In [ ]:

```
Hypothesis Assignment
```

Q 1.

A F&B manager wants to determine whether there is any significant difference in the diameter of the cutlet between two units. A randomly selected sample of cutlets was collected from both units and measured? Analyze the data and draw inferences at 5% significance level. Please state the assumptions and tests that you carried out to check validity of the assumptions.

In [5]:

```
# import the libraries
%matplotlib inline
import numpy as np
import scipy as sp
import pandas as pd
import statsmodels.api as sm
import statsmodels.formula.api as smf
import matplotlib as mpl
import matplotlib.pyplot as plt
import seaborn as sns
from statsmodels.stats.proportion import proportions_ztest
```

In [7]:

```
Q1_data = pd.read_csv("C:/Users/Hp/Downloads/Cutlets.csv")
Q1_data.head()
```

Out[7]:

	Unit A	Unit B
0	6.8090	6.7703
1	6.4376	7.5093
2	6.9157	6.7300
3	7.3012	6.7878
4	7.4488	7.1522

In [8]:

```
Q1_data.describe(include='all')
```

Out[8]:

	Unit A	Unit B
count	35.000000	35.000000
mean	7.019091	6.964297
std	0.288408	0.343401
min	6.437600	6.038000
25%	6.831500	6.753600
50%	6.943800	6.939900
75%	7.280550	7.195000
max	7.516900	7.545900

In [11]:

```
Unit_A=Q1_data['Unit A'].mean()
Unit_B=Q1_data['Unit B'].mean()
print('Unit A Mean= ',Unit_A,'\nUnit B Mean = ',Unit_B)
print('Unit A Mean > Unite B Mean = ',Unit_A>Unit_B)
```

```
Unit A Mean=  7.01909142857143
Unit B Mean =  6.964297142857142
Unit A Mean > Unite B Mean =  True
```

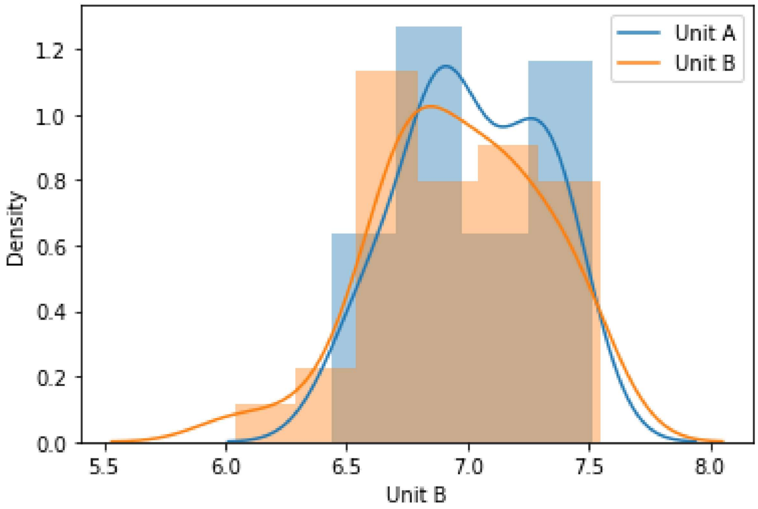
In [12]:

```
sns.distplot(Q1_data['Unit A'])
sns.distplot(Q1_data['Unit B'])
plt.legend(['Unit A','Unit B'])
```

```
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
```

Out[12]:

```
<matplotlib.legend.Legend at 0x1cd65416370>
```

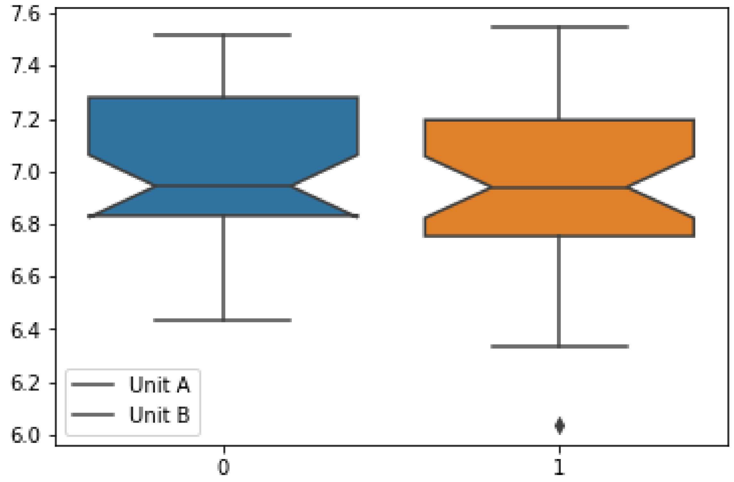


In [13]:

```
sns.boxplot(data=[Q1_data['Unit A'],Q1_data['Unit B']],notch=True)
plt.legend(['Unit A','Unit B'])
```

