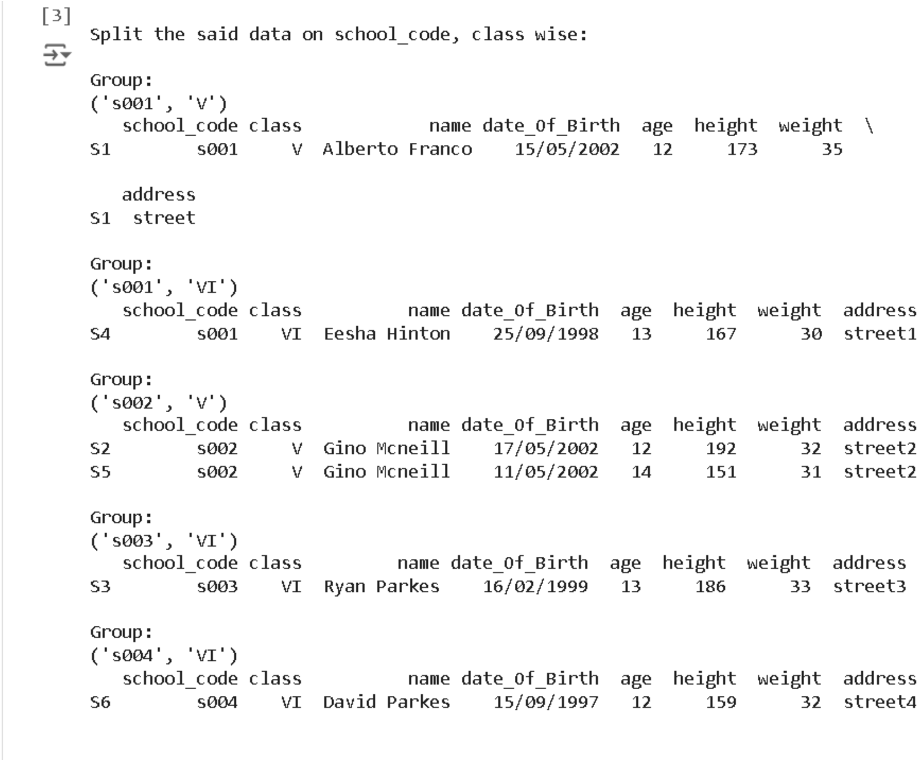
Corrected Code:



```
[3]  import pandas as pd
     pd.set_option('display.max_rows',None)
     student_data = pd.DataFrame({
         'school_code': ['s001','s002','s003','s001','s002','s004'],
         'class': ['V','V','VI','VI','V','VI'],
         'name': ['Alberto Franco','Gino Mcneill','Ryan Parkes','Eesha Hinton','Gino Mcneill','David Parkes'],
         'date_Of_Birth': ['15/05/2002','17/05/2002','16/02/1999','25/09/1998','11/05/2002','15/09/1997'],
         'age': [12,12,13,13,14,12],
         'height': [173,192,186,167,151,159],
         'weight': [35, 32, 33, 30, 31, 32],
         'address': ['street', 'street2', 'street3', 'street1','street2', 'street4']},
         index=['S1','S2','S3','S4','S5','S6'])
     print("Original Dataframe:")
     print(student_data)
     print('\nSplit the said data on school_code, class wise:')
     result = student_data.groupby(['school_code','class'])
     for name,group in result:
       print("\nGroup:")
       print(name)
       print(group)
```

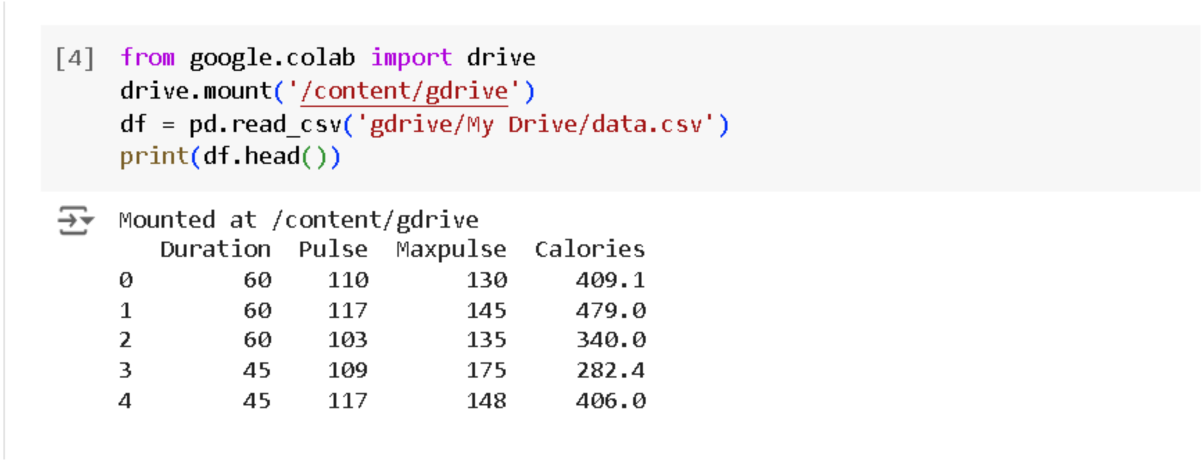


```
Original Dataframe:
   school_code class           name date_Of_Birth  age  height  weight  \
S1        s001     V  Alberto Franco    15/05/2002   12     173      35
S2        s002     V    Gino Mcneill    17/05/2002   12     192      32
S3        s003    VI     Ryan Parkes    16/02/1999   13     186      33
S4        s001    VI    Eesha Hinton    25/09/1998   13     167      30
S5        s002     V    Gino Mcneill    11/05/2002   14     151      31
S6        s004    VI    David Parkes    15/09/1997   12     159      32

    address
S1   street
S2  street2
S3  street3
S4  street1
S5  street2
S6  street4
```



```
[3]
Split the said data on school_code, class wise:

Group:
('s001', 'V')
   school_code class            name date_Of_Birth  age  height  weight  \
S1        s001     V  Alberto Franco    15/05/2002   12     173      35

   address
S1  street

Group:
('s001', 'VI')
   school_code class          name date_Of_Birth  age  height  weight  address
S4        s001    VI  Eesha Hinton    25/09/1998   13     167      30  street1

Group:
('s002', 'V')
   school_code class          name date_Of_Birth  age  height  weight  address
S2        s002     V  Gino Mcneill    17/05/2002   12     192      32  street2
S5        s002     V  Gino Mcneill    11/05/2002   14     151      31  street2

Group:
('s003', 'VI')
   school_code class         name date_Of_Birth  age  height  weight  address
S3        s003    VI  Ryan Parkes    16/02/1999   13     186      33  street3

Group:
('s004', 'VI')
   school_code class          name date_Of_Birth  age  height  weight  address
S6        s004    VI  David Parkes    15/09/1997   12     159      32  street4
```

