**Question:**

Read in the CSV dataset that is found at the following URL:

https://raw.githubusercontent.com/ryanleeallred/datasets/master/messy-data.csv

Print out the shape as well as the first 5 rows of the dataframe.Print out the datatypes of the dataframe columns (dataset features). Print out the summary statistics of the numeric values of your dataset i.e. min, max, mean, standard deviation, etc.Clean all NaN values from the dataset.Describe how you addressed the NaN values and give an explanation justifying your decision. Create scatter plots using Matplotlib. Can you find any interesting relationships in the data? Be sure to label your axis and to give your graphs a title.

* Comment
* Flag Question
* Flag for Copyright

**Subject:**

* Python Programming
* Due in 19 hours 58 minutes

**Reminder:** Don't copy and paste from somewhere else. Write original answers and cite sources.

**Answer:**

**The output of the result: -**

* *Displaying top five columns*

-----------------------------------------------------------------------

alpha beta gamma delta epsilon zeta eta

0 2 48 12 240 3.0 Yes AZ

1 3 46 18 230 5.0 NaN VT

2 4 44 24 220 7.0 No PA

3 5 42 30 210 9.0 Yes OK

4 6 44 36 220 11.0 Yes MD

-----------------------------------------------------------------------

* *DF info*

Size = 1155

**Shape = (165, 7)**

Dimension = 2

-----------------------------------------------------------------------

* *Columns info*

**alpha int64**

**beta int64**

**gamma int64**

**delta int64**

**epsilon float64**

**zeta object**

**eta object**

**dtype: object**

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 165 entries, 0 to 164

Data columns (total 7 columns):

alpha 165 non-null int64

beta 165 non-null int64

gamma 165 non-null int64

delta 165 non-null int64

epsilon 141 non-null float64

zeta 146 non-null object

eta 165 non-null object

dtypes: float64(1), int64(4), object(2)

memory usage: 9.1+ KB

None

-----------------------------------------------------------------------

Stats

alpha beta gamma delta epsilon

**count 165.000000 165.000000 165.000000 165.000000 141.000000**

**mean 46.169697 17.921212 277.018182 89.606061 164.794326**

**std 24.371521 17.769180 146.229125 88.845900 96.430843**

**min 2.000000 -11.000000 12.000000 -55.000000 3.000000**

25% 28.000000 7.000000 168.000000 35.000000 83.000000

50% 44.000000 15.000000 264.000000 75.000000 163.000000

75% 65.000000 32.000000 390.000000 160.000000 247.000000

**max 85.000000 50.000000 510.000000 250.000000 331.000000**

-----------------------------------------------------------------------

*Removing NaN values*

alpha beta gamma delta epsilon zeta eta

0 2 48 12 240 3.0 Yes AZ

2 4 44 24 220 7.0 No PA

3 5 42 30 210 9.0 Yes OK

4 6 44 36 220 11.0 Yes MD

5 7 46 42 230 13.0 No OH

*NaN values are the missing values (in technical terms null values) in the dataframe*

*They should not be confused with the 0 (zero), as zero is a value in the dataframe.*

*From the graph we can depict that values for Delta particles are decreasing and Gamma particles are increasing.*

**PS: - Please refer to the explanation part for the graph and the code with comments and to get detailed information.**

**Explanation:**

Understanding question: -

* Shape and first 5 columns of the data-frame.
* Data-types of the data-frame.
* Statistics values of the data-frame.
* Removing NaN values from the data-frame.
* Plotting graph using Matplotlib.

Pre-Knowledge: -

* Pandas
* Numpy
* Matplotlib

**Approach=>**

* *.shape()* method is used to find the data-frame number of rows and columns.
* *.dtypes* method to find the data-types of the column.
* *.describe()* method is used to find stats of the data-frame.
* *.dropna()* method is used to drop all the NaN values in the data-frame.
* *plt* function is used to plot graphs.
* *plt.title, plt.xlabel & plt.ylabel* is used to give title, x-axis and y-axis name.

***Code: - (with comments)***

#importing all the essential libraries

import pandas as pd #to read the csv file

import numpy as np #although this is optional for this question

import matplotlib.pyplot as plt #to draw the grap

#using pandas to read the csv file and storing in the dataframe df

'''Although it is not nesscery to pass sep (seperator) argument as delimitor'''

df = pd.read\_csv("messy\_data.csv", sep = ",") #in double quotes please keep the file path

print(df) # to print the entire dataframe

print("------------------------------------------------------------")

#printing the first five rows of the dataframe=>dataframe.head()=> pass the number by default it is 5

print(df.head())

print("------------------------------------------------------------")

print("DF info\n")

# dataframe=>size returns number of rows X columns

size = df.size

# dataframe=>shape returns the tuple of shape in (rows, columns)

shape = df.shape

# dataframe=>ndim returns the dimension 1 for 1D and 2 for 2D

dimension = df.ndim

#printing the size, shape and dimension of the dataframe

print("Size =",size,"\nShape =",shape,"\nDimension =",dimension)

print("------------------------------------------------------------")

print("Columns info\n")

#dataframe=>dtype returns the datatype of the dataframe

print(df.dtypes)

print("XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX")

#print all the information about the column use .info()

print(df.info())

print("------------------------------------------------------------")

print("Stats\n")

#printing out the statistics of dataframe=>dataframe.describe()

print(df.describe())

'''NaN values are the missing values (in technical terms null values) in the dataframe

They should not be confused with the 0 (zero), as zero is a value in the dataframe.