Out[13]:

```
<matplotlib.legend.Legend at 0x1cd65416fa0>
```



In [14]:

```
alpha=0.05
UnitA=pd.DataFrame(Q1_data['Unit A'])
UnitB=pd.DataFrame(Q1_data['Unit B'])
print(UnitA,UnitB)
```

```
    Unit A
0   6.8090
1   6.4376
2   6.9157
3   7.3012
4   7.4488
5   7.3871
6   6.8755
7   7.0621
8   6.6840
9   6.8236
10  7.3930
11  7.5169
12  6.9246
13  6.9256
14  6.5797
15  6.8394
16  6.5970
17  7.2705
18  7.2828
19  7.3495
20  6.9438
21  7.1560
22  6.5341
23  7.2854
24  6.9952
25  6.8568
26  7.2163
27  6.6801
28  6.9431
29  7.0852
30  6.7794
31  7.2783
32  7.1561
33  7.3943
34  6.9405      Unit B
0   6.7703
1   7.5093
2   6.7300
3   6.7878
4   7.1522
5   6.8110
6   7.2212
7   6.6606
8   7.2402
9   7.0503
10  6.8810
11  7.4059
12  6.7652
13  6.0380
14  7.1581
15  7.0240
16  6.6672
```

```
17   7.4314
18   7.3070
19   6.7478
20   6.8889
21   7.4220
22   6.5217
23   7.1688
24   6.7594
25   6.9399
26   7.0133
27   6.9182
28   6.3346
29   7.5459
30   7.0992
31   7.1180
32   6.6965
33   6.5780
34   7.3875
```

In [15]:

```
tStat,pValue =sp.stats.ttest_ind(UnitA,UnitB)
print("P-Value:{0} T-Statistic:{1}".format(pValue,tStat))
```

```
P-Value:[0.47223947] T-Statistic:[0.72286887]
```

In [16]:

```
if pValue <0.05:
  print('we reject null hypothesis')
else:
  print('we accept null hypothesis')
```

```
we accept null hypothesis
```

Inference is that there is no significant difference in the diameters of Unit A and Unit B

In [ ]:

```
Q2
```

In [ ]:

```
A hospital wants to determine whether there is any
difference in the average Turn Around Time (TAT) of
reports of the laboratories on their preferred list.
They collected a random sample and recorded TAT for
reports of 4 laboratories.
TAT is defined as sample collected to report dispatch.
Analyze the data and determine whether there is any
difference in average TAT among the different laboratories
at 5% significance level.
```

In [19]:

```
LabTAT =pd.read_csv('C:/Users/Hp/Downloads/LabTAT.csv')
LabTAT.head()
```

Out[19]:

	Laboratory 1	Laboratory 2	Laboratory 3	Laboratory 4
0	185.35	165.53	176.70	166.13
1	170.49	185.91	198.45	160.79
2	192.77	194.92	201.23	185.18
3	177.33	183.00	199.61	176.42
4	193.41	169.57	204.63	152.60

In [20]:

```
LabTAT.describe()
```

Out[20]:

	Laboratory 1	Laboratory 2	Laboratory 3	Laboratory 4
count	120.000000	120.000000	120.000000	120.00000
mean	178.361583	178.902917	199.913250	163.68275
std	13.173594	14.957114	16.539033	15.08508
min	138.300000	140.550000	159.690000	124.06000
25%	170.335000	168.025000	188.232500	154.05000
50%	178.530000	178.870000	199.805000	164.42500
75%	186.535000	189.112500	211.332500	172.88250
max	216.390000	217.860000	238.700000	205.18000