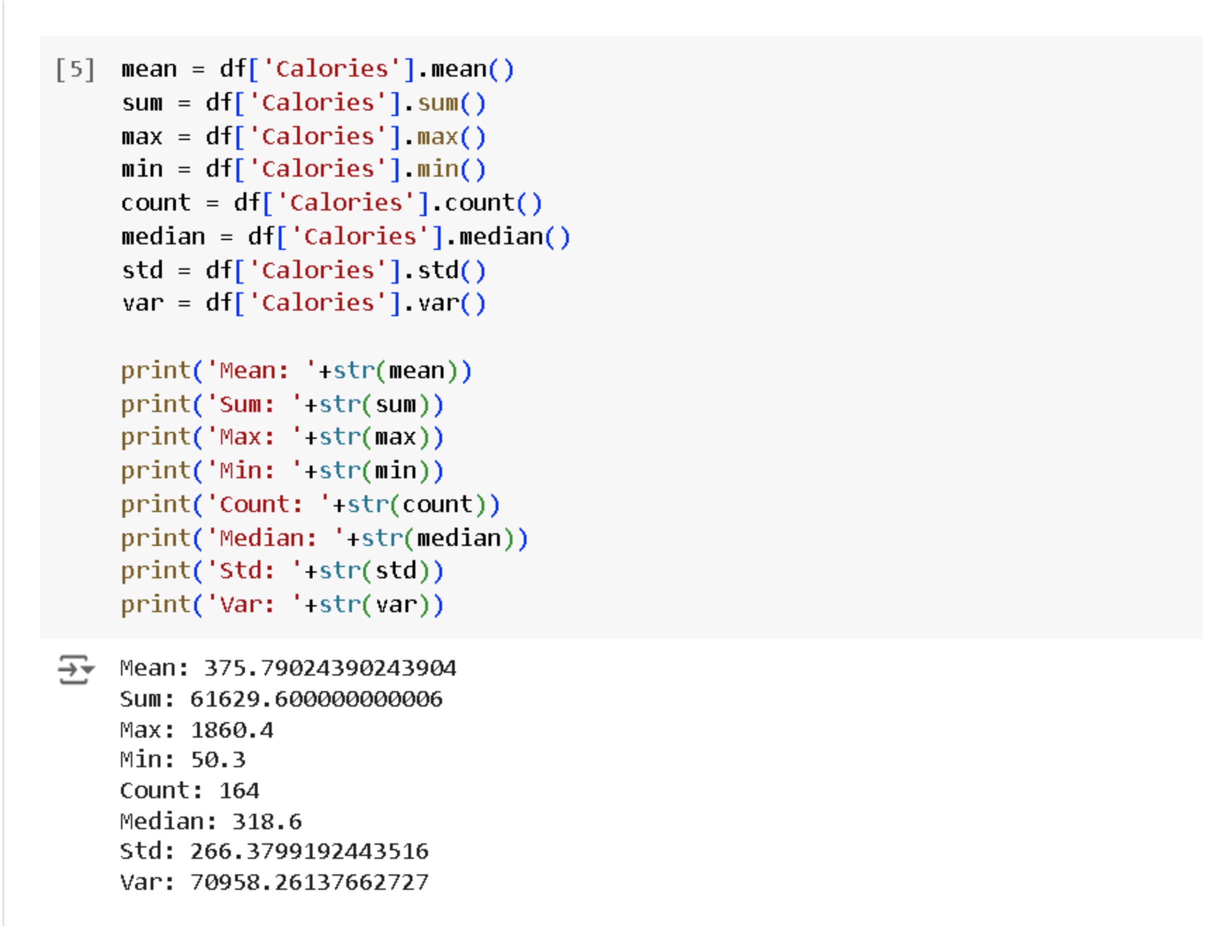
6. Read the provided CSV file ‘data.csv’.
https://drive.google.com/drive/folders/1h8C3mLsso-R-sIOLsvoYwPLzy2fJ4IOF?usp=sharing



```
[4] from google.colab import drive
    drive.mount('/content/gdrive')
    df = pd.read_csv('gdrive/My Drive/data.csv')
    print(df.head())
```

```
Mounted at /content/gdrive
   Duration  Pulse  Maxpulse  Calories
0        60    110       130     409.1
1        60    117       145     479.0
2        60    103       135     340.0
3        45    109       175     282.4
4        45    117       148     406.0
```

