Q1) Identify the Data type for the Following:

Activity	Data Type
Number of beatings from Wife	Discrete Data type
Results of rolling a dice	Discrete Data type
Weight of a person	Continuous Data type
Weight of Gold	Continuous Data type
Distance between two places	Continuous Data type
Length of a leaf	Continuous Data type
Dog's weight	Continuous Data type
Blue Color	Discrete Data type
Number of kids	Discrete Data type
Number of tickets in Indian railways	Discrete Data type
Number of times married	Discrete Data type
Gender (Male or Female)	Discrete Data type

Q2) Identify the Data types, which were among the following

Nominal, Ordinal, Interval, Ratio.

Data	Data Type
Gender	Discrete Data type - Nominal
High School Class Ranking	Discrete Data type - Nominal
Celsius Temperature	Continuous Data type - Interval
Weight	Continuous Data type - Ratio
Hair Color	Discrete Data type - Ratio
Socioeconomic Status	Continuous Data type - Interval
Fahrenheit Temperature	Continuous Data type - Ratio
Height	Continuous Data type - Ratio
Type of living accommodation	Discrete Data type - Ordinal
Level of Agreement	Discrete Data type - Interval
IQ(Intelligence Scale)	Discrete Data type - Interval
Sales Figures	Discrete Data type - Interval
Blood Group	Discrete Data type - Ratio
Time Of Day	Continuous Data type - Interval
Time on a Clock with Hands	Continuous Data type - Interval
Number of Children	Discrete Data type - Interval
Religious Preference	Discrete Data type - Ratio

Barometer Pressure	Continuous Data type - Interval
SAT Scores	Continuous Data type - Ratio
Years of Education	Discrete Data type - Nominal

Q3) Three Coins are tossed, find the probability that two heads and one tail are obtained?

Solution: -

Probability of 3 coins are tossing at a time. The possible outcomes are.

Head - H

Tail - T

{HHH, TTT, HHT, HTH, THH, TTH, THT, THH}

The Probability out comes are

1/8+1/8+1/8 = 3/8 or 0.375

Q4) Two Dice are rolled, find the probability that sum is

a) Equal to 1
b) Less than or equal to 4
c) Sum is divisible by 2 and 3

Solution: -

(**A)** The Probability = 0

Because 2 dice are rolled at a time we get (1,1), So the corresponding sum is not equal to 1.

i.e., 0/36 = 0

(**B**) The Probability out comes are (1,3) (2,2) (3,1) = 3

Outcomes is 3

Probability = 3/36 = 1/12

(**C**) The sum is divisible by 2 and 3 are 6, 12

The Possible ways of the 6 sum are (1,5), (2,4), (3,3), (4,2), (5,1).

Possible way for the 12 is (6,6).

The possible ways are 6.

The total possible outcomes are 36

Probability = Number outcomes/Total number of possible outcomes

6/36 = 1/6

Q5) A bag contains 2 red, 3 green and 2 blue balls. Two balls are drawn at random. What is the probability that none of the balls drawn is blue?
Solution: -

Total number of balls = (2 + 3 + 2) = 7

Let S be the sample space.

Then n(S) = Number of ways of drawing 2 balls out of 7 is

= $7\ C_2$

= (7 x 6)/(2 x 1)

= 21

Let E = Event drawing 2 balls,none of which is blue.

i.e., n(E) = Number of ways of drawing 2 balls out of (2+3)balls.

= $5\ C_2$

= (5 x 4)/(2 x 1)

= 10

P(E) = n(E) / n(S)

= 10 / 21

Q6) Calculate the Expected number of candies for a randomly selected child

Below are the probabilities of count of candies for children (ignoring the nature of the child-Generalized view)

CHILD	Candies count	Probability
A	1	0.015
B	4	0.20
C	3	0.65
D	5	0.005

E	6	0.01
F	2	0.120

Child A – probability of having 1 candy = 0.015.

Child B – probability of having 4 candies = 0.20

Solution: -

Child A - Probability of having 1 candy = 0.015.

Child B – probability of having 4 candies = 0.20

The Expected number of candies for randomly selected child are

1*0.015 + 4*0.20 + 3*0.65 + 5*0.005 + 6*0.01 + 2*0.12

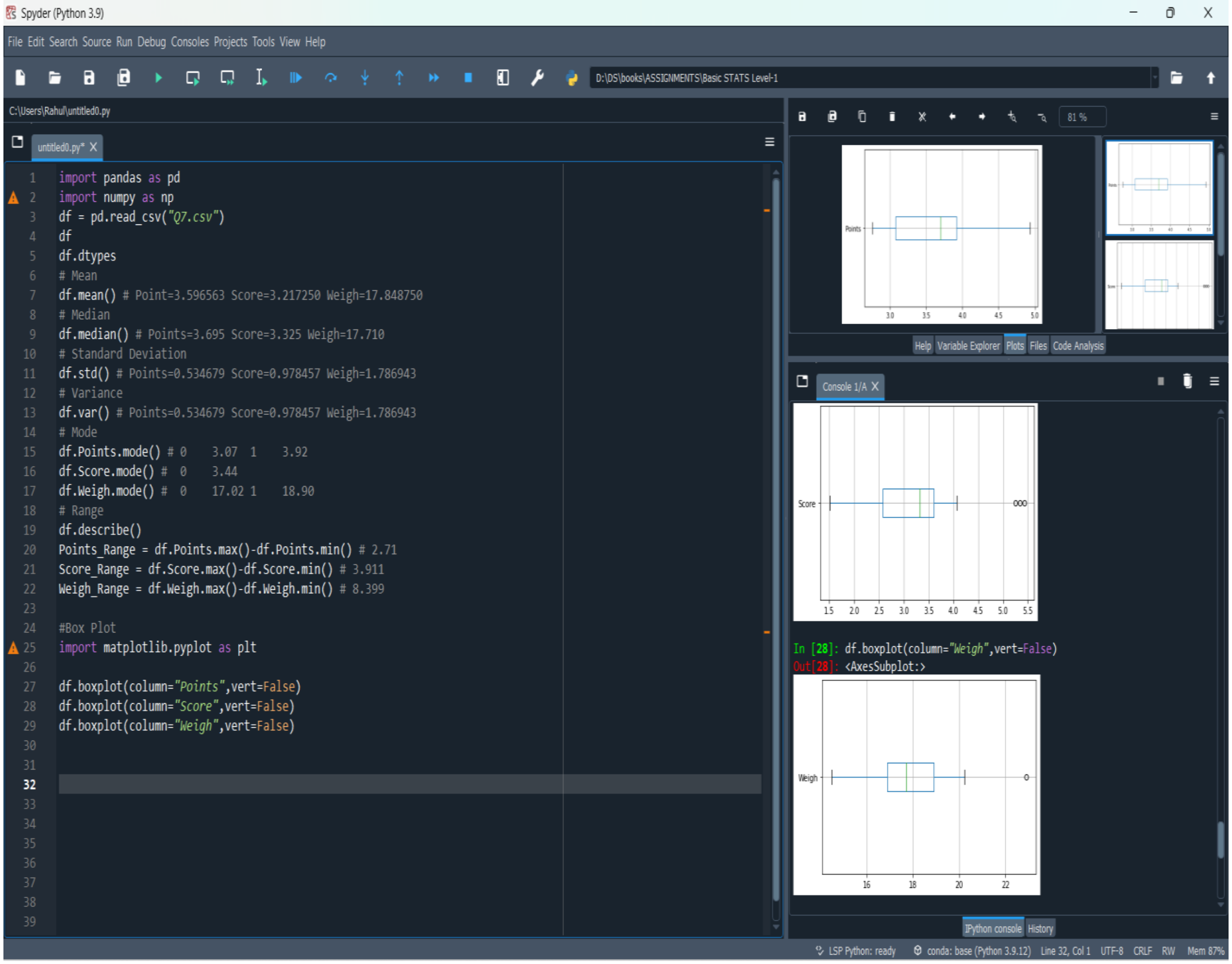
Q7) Calculate Mean, Median, Mode, Variance, Standard Deviation, Range & comment about the values / draw inferences, for the given dataset

- For Points, Score, Weigh >
  Find Mean, Median, Mode, Variance, Standard Deviation, and Range and also Comment about the values/ Draw some inferences.