In [21]:

```
Laboratory_1=LabTAT['Laboratory 1'].mean()
Laboratory_2=LabTAT['Laboratory 2'].mean()
Laboratory_3=LabTAT['Laboratory 3'].mean()
Laboratory_4=LabTAT['Laboratory 4'].mean()

print('Laboratory 1 Mean = ',Laboratory_1)
print('Laboratory 2 Mean = ',Laboratory_2)
print('Laboratory 3 Mean = ',Laboratory_3)
print('Laboratory 4 Mean = ',Laboratory_4)
```

```
Laboratory 1 Mean =  178.36158333333339
Laboratory 2 Mean =  178.9029166666668
Laboratory 3 Mean =  199.91325000000003
Laboratory 4 Mean =  163.6827499999999
```

In [23]:

```
print('Laboratory_1 > Laboratory_2 = ',Laboratory_1 > Laboratory_2)
print('Laboratory_2 > Laboratory_3 = ',Laboratory_2 > Laboratory_3)
print('Laboratory_3 > Laboratory_4 = ',Laboratory_3 > Laboratory_4)
print('Laboratory_4 > Laboratory_1 = ',Laboratory_4 > Laboratory_1)
```

```
Laboratory_1 > Laboratory_2 =  False
Laboratory_2 > Laboratory_3 =  False
Laboratory_3 > Laboratory_4 =  True
Laboratory_4 > Laboratory_1 =  False
```

In [ ]:

```
The Null and Alternative Hypothesis

There are no significant differences between the groups'
mean Lab values. H0:μ1=μ2=μ3=μ4=μ5

There is a significant difference between the groups' mean Lab
values. Ha:μ1≠μ2≠μ3≠μ4
```

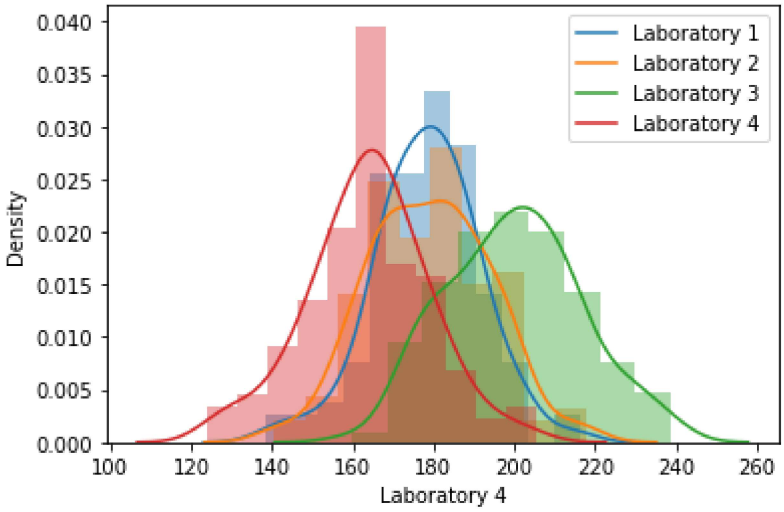
In [24]:

```
sns.distplot(LabTAT['Laboratory 1'])
sns.distplot(LabTAT['Laboratory 2'])
sns.distplot(LabTAT['Laboratory 3'])
sns.distplot(LabTAT['Laboratory 4'])
plt.legend(['Laboratory 1','Laboratory 2','Laboratory 3','Laboratory 4'])
```

```
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
```

Out[24]:

```
<matplotlib.legend.Legend at 0x1cd6bfe5760>
```

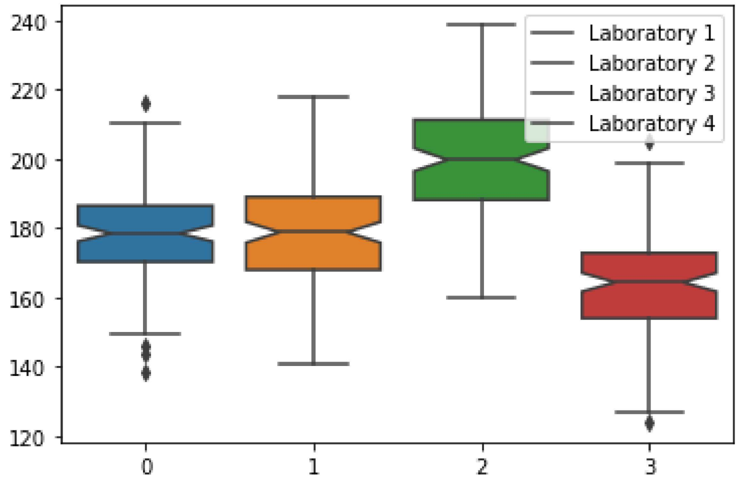


In [25]:

```
sns.boxplot(data=[LabTAT['Laboratory 1'],LabTAT['Laboratory 2'],LabTAT['Laboratory 3'],LabT
plt.legend(['Laboratory 1','Laboratory 2','Laboratory 3','Laboratory 4'])
```