7. Show the basic statistical description about the data.



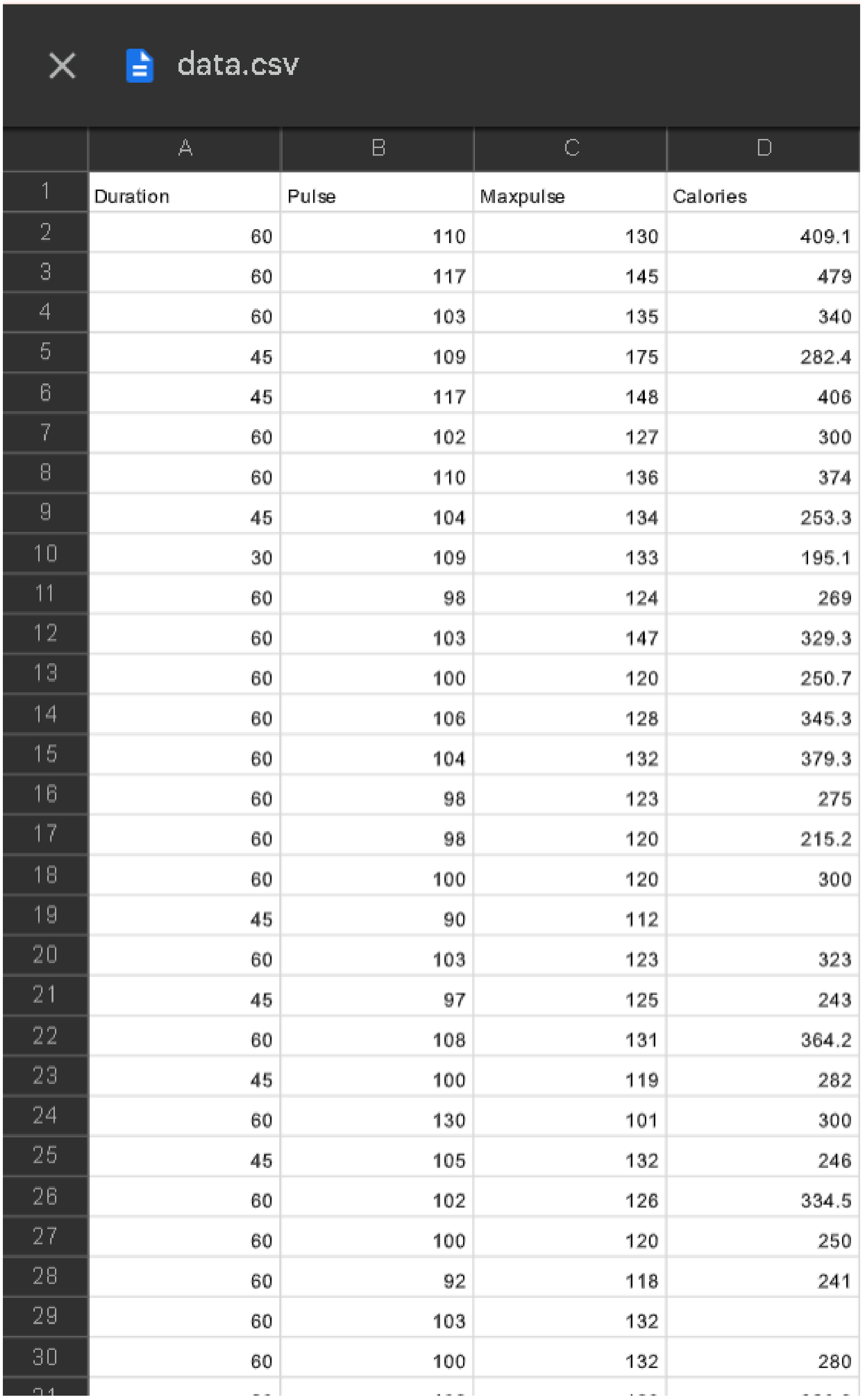
```
[5] mean = df['Calories'].mean()
    sum = df['Calories'].sum()
    max = df['Calories'].max()
    min = df['Calories'].min()
    count = df['Calories'].count()
    median = df['Calories'].median()
    std = df['Calories'].std()
    var = df['Calories'].var()

    print('Mean: '+str(mean))
    print('Sum: '+str(sum))
    print('Max: '+str(max))
    print('Min: '+str(min))
    print('Count: '+str(count))
    print('Median: '+str(median))
    print('Std: '+str(std))
    print('Var: '+str(var))
```

```
Mean: 375.79024390243904
Sum: 61629.600000000006
Max: 1860.4
Min: 50.3
Count: 164
Median: 318.6
Std: 266.3799192443516
Var: 70958.26137662727
```

8. . Check if the data has null values.

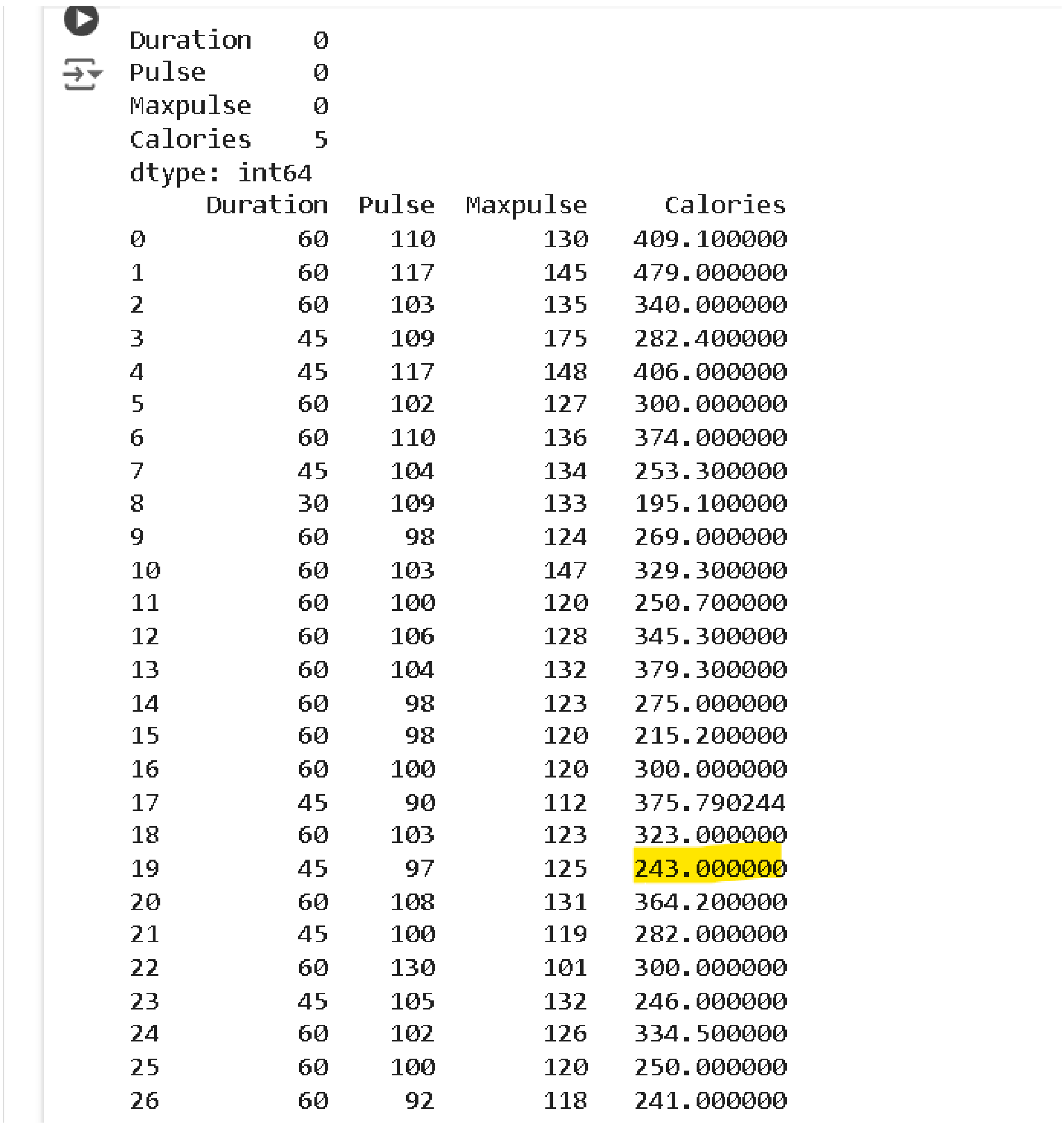
a. Replace the null values with the mean

```
[8] null_values = df.isnull().sum()
    mean_values = df.fillna(df.mean())
    print(null_values)
    print(mean_values)
```