All the Mean, Median, Mode, Variance, Standard Deviation, and Range are calculated

Solution: -

Points: Mean = 3.596563, Median = 3.695, Mode = “numeric”, Variance = 0.2858814, Standard deviation = 0.5346787.
Score: Mean = 3.21725, Median = 3.325, Mode = “numeric”, Variance = 0.957379, Standard deviation= 0.9784574
Note: Mean value are closer for both ‘Point’ and ‘Score’.
Weight: Mean = 17.84875, Median = 17.71, Mode = “numeric”, Variance = 3.193166, Standard deviation = 1.786943


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\DS\books\ASSIGNMENTS\Basic STATS Level-1
C:\Users\Rahul\untitled0.py
untitled0.py*
81 %
Points
Score
Weigh
Help Variable Explorer Plots Files Code Analysis
Console 1/A
IPython console History
LSP Python: ready conda: base (Python 3.9.12) Line 32, Col 1 UTF-8 CRLF RW Mem 87%

```python
import pandas as pd
import numpy as np
df = pd.read_csv("Q7.csv")
df
df.dtypes
# Mean
df.mean() # Point=3.596563 Score=3.217250 Weigh=17.848750
# Median
df.median() # Points=3.695 Score=3.325 Weigh=17.710
# Standard Deviation
df.std() # Points=0.534679 Score=0.978457 Weigh=1.786943
# Variance
df.var() # Points=0.534679 Score=0.978457 Weigh=1.786943
# Mode
df.Points.mode() # 0    3.07  1    3.92
df.Score.mode() #  0    3.44
df.Weigh.mode() #  0    17.02 1    18.90
# Range
df.describe()
Points_Range = df.Points.max()-df.Points.min() # 2.71
Score_Range = df.Score.max()-df.Score.min() # 3.911
Weigh_Range = df.Weigh.max()-df.Weigh.min() # 8.399

#Box Plot
import matplotlib.pyplot as plt

df.boxplot(column="Points",vert=False)
df.boxplot(column="Score",vert=False)
df.boxplot(column="Weigh",vert=False)
```

```
In [28]: df.boxplot(column="Weigh",vert=False)
Out[28]: <AxesSubplot:>
```

Calculation values are done in the Python and the values are in the code itself.

Q8) Calculate Expected Value for the problem below

a) The weights (X) of patients at a clinic (in pounds), are
108, 110, 123, 134, 135, 145, 167, 187, 199

Assume one of the patients is chosen at random. What is the Expected Value of the Weight of that patient?

Solution: -

$$\sum[\mathbf{x} \cdot \mathbf{p}(\mathbf{x})]$$

The Probability of patients = 1/9

X = 108, 110, 123, 134, 135, 145, 167, 187, 199

Then

Expected Value = 1/9(108+110+123+134+135+145+167+187+199)

=1/9(1308)

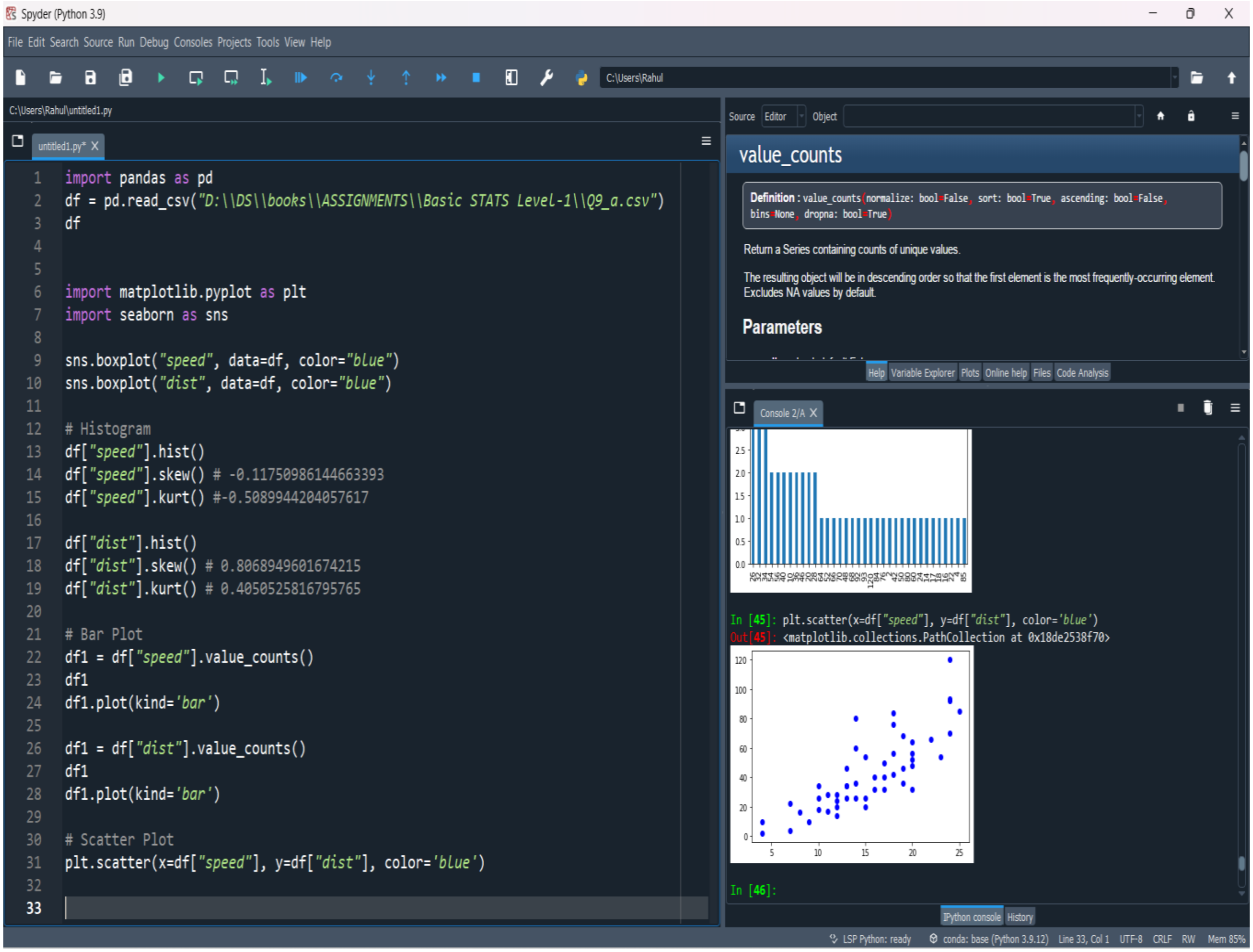
= 145.33ur

The Expected value of Weight of the Patient is **145.33**

Q9) Calculate Skewness, Kurtosis & draw inferences on the following data

Cars speed and distance

Use Q9_a.csv


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Rahul
C:\Users\Rahul\untitled1.py
untitled1.py*
Source Editor Object
Help Variable Explorer Plots Online help Files Code Analysis
Console 2/A
IPython console History
LSP Python: ready conda: base (Python 3.9.12) Line 33, Col 1 UTF-8 CRLF RW Mem 85%

```
import pandas as pd
df = pd.read_csv("D:\\DS\\books\\ASSIGNMENTS\\Basic STATS Level-1\\Q9_a.csv")
df


import matplotlib.pyplot as plt
import seaborn as sns

sns.boxplot("speed", data=df, color="blue")
sns.boxplot("dist", data=df, color="blue")

# Histogram
df["speed"].hist()
df["speed"].skew() # -0.11750986144663393
df["speed"].kurt() #-0.5089944204057617

df["dist"].hist()
df["dist"].skew() # 0.8068949601674215
df["dist"].kurt() # 0.4050525816795765

# Bar Plot
df1 = df["speed"].value_counts()
df1
df1.plot(kind='bar')

df1 = df["dist"].value_counts()
df1
df1.plot(kind='bar')

# Scatter Plot
plt.scatter(x=df["speed"], y=df["dist"], color='blue')

```

value_counts

Definition : value_counts(normalize: bool=False, sort: bool=True, ascending: bool=False, bins=None, dropna: bool=True)

Return a Series containing counts of unique values.