For example consider this dataframe:-

|col1|

------

|23 |

------

|24 |

------

|0 | => zero value

------

|NaN | => null value

------

'''

#to remove all the NaN values from the dataframe =>.dropna()

df = df.dropna()

print("------------------------------------------------------------")

print(df.head())

'''NOTE (by default, inplace is False

there is one parameter call inplace this can only take boolean values

True => if inplace = True then data is renamed as new data, basically replaces the dataframe after the operations

False => if inplace = False then data is displayed as copied dataframe, basically makes the new copy dataframe after the operations'''

# gca stands for 'get current axis'

ax = plt.gca() #it is used to print the graph on the same cartesian plane

#plot the scatter graph where x-axis is eplison (index) and y-axis are aplha,beta,gamma and delta

df.plot(kind='scatter',x='epsilon',y='alpha',color='red', ax=ax)

df.plot(kind='scatter',x='epsilon',y='beta', color='blue', ax=ax)

df.plot(kind='scatter',x='epsilon',y='gamma', color='green', ax=ax)

df.plot(kind='scatter',x='epsilon',y='delta', color='orange',ax=ax)

# Add title and axis names

plt.title('Data')

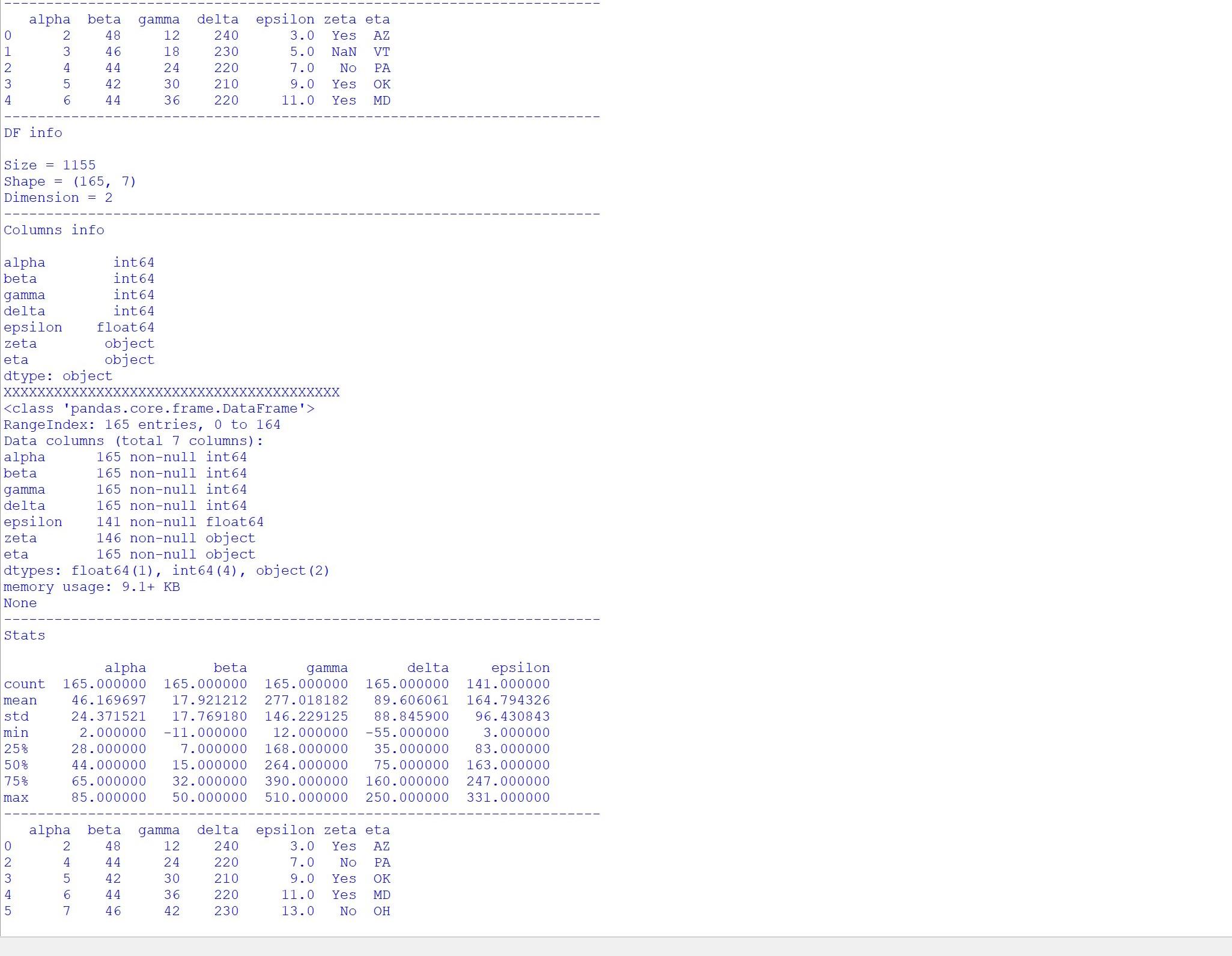
plt.xlabel('Epsilon')