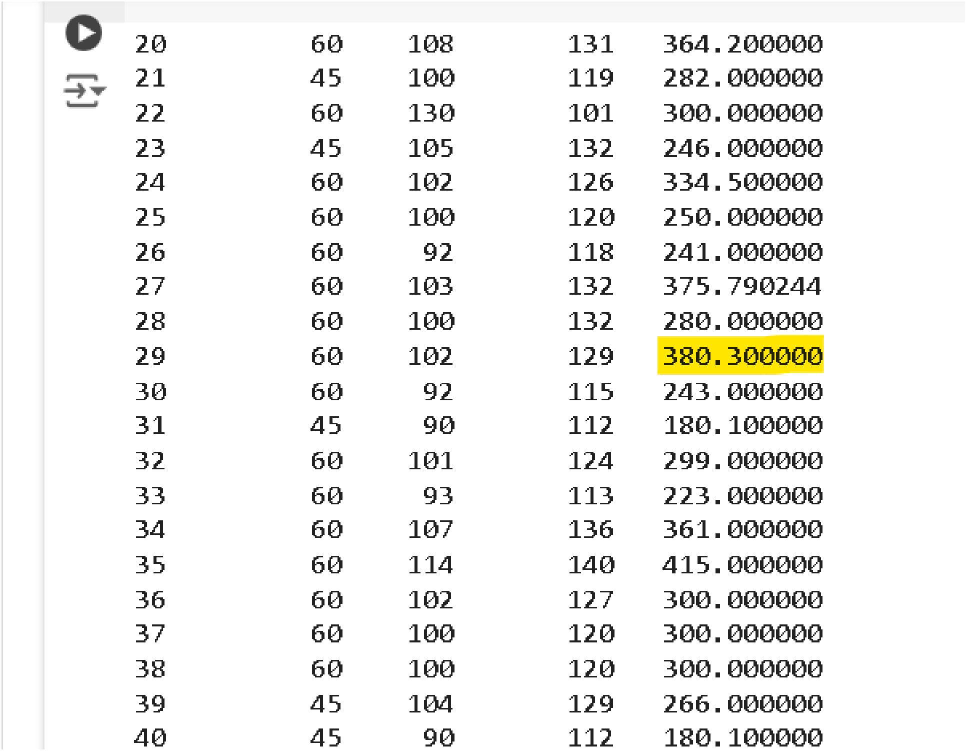


data.csv

	A	B	C	D
1	Duration	Pulse	Maxpulse	Calories
2	60	110	130	409.1
3	60	117	145	479
4	60	103	135	340
5	45	109	175	282.4
6	45	117	148	406
7	60	102	127	300
8	60	110	136	374
9	45	104	134	253.3
10	30	109	133	195.1
11	60	98	124	269
12	60	103	147	329.3
13	60	100	120	250.7
14	60	106	128	345.3
15	60	104	132	379.3
16	60	98	123	275
17	60	98	120	215.2
18	60	100	120	300
19	45	90	112	
20	60	103	123	323
21	45	97	125	243
22	60	108	131	364.2
23	45	100	119	282
24	60	130	101	300
25	45	105	132	246
26	60	102	126	334.5
27	60	100	120	250
28	60	92	118	241
29	60	103	132	
30	60	100	132	280