The resulting object will be in descending order so that the first element is the most frequently-occurring element. Excludes NA values by default.

Parameters

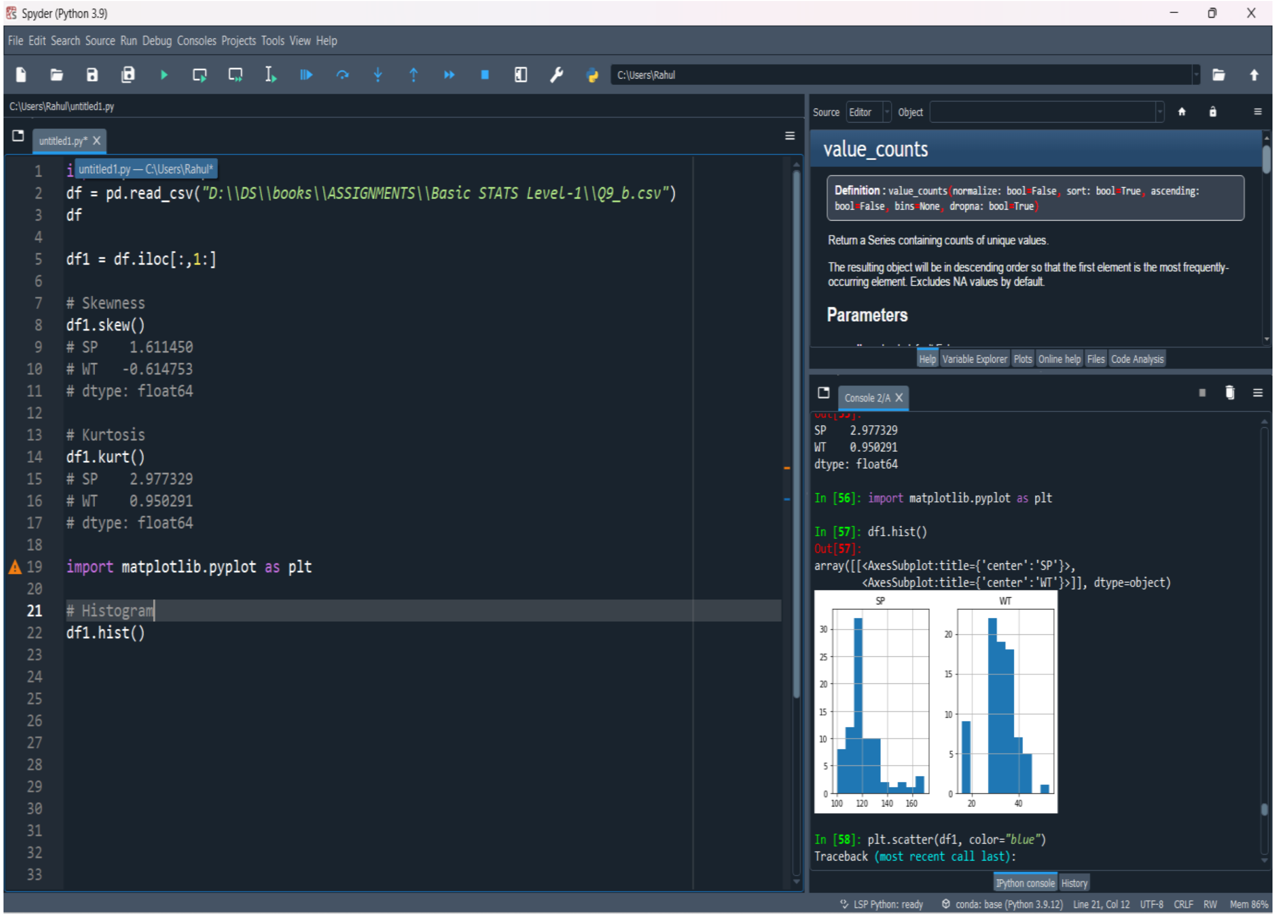
```
In [45]: plt.scatter(x=df["speed"], y=df["dist"], color='blue')
Out[45]: <matplotlib.collections.PathCollection at 0x18de2538f70>

In [46]:
```

Speed Skewness = -0.1175, Speed Kurtosis = -0.50899

Distance Skewness = 0.8068, Distance Kurtosis = 0.4050

SP and Weight(WT)

Use Q9_b.csv



Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Rahul
C:\Users\Rahul\untitled1.py
untitled1.py*
untitled1.py — C:\Users\Rahul*
Source Editor Object
Help Variable Explorer Plots Online help Files Code Analysis
Console 2/A
IPython console History
LSP Python: ready conda: base (Python 3.9.12) Line 21, Col 12 UTF-8 CRLF RW Mem 86%
SP
WT
0 5 10 15 20 25 30
100 120 140 160
0 5 10 15 20
20 40

```
i
df = pd.read_csv("D:\\DS\\books\\ASSIGNMENTS\\Basic STATS Level-1\\Q9_b.csv")
df

df1 = df.iloc[:,1:]

# Skewness
df1.skew()
# SP    1.611450
# WT   -0.614753
# dtype: float64

# Kurtosis
df1.kurt()
# SP    2.977329
# WT    0.950291
# dtype: float64

import matplotlib.pyplot as plt

# Histogram
df1.hist()
```

value_counts

Definition : value_counts(normalize: bool=False, sort: bool=True, ascending: bool=False, bins=None, dropna: bool=True)

Return a Series containing counts of unique values.

The resulting object will be in descending order so that the first element is the most frequently-occurring element. Excludes NA values by default.

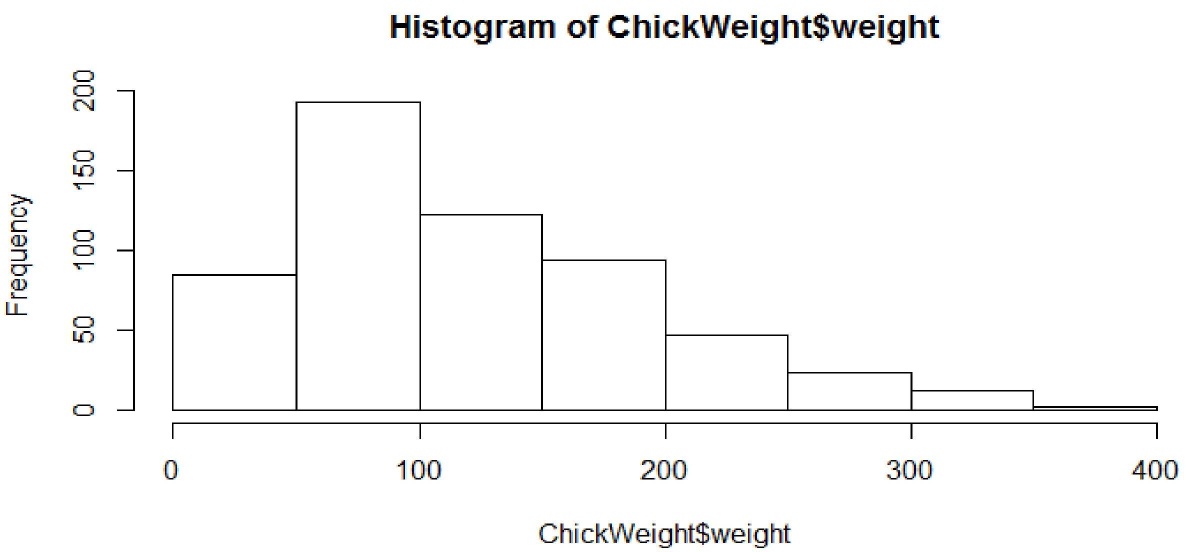
Parameters

```
SP    2.977329
WT    0.950291
dtype: float64

In [56]: import matplotlib.pyplot as plt

In [57]: df1.hist()
Out[57]:
array([[<AxesSubplot:title={'center':'SP'}>,
        <AxesSubplot:title={'center':'WT'}>]], dtype=object)

In [58]: plt.scatter(df1, color="blue")
Traceback (most recent call last):
```

SP Skewness = 1.6114, SP Kurtosis = 2.9773

WT Skewness = -0.6147, WT Kurtosis = 0.9502

Q10) Draw inferences about the following boxplot & histogram

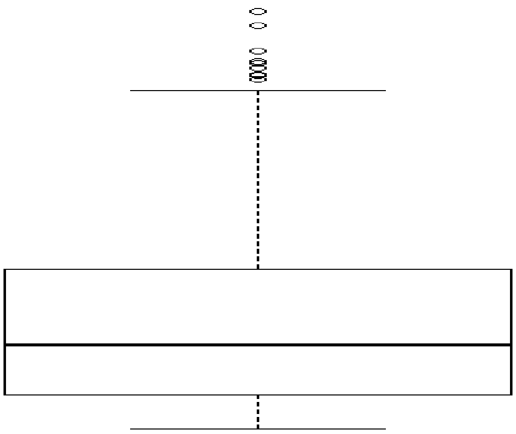


Sol :- The most of the data points are concerated in the range 50 – 100 with high frequency of 200.

The expected value the above distributon is 75.

The least range of weight is 400 somewere around 0-10.

Skewness – Noticed a long tail towards right so it is heavily right skewed.



Sol:- Medican is less than mean right skewed and we have outlier on the upper side of the box plot and there is less data points between Q1 and bottom point.

Q11) Suppose we want to estimate the average weight of an adult male in Mexico. We draw a random sample of 2,000 men from a population of 3,000,000 men and weigh them. We find that the average person in our sample weighs 200 pounds, and the standard deviation of the sample is 30 pounds. Calculate 94%,98%,96% confidence interval?

Sol:- The Mean X= 200

Standard Deviation **s**= 30

No. of samples n= 2000

The degree of freedom = 200-1 = 199

Considering a 94% confidence level, using a calculator, with 200 - 1 = 199 df, the critical value is t = 1.8916, hence

The Interval = $x \pm t\frac{s}{\sqrt{n}}$

$200 - 1.8916\frac{30}{\sqrt{2000}}$ = 198.73

$$200 + 1.8916\frac{30}{\sqrt{2000}} = 201.27$$

The 94% confidence interval is (198.73, 201.27)

Considering a 96% confidence level, using a calculator, with 200 - 1 = 199 df, the critical value is t = 2.0673, hence

$$200 - 2.0673\frac{30}{\sqrt{2000}} = 198.61$$

$$200 + 2.0673\frac{30}{\sqrt{2000}} = 201.39$$

The 96% confidence interval is (198.61, 201.39)

Considering a 98% confidence level, using a calculator, with 200 - 1 = 199 df, the critical value is t = 2.3452, hence

$$200 - 2.3452\frac{30}{\sqrt{2000}} = 198.43$$

$$200 + 2.3452\frac{30}{\sqrt{2000}} = 201.57$$

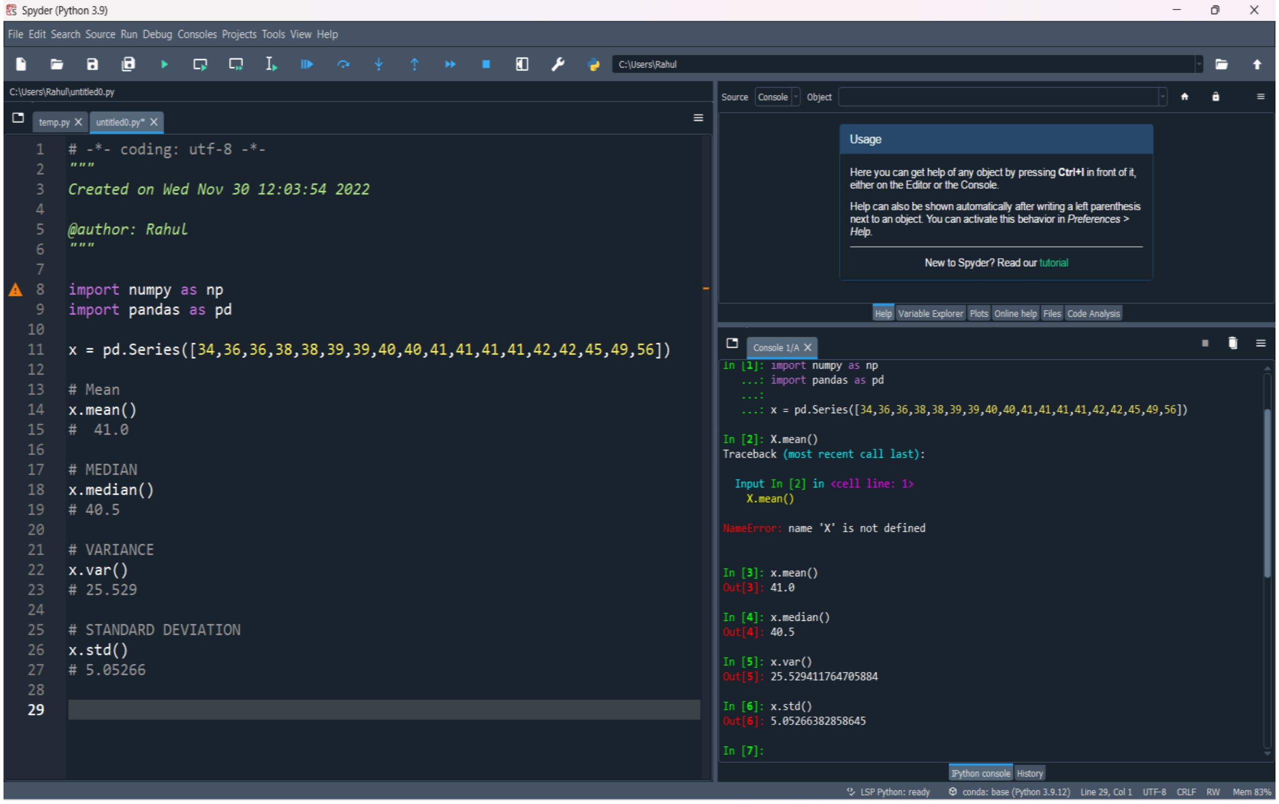
The 98% confidence interval is (198.43, 201.57)

Q12) Below are the scores obtained by a student in tests

34,36,36,38,38,39,39,40,40,41,41,41,41,42,42,45,49,56

1) Find mean, median, variance, standard deviation.
2) What can we say about the student marks?