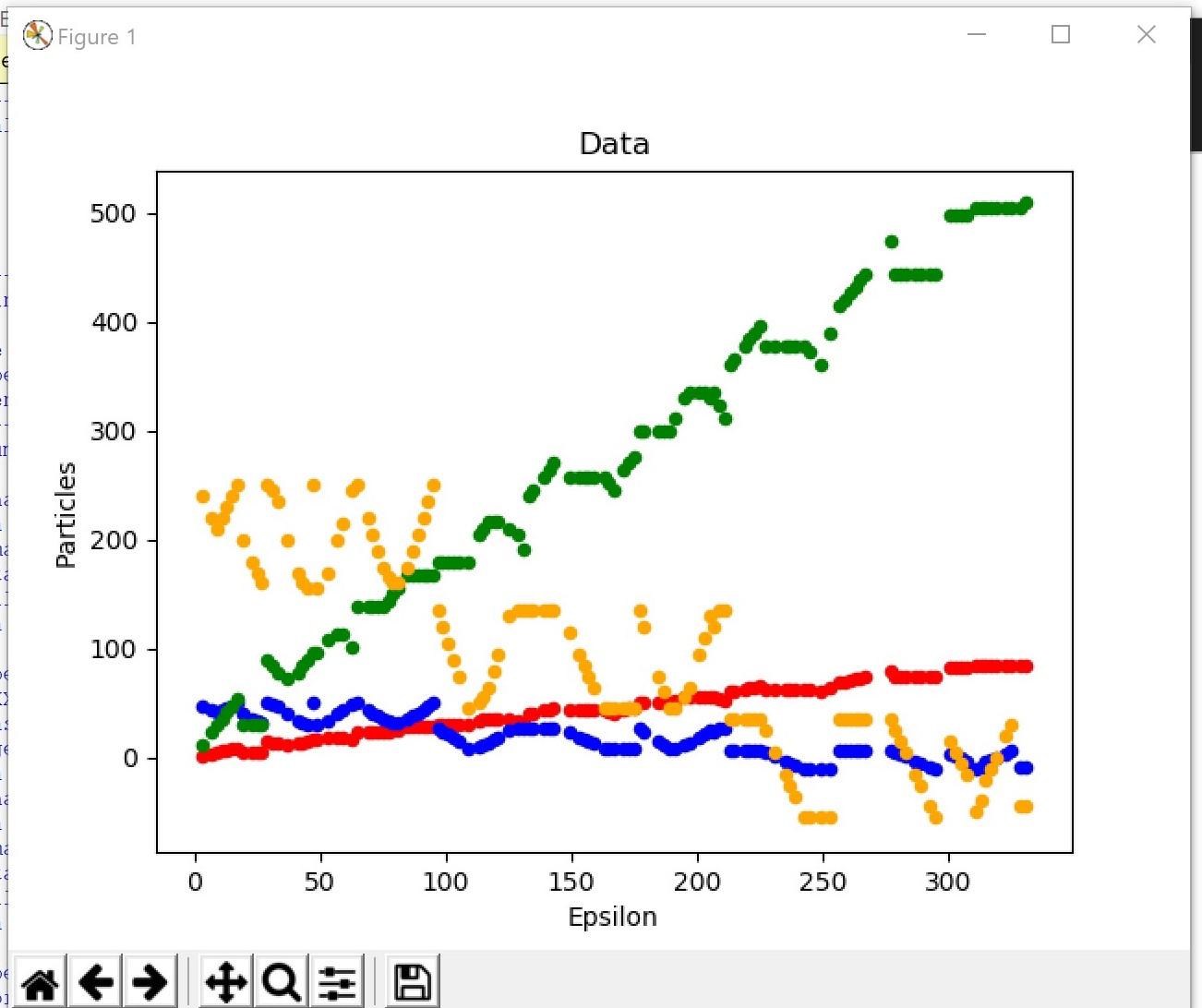
plt.ylabel('Particles')

#display the graph

plt.show()

***Output:-***

***Graph: -***

***﻿***

* SharedScreenshot.jpg
* SharedScreenshot0.jpg