Out[25]:

```
<matplotlib.legend.Legend at 0x1cd6c0d1a90>
```



In [26]:

```
alpha=0.05
Lab1=pd.DataFrame(LabTAT['Laboratory 1'])
Lab2=pd.DataFrame(LabTAT['Laboratory 2'])
Lab3=pd.DataFrame(LabTAT['Laboratory 3'])
Lab4=pd.DataFrame(LabTAT['Laboratory 4'])
print(Lab1,Lab1,Lab3,Lab4)
```

```
     Laboratory 1
0          185.35
1          170.49
2          192.77
3          177.33
4          193.41
..            ...
115        178.49
116        176.08
117        202.48
118        182.40
119        182.09

[120 rows x 1 columns]      Laboratory 1
0          185.35
1          170.49
2          192.77
3          177.33
4          193.41
..            ...
115        178.49
116        176.08
117        202.48
118        182.40
119        182.09

[120 rows x 1 columns]      Laboratory 3
0          176.70
1          198.45
2          201.23
3          199.61
4          204.63
..            ...
115        193.80
116        215.25
117        203.99
118        194.52
119        221.49

[120 rows x 1 columns]      Laboratory 4
0          166.13
1          160.79
2          185.18
3          176.42
4          152.60
..            ...
115        172.68
116        177.64
117        170.27
118        150.87
119        162.21
```

```
[120 rows x 1 columns]
```

In [27]:

```
tStat, pvalue = sp.stats.f_oneway(Lab1,Lab2,Lab3,Lab4)
print("P-Value:{0} T-Statistic:{1}".format(pValue,tStat))
```

```
P-Value:[0.47223947] T-Statistic:[118.70421654]
```

In [28]:

```
if pValue < 0.05:
  print('we reject null hypothesis')
else:
  print('we accept null hypothesis')
```

```
we accept null hypothesis
```

Inference is that there no significant difference in the average TAT for all the labs.

3. Sales of products in four different regions is tabulated for males and females. Find if male-female buyer rations are similar across regions

In [31]:

```
BuyerRatio =pd.read_csv('C:/Users/Hp/Downloads/BuyerRatio.csv')
BuyerRatio.head()
```

Out[31]:

	Observed Values	East	West	North	South
0	Males	50	142	131	70
1	Females	435	1523	1356	750

In [32]:

```
BuyerRatio.describe()
```

Out[32]:

	East	West	North	South
count	2.000000	2.000000	2.000000	2.000000
mean	242.500000	832.500000	743.500000	410.000000
std	272.236111	976.514465	866.205807	480.832611
min	50.000000	142.000000	131.000000	70.000000
25%	146.250000	487.250000	437.250000	240.000000
50%	242.500000	832.500000	743.500000	410.000000
75%	338.750000	1177.750000	1049.750000	580.000000
max	435.000000	1523.000000	1356.000000	750.000000

In [33]:

```
East=BuyerRatio['East'].mean()
West=BuyerRatio['West'].mean()
North=BuyerRatio['North'].mean()
South=BuyerRatio['South'].mean()

print('East Mean = ',East)
print('West Mean = ',West)
print('North Mean = ',North)
print('South Mean = ',South)
```

```
East Mean =  242.5
West Mean =  832.5
North Mean =  743.5
South Mean =  410.0
```

The Null and Alternative Hypothesis

There are no significant differences between the groups' mean Lab values. H0:μ1=μ2=μ3=μ4=μ5

There is a significant difference between the groups' mean Lab values. Ha:μ1≠μ2≠μ3≠μ4