Sol:- (1)


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Rahul
C:\Users\Rahul\untitled0.py
temp.py
untitled0.py*
Source Console Object
Help Variable Explorer Plots Online help Files Code Analysis
Console 1/A
IPython console History
LSP Python: ready conda: base (Python 3.9.12) Line 29, Col 1 UTF-8 CRLF RW Mem 83%

```python
# -*- coding: utf-8 -*-
"""
Created on Wed Nov 30 12:03:54 2022

@author: Rahul
"""

import numpy as np
import pandas as pd

x = pd.Series([34,36,36,38,38,39,39,40,40,41,41,41,41,42,42,45,49,56])

# Mean
x.mean()
#  41.0

# MEDIAN
x.median()
# 40.5

# VARIANCE
x.var()
# 25.529

# STANDARD DEVIATION
x.std()
# 5.05266

```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
In [1]: import numpy as np
   ...: import pandas as pd
   ...:
   ...: x = pd.Series([34,36,36,38,38,39,39,40,40,41,41,41,41,42,42,45,49,56])

In [2]: X.mean()
Traceback (most recent call last):

  Input In [2] in <cell line: 1>
    X.mean()

NameError: name 'X' is not defined


In [3]: x.mean()
Out[3]: 41.0

In [4]: x.median()
Out[4]: 40.5

In [5]: x.var()
Out[5]: 25.529411764705884

In [6]: x.std()
Out[6]: 5.05266382858645

In [7]:
```

(2)

Students get the average marts is 41, minimum marks are 34 and Maximum marks are 56.

Q13) What is the nature of skewness when mean, median of data are equal?

Sol:- if the nature skewness mean and median is equal then it is a “Symmetrical”.

Q14) What is the nature of skewness when mean > median?

Sol:- The nature of skewness when mean > median then it is a “Right Skewed”.

Q15) What is the nature of skewness when median > mean?

Sol:- The nature of skewness when median > mean then “Left Skewed”.

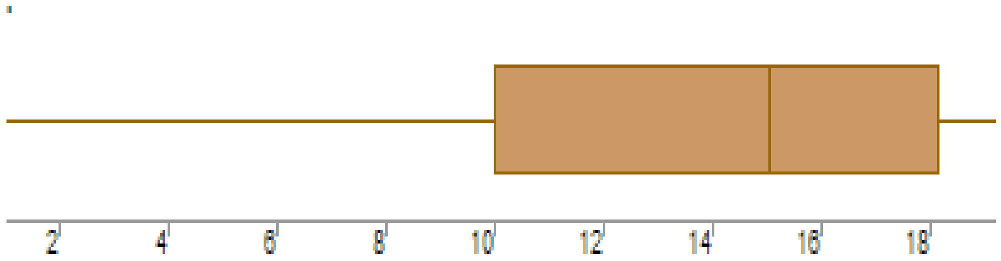
Q16) What does positive kurtosis value indicates for a data?

Sol:- The data is normally distributed and kurtosis value is 0.

Q17) What does negative kurtosis value indicates for a data?

Sol:- The distribution of the data has lighter tails and a flatter peaks than the normal distribution.

Q18) Answer the below questions using the below boxplot visualization.



What can we say about the distribution of the data?

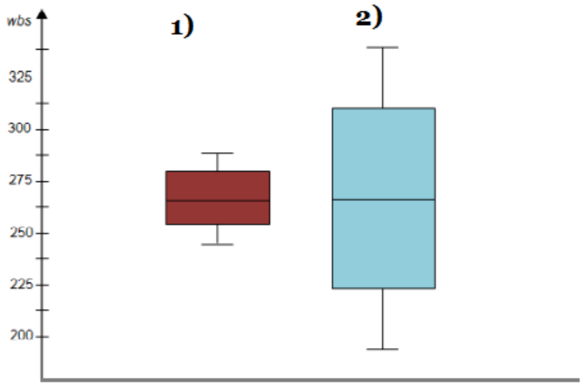
Sol:- Let’s assume above box plot is about age’s of the students in a school. 50% of the people are above 10 yrs old and remaining are less. And students who’s age is above 15 are approx 40%.

What is nature of skewness of the data?

Sol:- The Nature of skewness is Left Skewed, median is greater then mean.

What will be the IQR of the data (approximately)?
Sol:- The Approximately the value is -8

Q19) Comment on the below Boxplot visualizations?



Draw an Inference from the distribution of data for Boxplot 1 with respect Boxplot 2.

Sol:- By observing the above the Boxplot from the both the plots whisker's level is high in boxplot 2, mean and median are equal hence the distribution is Symmetrical.

Q 20) Calculate probability from the given dataset for the below cases

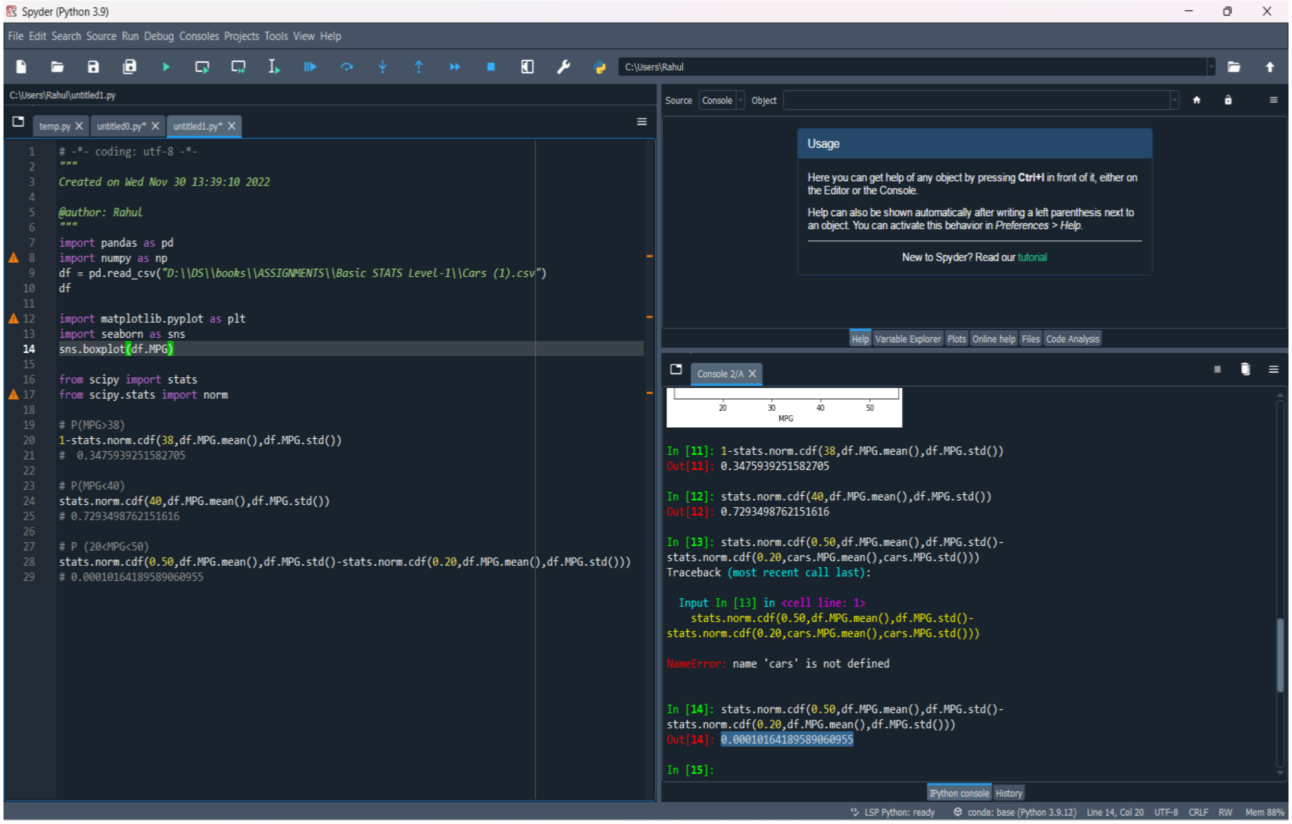
Data _set: Cars.csv

Calculate the probability of MPG of Cars for the below cases.

MPG <- Car $MPG

a. P(MPG>38)
b. P(MPG<40)
c. P (20<MPG<50)

Sol:-


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Rahul
C:\Users\Rahul\untitled1.py
temp.py untitled0.py* untitled1.py*
Source Console Object
Help Variable Explorer Plots Online help Files Code Analysis
Console 2/A
20 30 40 50
MPG
IPython console History
LSP Python: ready conda: base (Python 3.9.12) Line 14, Col 20 UTF-8 CRLF RW Mem 88%

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

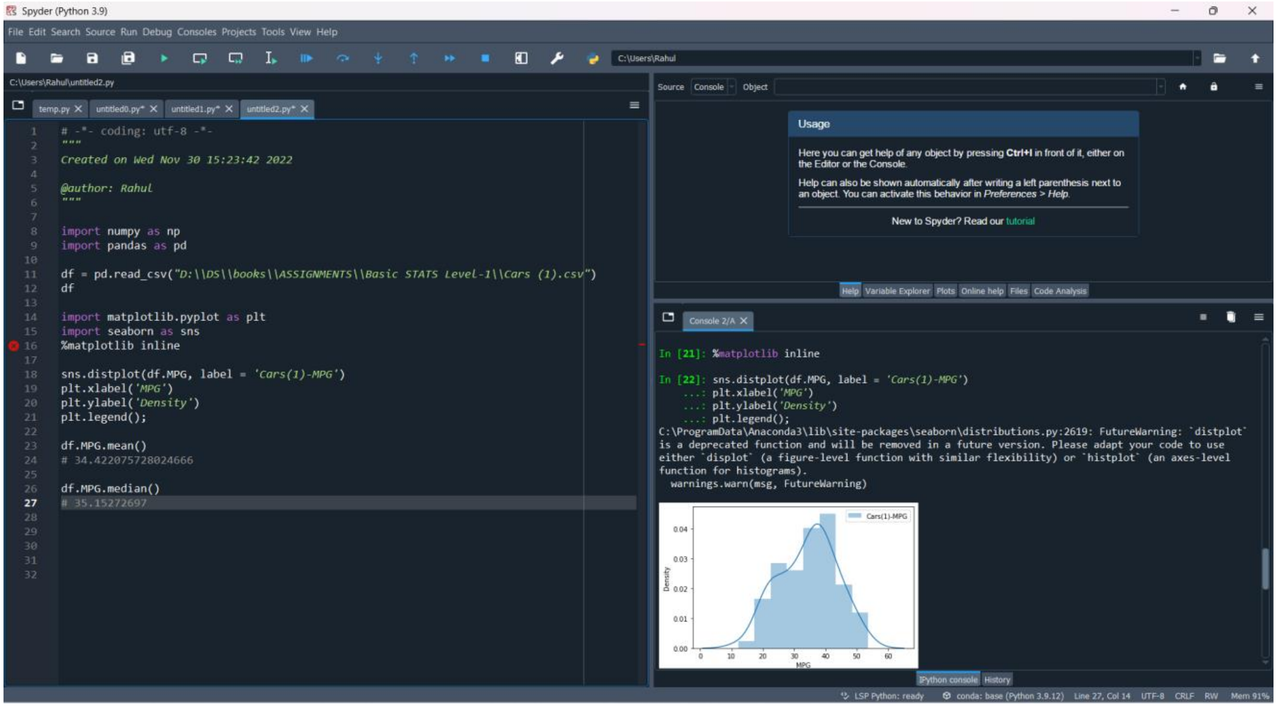
```python
# -*- coding: utf-8 -*-
"""
Created on Wed Nov 30 13:39:10 2022