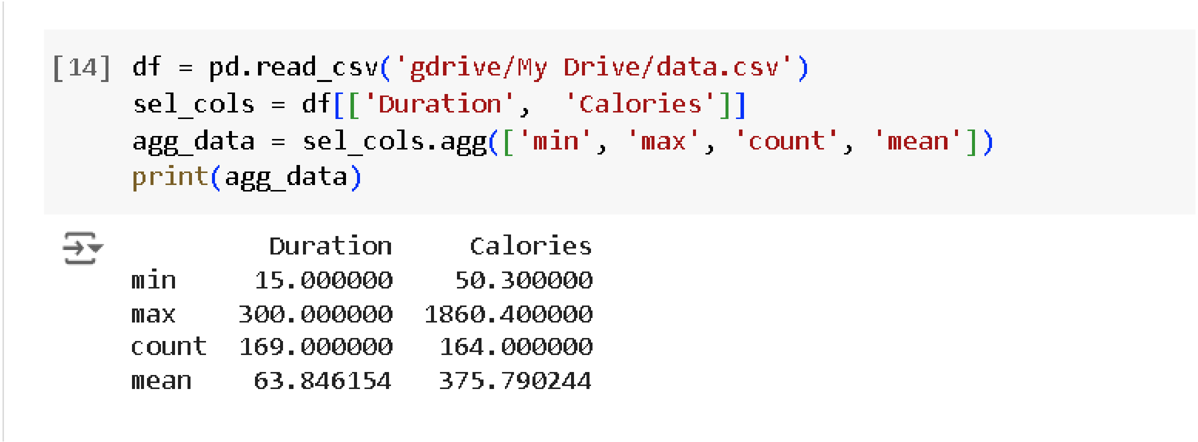


```
Duration    0
Pulse       0
Maxpulse    0
Calories    5
dtype: int64
    Duration  Pulse  Maxpulse    Calories
0         60    110       130  409.100000
1         60    117       145  479.000000
2         60    103       135  340.000000
3         45    109       175  282.400000
4         45    117       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
25        60    100       120  250.000000
26        60     92       118  241.000000
```



```
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
25         60   100      120   250.000000
26         60    92      118   241.000000
27         60   103      132   375.790244
28         60   100      132   280.000000
29         60   102      129   380.300000
30         60    92      115   243.000000
31         45    90      112   180.100000
32         60   101      124   299.000000
33         60    93      113   223.000000
34         60   107      136   361.000000
35         60   114      140   415.000000
36         60   102      127   300.000000
37         60   100      120   300.000000
38         60   100      120   300.000000
39         45   104      129   266.000000
40         45    90      112   180.100000
```

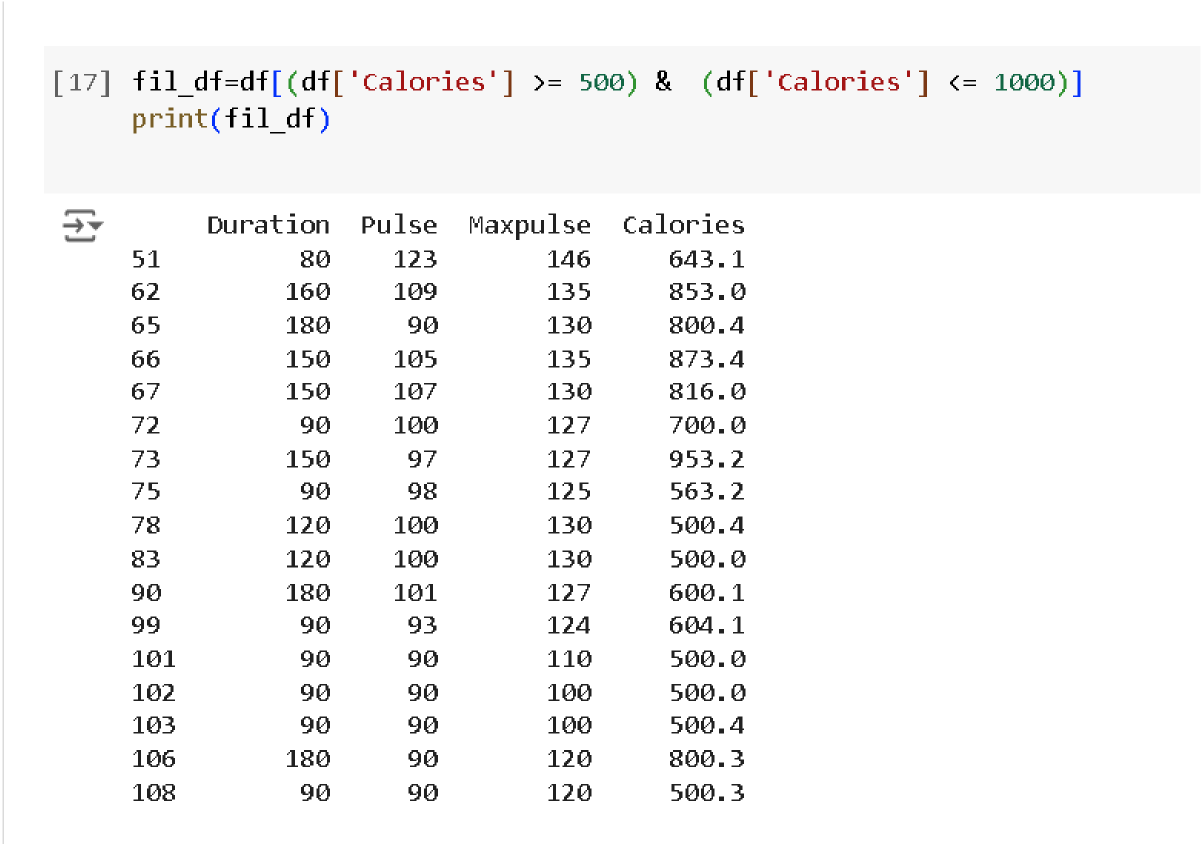
9. Select at least two columns and aggregate the data using: min, max, count, mean.



```
[14] df = pd.read_csv('gdrive/My Drive/data.csv')
     sel_cols = df[['Duration',  'Calories']]
     agg_data = sel_cols.agg(['min', 'max', 'count', 'mean'])
     print(agg_data)
```

```
         Duration     Calories
min     15.000000    50.300000
max    300.000000  1860.400000
count  169.000000   164.000000
mean    63.846154   375.790244
```

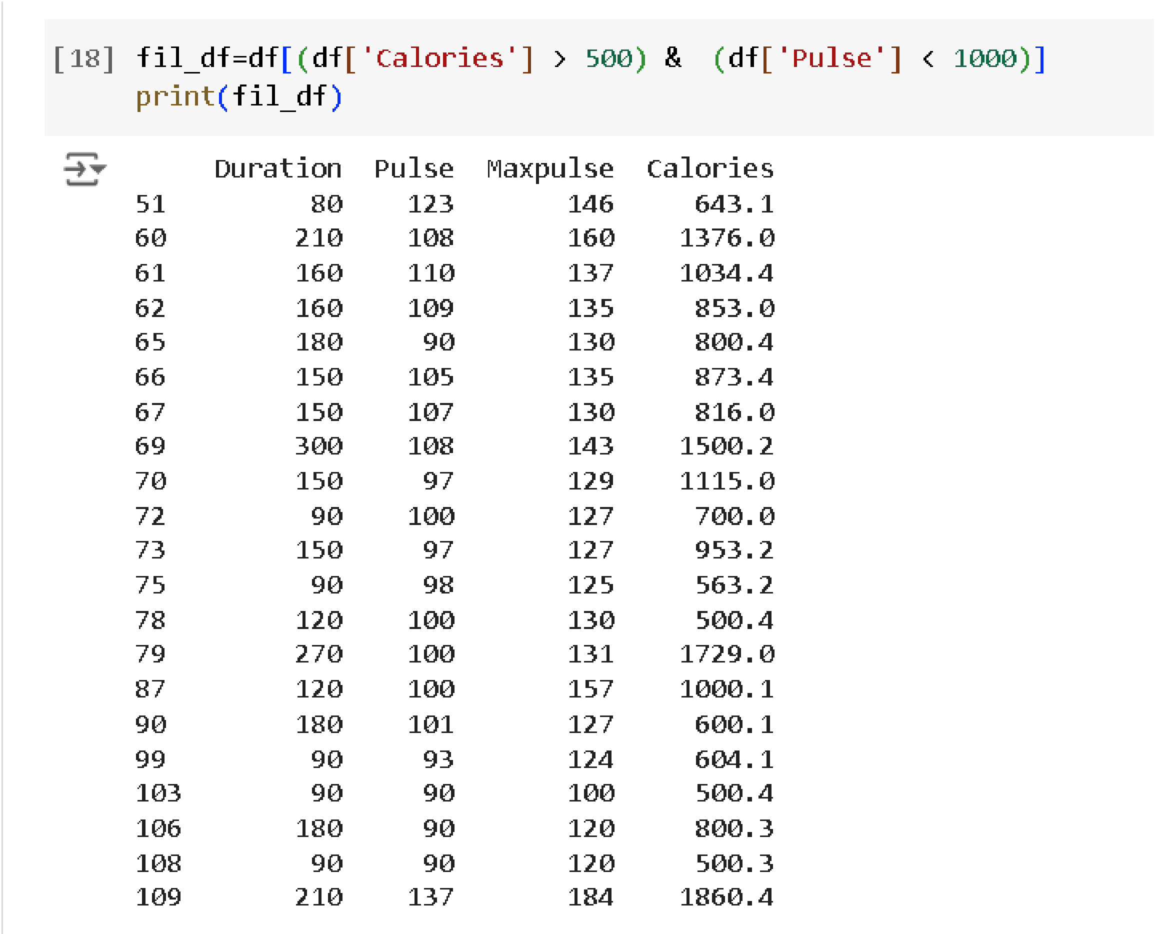
10. Filter the dataframe to select the rows with calories values between 500 and 1000.



```
[17] fil_df=df[(df['Calories'] >= 500) &  (df['Calories'] <= 1000)]
     print(fil_df)
```

```
     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
83        120    100       130     500.0
90        180    101       127     600.1
99         90     93       124     604.1
101        90     90       110     500.0
102        90     90       100     500.0
103        90     90       100     500.4
106       180     90       120     800.3
108        90     90       120     500.3
```

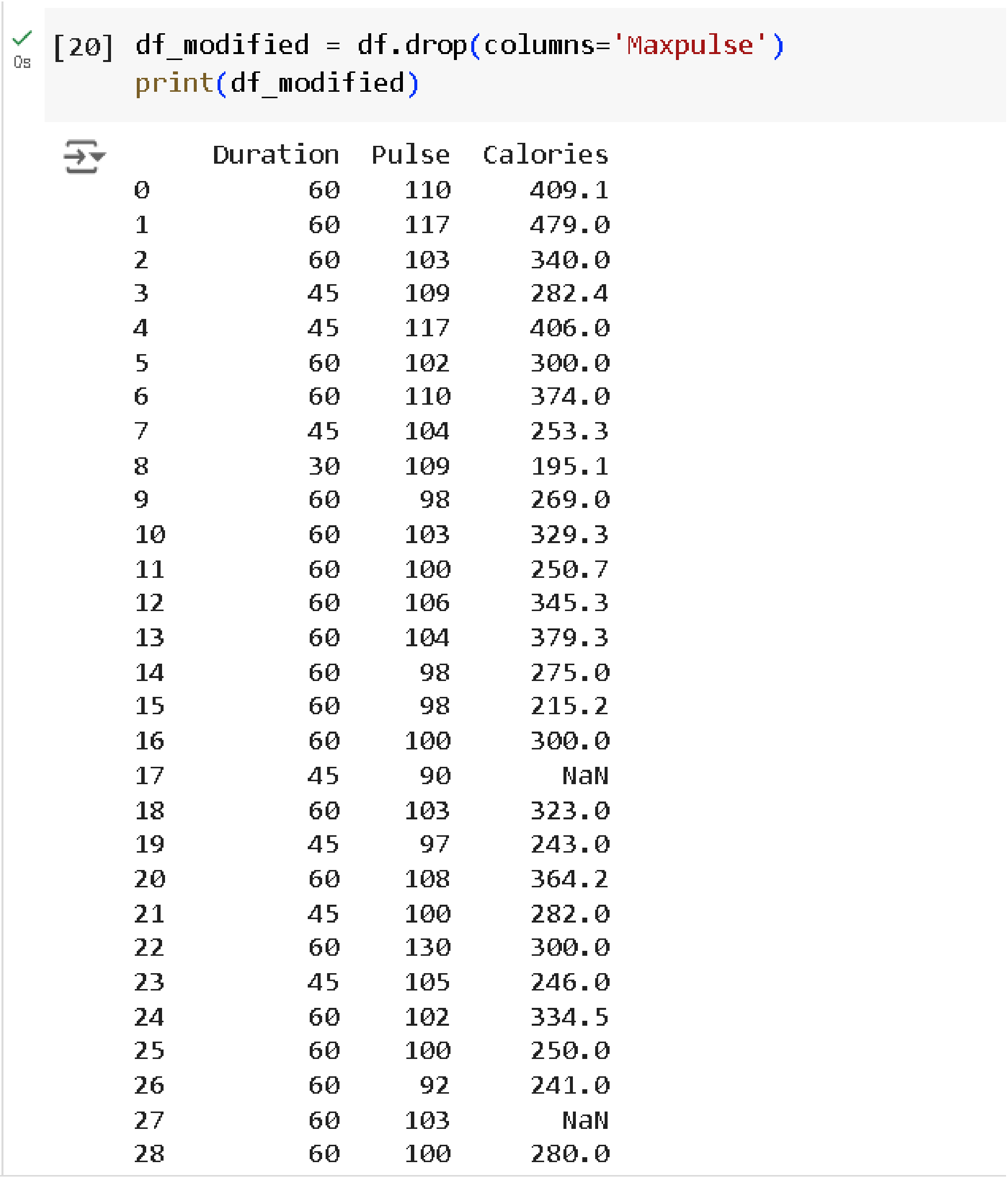
11. Filter the dataframe to select the rows with calories values > 500 and pulse < 100.



```
[18] fil_df=df[(df['Calories'] > 500) &  (df['Pulse'] < 1000)]
     print(fil_df)
```

```
     Duration  Pulse  Maxpulse  Calories
51         80    123       146     643.1
60        210    108       160    1376.0
61        160    110       137    1034.4
62        160    109       135     853.0
65        180     90       130     800.4
66        150    105       135     873.4
67        150    107       130     816.0
69        300    108       143    1500.2
70        150     97       129    1115.0
72         90    100       127     700.0
73        150     97       127     953.2
75         90     98       125     563.2
78        120    100       130     500.4
79        270    100       131    1729.0
87        120    100       157    1000.1
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
109       210    137       184    1860.4
```

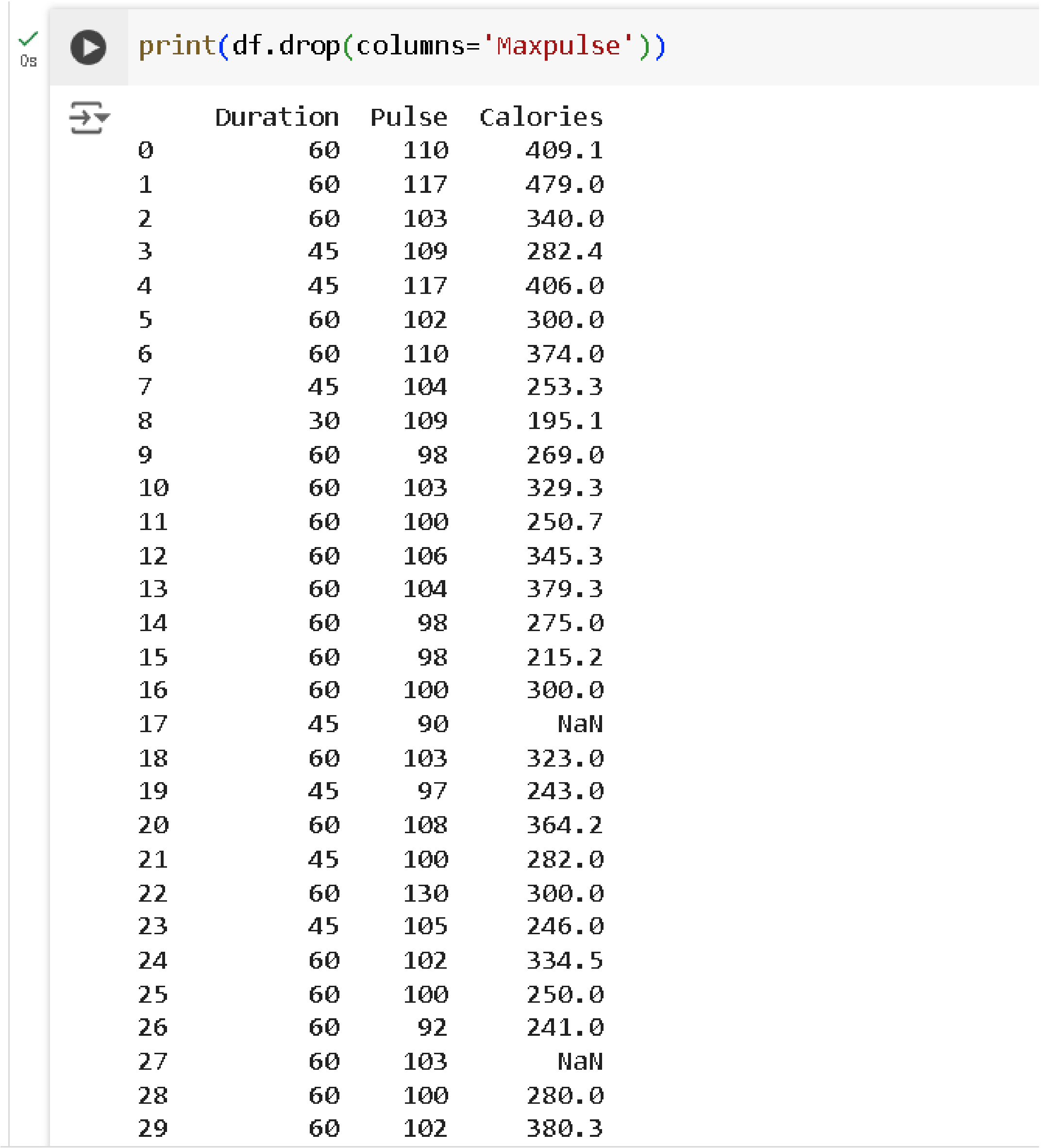
12. Create a new “df_modified” dataframe that contains all the columns from df except for “Maxpulse”



```
df_modified = df.drop(columns='Maxpulse')
print(df_modified)
```

```
    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
5         60    102     300.0
6         60    110     374.0
7         45    104     253.3
8         30    109     195.1
9         60     98     269.0
10        60    103     329.3
11        60    100     250.7
12        60    106     345.3
13        60    104     379.3
14        60     98     275.0
15        60     98     215.2
16        60    100     300.0
17        45     90       NaN
18        60    103     323.0
19        45     97     243.0
20        60    108     364.2
21        45    100     282.0
22        60    130     300.0
23        45    105     246.0
24        60    102     334.5
25        60    100     250.0
26        60     92     241.0
27        60    103       NaN
28        60    100     280.0
```

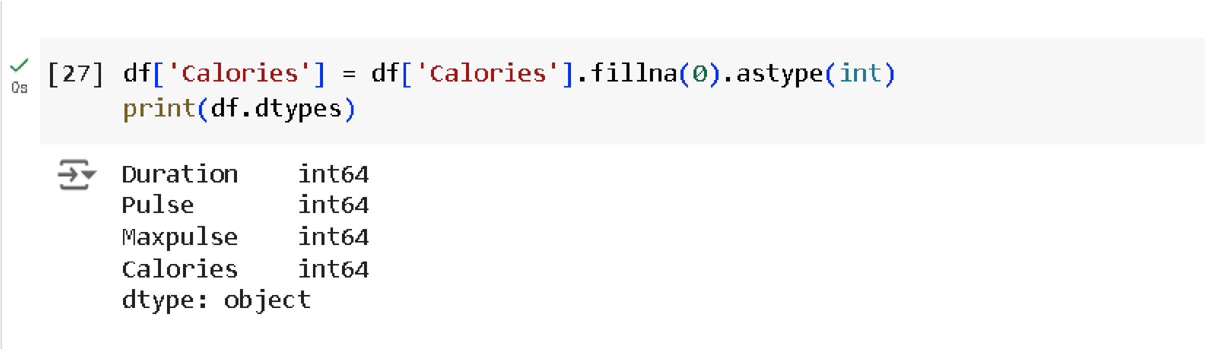
13. Delete the “Maxpulse” column from the main df dataframe



```
print(df.drop(columns='Maxpulse'))
```

```
    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
5         60    102     300.0
6         60    110     374.0
7         45    104     253.3
8         30    109     195.1
9         60     98     269.0
10        60    103     329.3
11        60    100     250.7
12        60    106     345.3
13        60    104     379.3
14        60     98     275.0
15        60     98     215.2
16        60    100     300.0
17        45     90       NaN
18        60    103     323.0
19        45     97     243.0
20        60    108     364.2
21        45    100     282.0
22        60    130     300.0
23        45    105     246.0
24        60    102     334.5
25        60    100     250.0
26        60     92     241.0
27        60    103       NaN
28        60    100     280.0
29        60    102     380.3
```

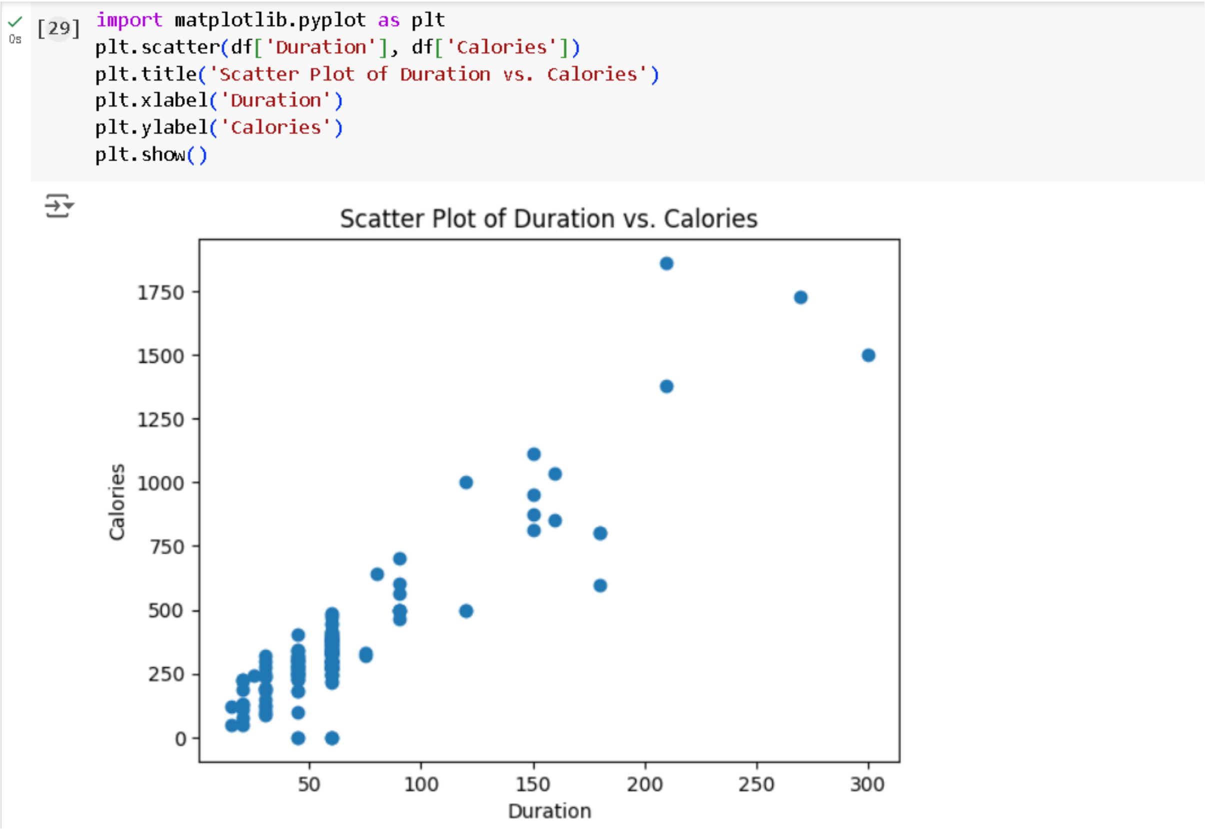
14. Convert the datatype of Calories column to int datatype.



```
[27] df['Calories'] = df['Calories'].fillna(0).astype(int)
     print(df.dtypes)
```

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

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



```
import matplotlib.pyplot as plt
plt.scatter(df['Duration'], df['Calories'])
plt.title('Scatter Plot of Duration vs. Calories')
plt.xlabel('Duration')
plt.ylabel('Calories')
plt.show()
```