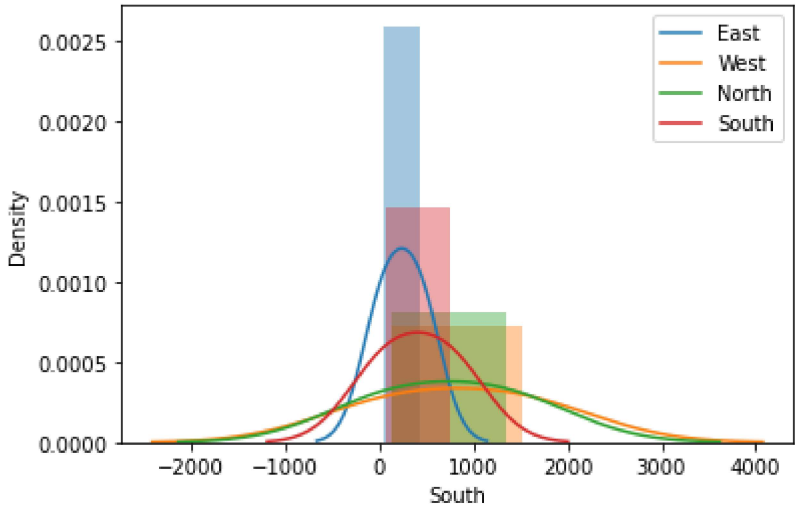
In [34]:

```
sns.distplot(BuyerRatio['East'])
sns.distplot(BuyerRatio['West'])
sns.distplot(BuyerRatio['North'])
sns.distplot(BuyerRatio['South'])
plt.legend(['East','West','North','South'])
```

```
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
C:\Users\HP\anaconda3\lib\site-packages\seaborn\distributions.py:2619: Futur
eWarning: `distplot` is a deprecated function and will be removed in a futur
e version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for h
istograms).
  warnings.warn(msg, FutureWarning)
```

Out[34]:

```
<matplotlib.legend.Legend at 0x1cd6d1dfd60>
```

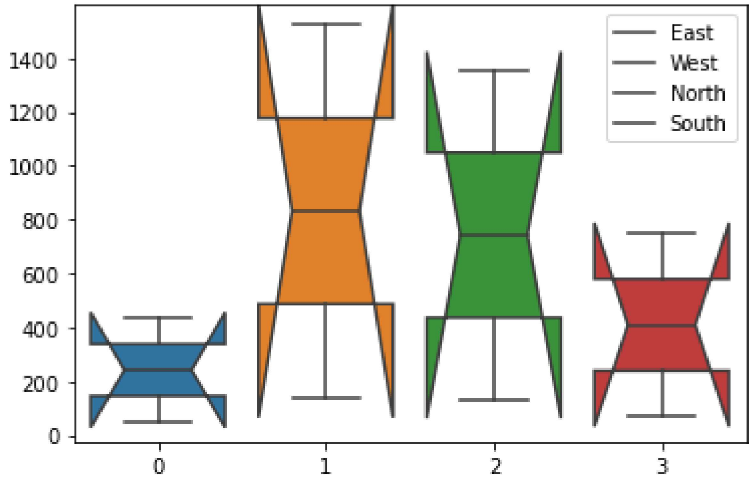


In [35]:

```
sns.boxplot(data=[BuyerRatio['East'],BuyerRatio['West'],BuyerRatio['North'],BuyerRatio['Sou
plt.legend(['East','West','North','South'])
```

Out[35]:

```
<matplotlib.legend.Legend at 0x1cd6d173850>
```



In [36]:

```
alpha=0.05
Male = [50,142,131,70]
Female=[435,1523,1356,750]
Sales=[Male,Female]
print(Sales)
```

```
[[50, 142, 131, 70], [435, 1523, 1356, 750]]
```

In [37]:

```
chiStats = sp.stats.chi2_contingency(Sales)
print('Test t=%f p-value=%f' % (chiStats[0], chiStats[1]))
print('Interpret by p-Value')
if chiStats[1] < 0.05:
  print('we reject null hypothesis')
else:
  print('we accept null hypothesis')
```

```
Test t=1.595946 p-value=0.660309
Interpret by p-Value
we accept null hypothesis
```

In [38]:

```
#critical value = 0.1
alpha = 0.05
critical_value = sp.stats.chi2.ppf(q = 1 - alpha,df=chiStats[2])# Find the critical value f
                      #degree of freedom

observed_chi_val = chiStats[0]
#if observed chi-square < critical chi-square, then variables are not related
#if observed chi-square > critical chi-square, then variables are not independent (and henc
print('Interpret by critical value')
if observed_chi_val <= critical_value:
    # observed value is not in critical area therefore we accept null hypothesis
    print ('Null hypothesis cannot be rejected (variables are not related)')
else:
    # observed value is in critical area therefore we reject null hypothesis
    print ('Null hypothesis cannot be excepted (variables are not independent)')
```

```
Interpret by critical value
Null hypothesis cannot be rejected (variables are not related)
```