@author: Rahul
"""
import pandas as pd
import numpy as np
df = pd.read_csv("D:\\DS\\books\\ASSIGNMENTS\\Basic STATS Level-1\\Cars (1).csv")
df

import matplotlib.pyplot as plt
import seaborn as sns
sns.boxplot(df.MPG)

from scipy import stats
from scipy.stats import norm

# P(MPG>38)
1-stats.norm.cdf(38,df.MPG.mean(),df.MPG.std())
#  0.3475939251582705

# P(MPG<40)
stats.norm.cdf(40,df.MPG.mean(),df.MPG.std())
# 0.7293498762151616

# P (20<MPG<50)
stats.norm.cdf(0.50,df.MPG.mean(),df.MPG.std()-stats.norm.cdf(0.20,df.MPG.mean(),df.MPG.std()))
# 0.00010164189589060955
```

```
In [11]: 1-stats.norm.cdf(38,df.MPG.mean(),df.MPG.std())
Out[11]: 0.3475939251582705

In [12]: stats.norm.cdf(40,df.MPG.mean(),df.MPG.std())
Out[12]: 0.7293498762151616

In [13]: stats.norm.cdf(0.50,df.MPG.mean(),df.MPG.std()-
stats.norm.cdf(0.20,cars.MPG.mean(),cars.MPG.std()))
Traceback (most recent call last):

  Input In [13] in <cell line: 1>
    stats.norm.cdf(0.50,df.MPG.mean(),df.MPG.std()-
stats.norm.cdf(0.20,cars.MPG.mean(),cars.MPG.std()))

NameError: name 'cars' is not defined


In [14]: stats.norm.cdf(0.50,df.MPG.mean(),df.MPG.std()-
stats.norm.cdf(0.20,df.MPG.mean(),df.MPG.std()))
Out[14]: 0.00010164189589060955

In [15]:
```

Q 21) Check whether the data follows normal distribution

a) Check whether the MPG of Cars follows Normal Distribution

Dataset: Cars.csv


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Rahul
C:\Users\Rahul\untitled2.py
temp.py untitled0.py* untitled1.py* untitled2.py*
Source Console Object
Help Variable Explorer Plots Online help Files Code Analysis
Console 2/A
Cars(1)-MPG
0.04
0.03
0.02
0.01
0.00
Density
0 10 20 30 40 50 60
MPG
IPython console History
LSP Python: ready conda: base (Python 3.9.12) Line 27, Col 14 UTF-8 CRLF RW Mem 91%

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

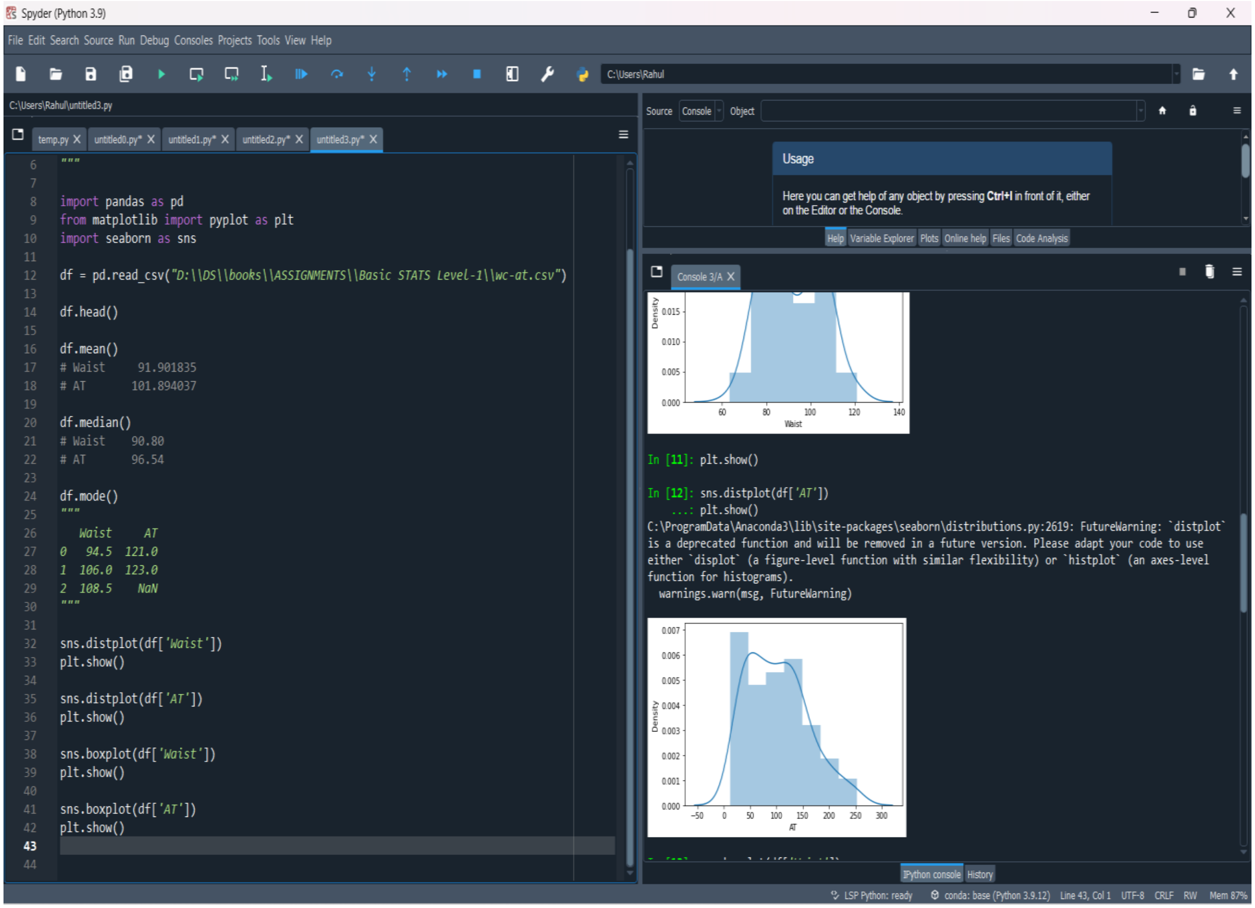
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```python
# -*- coding: utf-8 -*-
"""
Created on Wed Nov 30 15:23:42 2022

@author: Rahul
"""

import numpy as np
import pandas as pd

df = pd.read_csv("D:\\DS\\books\\ASSIGNMENTS\\Basic STATS Level-1\\Cars (1).csv")
df

import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

sns.distplot(df.MPG, label = 'Cars(1)-MPG')
plt.xlabel('MPG')
plt.ylabel('Density')
plt.legend();

df.MPG.mean()
# 34.422075728024666

df.MPG.median()
# 35.15272697
```

```
In [21]: %matplotlib inline

In [22]: sns.distplot(df.MPG, label = 'Cars(1)-MPG')
    ...: plt.xlabel('MPG')
    ...: plt.ylabel('Density')
    ...: plt.legend();
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot`
is a deprecated function and will be removed in a future version. Please adapt your code to use
either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level
function for histograms).
  warnings.warn(msg, FutureWarning)
```

b) Check Whether the Adipose Tissue (AT) and Waist Circumference(Waist) from wc-at data set follows Normal Distribution
 Dataset: wc-at.csv



```
"""


import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns

df = pd.read_csv("D:\\DS\\books\\ASSIGNMENTS\\Basic STATS Level-1\\wc-at.csv")

df.head()

df.mean()
# Waist     91.901835
# AT       101.894037

df.median()
# Waist    90.80
# AT       96.54

df.mode()
"""
   Waist     AT
0   94.5  121.0
1  106.0  123.0
2  108.5    NaN
"""

sns.distplot(df['Waist'])
plt.show()

sns.distplot(df['AT'])
plt.show()

sns.boxplot(df['Waist'])
plt.show()

sns.boxplot(df['AT'])
plt.show()
```

```
In [11]: plt.show()

In [12]: sns.distplot(df['AT'])
    ...: plt.show()
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot`
is a deprecated function and will be removed in a future version. Please adapt your code to use
either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level
function for histograms).
  warnings.warn(msg, FutureWarning)
```

Q 22) Calculate the Z scores of 90% confidence interval,94% confidence interval, 60% confidence interval

Sol:-

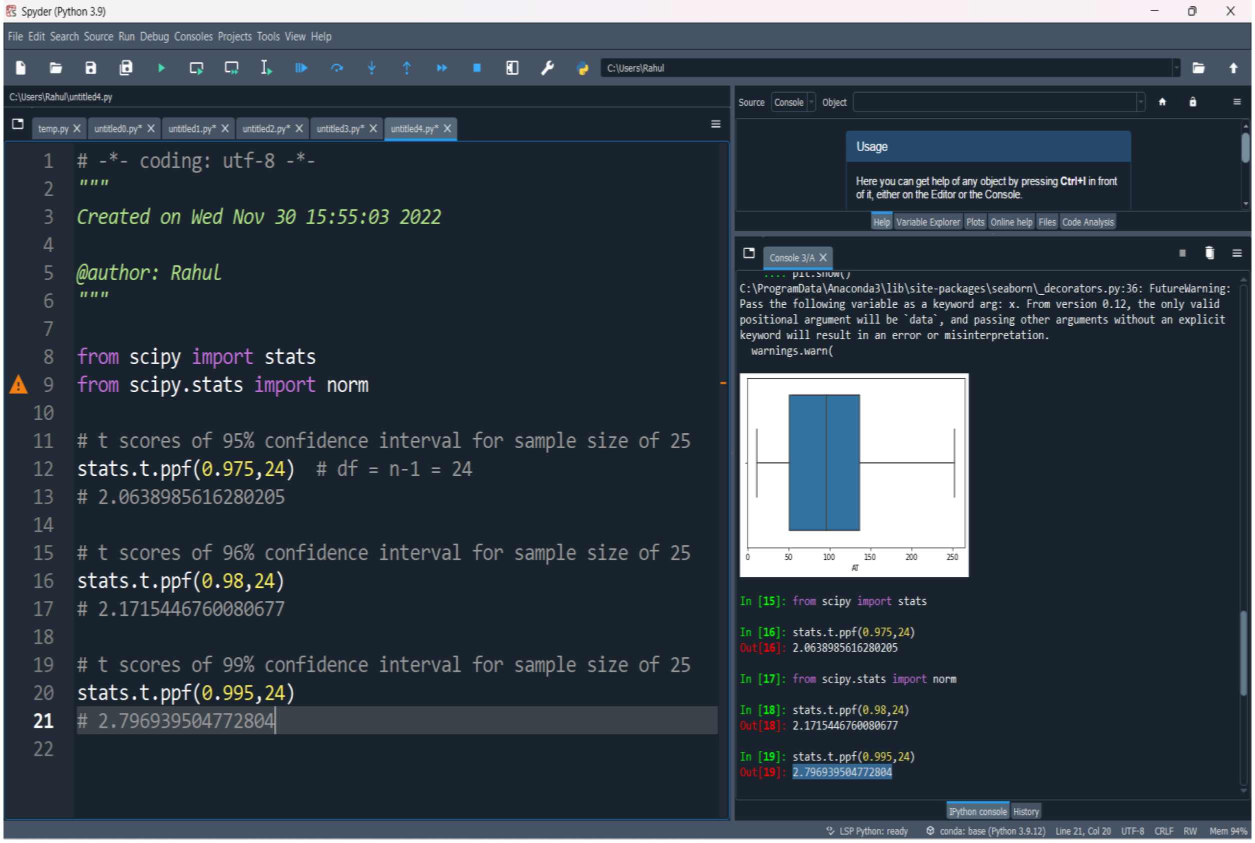
Z score of 90% confidence interval is 1.65

Z score of 94% confidence interval is 1.55

Z score of 60% confidence interval is 0.85

Q 23) Calculate the t scores of 95% confidence interval, 96% confidence interval, 99% confidence interval for sample size of 25

Sol:-



```
# -*- coding: utf-8 -*-
"""
Created on Wed Nov 30 15:55:03 2022

@author: Rahul
"""

from scipy import stats
from scipy.stats import norm

# t scores of 95% confidence interval for sample size of 25
stats.t.ppf(0.975,24)  # df = n-1 = 24
# 2.0638985616280205

# t scores of 96% confidence interval for sample size of 25
stats.t.ppf(0.98,24)
# 2.1715446760080677

# t scores of 99% confidence interval for sample size of 25
stats.t.ppf(0.995,24)
# 2.796939504772804
```

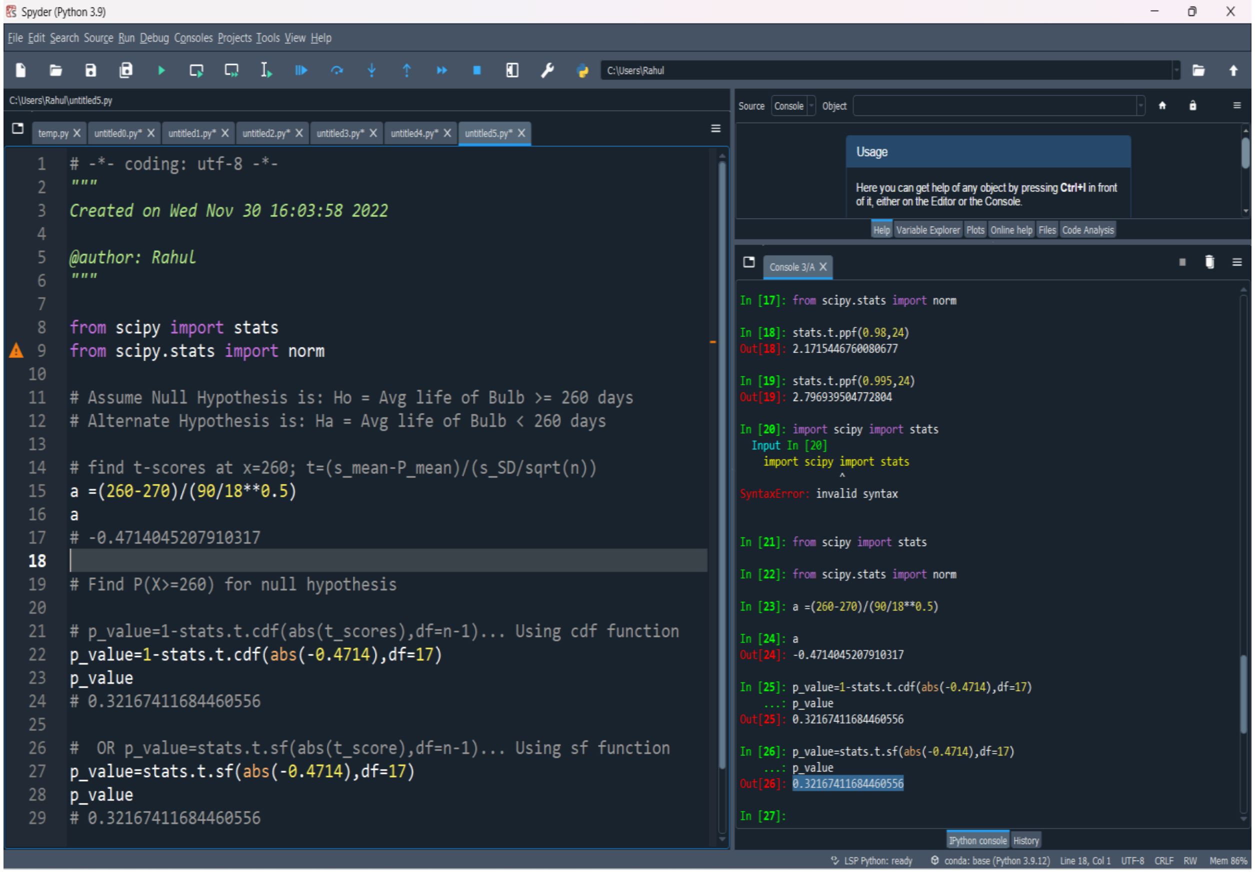
```
    ....: plt.show()
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit
keyword will result in an error or misinterpretation.
  warnings.warn(

In [15]: from scipy import stats

In [16]: stats.t.ppf(0.975,24)
Out[16]: 2.0638985616280205

In [17]: from scipy.stats import norm

In [18]: stats.t.ppf(0.98,24)
Out[18]: 2.1715446760080677

In [19]: stats.t.ppf(0.995,24)
Out[19]: 2.796939504772804
```

Q 24**)** A Government company claims that an average light bulb lasts 270 days. A researcher randomly selects 18 bulbs for testing. The sampled bulbs last an average of 260 days, with a standard deviation of 90 days. If the CEO's claim were true, what is the probability that 18 randomly selected bulbs would have an average life of no more than 260 days

Hint:

rcode → pt(tscore,df)

df → degrees of freedom



```python
# -*- coding: utf-8 -*-
"""
Created on Wed Nov 30 16:03:58 2022

@author: Rahul
"""

from scipy import stats
from scipy.stats import norm

# Assume Null Hypothesis is: Ho = Avg life of Bulb >= 260 days
# Alternate Hypothesis is: Ha = Avg life of Bulb < 260 days

# find t-scores at x=260; t=(s_mean-P_mean)/(s_SD/sqrt(n))
a =(260-270)/(90/18**0.5)
a
# -0.4714045207910317

# Find P(X>=260) for null hypothesis

# p_value=1-stats.t.cdf(abs(t_scores),df=n-1)... Using cdf function
p_value=1-stats.t.cdf(abs(-0.4714),df=17)
p_value
# 0.32167411684460556

#  OR p_value=stats.t.sf(abs(t_score),df=n-1)... Using sf function
p_value=stats.t.sf(abs(-0.4714),df=17)
p_value
# 0.32167411684460556
```

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

```
In [17]: from scipy.stats import norm

In [18]: stats.t.ppf(0.98,24)
Out[18]: 2.1715446760080677

In [19]: stats.t.ppf(0.995,24)
Out[19]: 2.796939504772804

In [20]: import scipy import stats
  Input In [20]
    import scipy import stats
                 ^
SyntaxError: invalid syntax


In [21]: from scipy import stats

In [22]: from scipy.stats import norm

In [23]: a =(260-270)/(90/18**0.5)

In [24]: a
Out[24]: -0.4714045207910317

In [25]: p_value=1-stats.t.cdf(abs(-0.4714),df=17)
    ...: p_value
Out[25]: 0.32167411684460556

In [26]: p_value=stats.t.sf(abs(-0.4714),df=17)
    ...: p_value
Out[26]: 0.32167411684460556

In [27]:
```