Inference : proportion of male and female across regions is same

4. TeleCall uses 4 centers around the globe to process customer order forms. They audit a certain % of the customer order forms. Any error in order form renders it defective and has to be reworked before processing. The manager wants to check whether the defective % varies by centre. Please analyze the data at 5% significance level and help the manager draw appropriate inferences

In [40]:

```
Customer = pd.read_csv('C:/Users/Hp/Downloads/Costomer+OrderForm.csv')
Customer.head()
```

Out[40]:

	Phillippines	Indonesia	Malta	India
0	Error Free	Error Free	Defective	Error Free
1	Error Free	Error Free	Error Free	Defective
2	Error Free	Defective	Defective	Error Free
3	Error Free	Error Free	Error Free	Error Free
4	Error Free	Error Free	Defective	Error Free

In [41]:

```
Customer.describe()
```

Out[41]:

	Phillippines	Indonesia	Malta	India
count	300	300	300	300
unique	2	2	2	2
top	Error Free	Error Free	Error Free	Error Free
freq	271	267	269	280

In [42]:

```
Phillippines_value=Customer['Phillippines'].value_counts()
Indonesia_value=Customer['Indonesia'].value_counts()
Malta_value=Customer['Malta'].value_counts()
India_value=Customer['India'].value_counts()
print(Phillippines_value)
print(Indonesia_value)
print(Malta_value)
print(India_value)
```

```
Error Free    271
Defective      29
Name: Phillippines, dtype: int64
Error Free    267
Defective      33
Name: Indonesia, dtype: int64
Error Free    269
Defective      31
Name: Malta, dtype: int64
Error Free    280
Defective      20
Name: India, dtype: int64
```

In [ ]:

```
chiStats = sp.stats.chi2_contingency([[271,267,269,280],[29,33,31,20]])
print('Test t=%f p-value=%f' % (chiStats[0], chiStats[1]))
print('Interpret by p-Value')
if chiStats[1] < 0.05:
  print('we reject null hypothesis')
else:
  print('we accept null hypothesis')
```

In [43]:

```
#critical value = 0.1
alpha = 0.05
critical_value = sp.stats.chi2.ppf(q = 1 - alpha,df=chiStats[2])
observed_chi_val = chiStats[0]
print('Interpret by critical value')
if observed_chi_val <= critical_value:
       print ('Null hypothesis cannot be rejected (variables are not related)')
else:
       print ('Null hypothesis cannot be excepted (variables are not independen
```

```
  File "C:\Users\HP\AppData\Local\Temp/ipykernel_14796/2994648028.py", line
9
    print ('Null hypothesis cannot be excepted (variables are not independen

 ^
SyntaxError: EOL while scanning string literal
```

In [44]:

```
#critical value = 0.1
alpha = 0.05
critical_value = sp.stats.chi2.ppf(q = 1 - alpha,df=chiStats[2])
observed_chi_val = chiStats[0]
print('Interpret by critical value')
if observed_chi_val <= critical_value:
       print ('Null hypothesis cannot be rejected (variables are not related)')
else:
       print ('Null hypothesis cannot be excepted (variables are not independent)')
```

```
Interpret by critical value
Null hypothesis cannot be rejected (variables are not related)
```

In [45]:

```
chiStats = sp.stats.chi2_contingency([[271,267,269,280],[29,33,31,20]])
print('Test t=%f p-value=%f' % (chiStats[0], chiStats[1]))
print('Interpret by p-Value')
if chiStats[1] < 0.05:
  print('we reject null hypothesis')
else:
  print('we accept null hypothesis')
```

```
Test t=3.858961 p-value=0.277102
Interpret by p-Value
we accept null hypothesis
```

In [46]:

```
#critical value = 0.1
alpha = 0.05
critical_value = sp.stats.chi2.ppf(q = 1 - alpha,df=chiStats[2])
observed_chi_val = chiStats[0]
print('Interpret by critical value')
if observed_chi_val <= critical_value:
       print ('Null hypothesis cannot be rejected (variables are not related)')
else:
       print ('Null hypothesis cannot be excepted (variables are not independent)')
```

```
Interpret by critical value
Null hypothesis cannot be rejected (variables are not related)
```

Inference is that proportion of defective % across the center is same.

In [ ]: