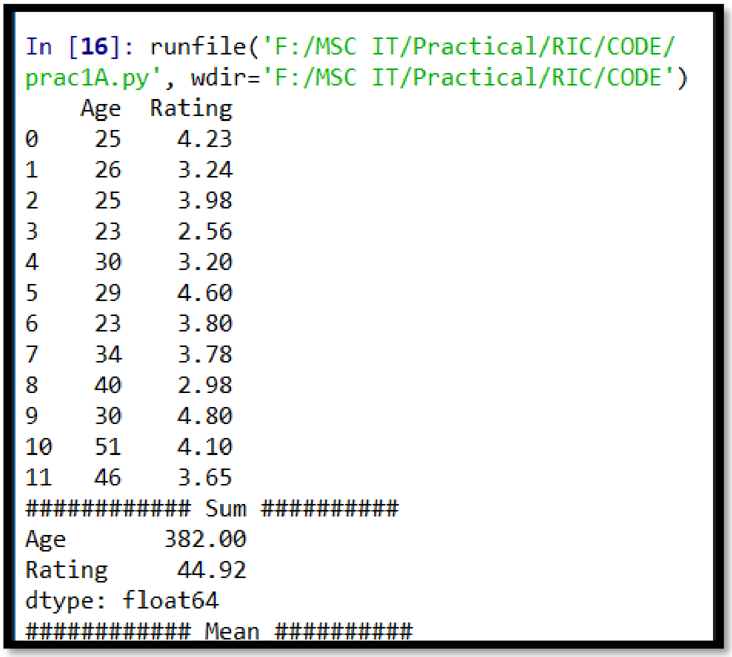
VSIT

Research in Computing
# Practical # 1

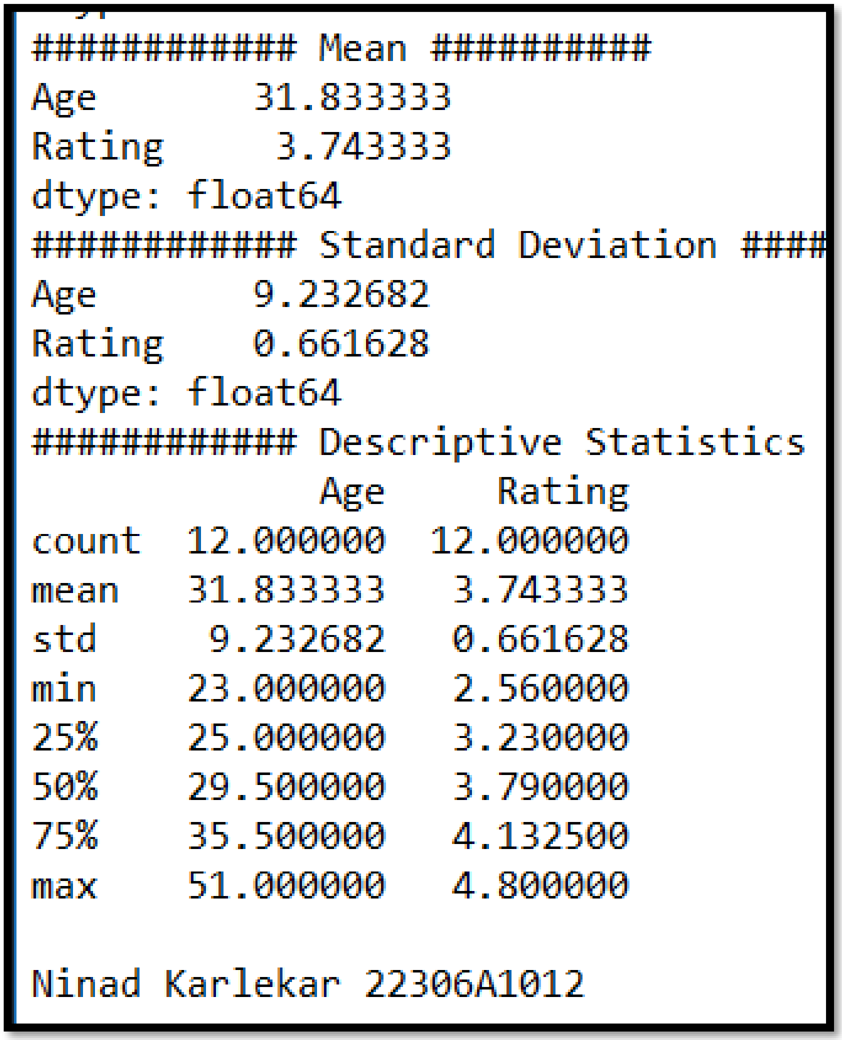
Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	descriptive statistics of data	**Batch**	1

A. Write a program for obtaining descriptive statistics of data.

```
#Practical 1A: Write a python program on descriptive statistics analysis.
################################################################
import pandas as pd
#Create a Dictionary of series
d = {'Age':pd.Series([25,26,25,23,30,29,23,34,40,30,51,46]),
'Rating':pd.Series([4.23,3.24,3.98,2.56,3.20,4.6,3.8,3.78,2.98,4.80,4.10,3.65])}
#Create a DataFrame
df = pd.DataFrame(d)
print(df)
print('############ Sum ########## ')
print (df.sum())
print('############ Mean ########## ')
print (df.mean())
print('############ Standard Deviation ########## ')
print (df.std())
print('\nNinad Karlekar 22306A1012')
```



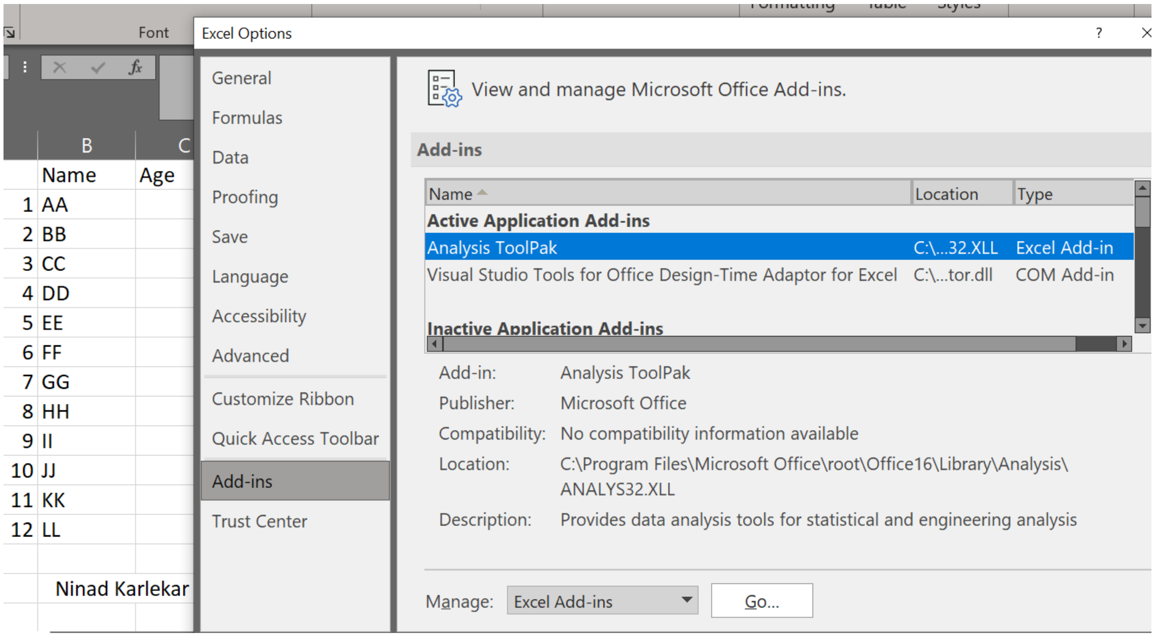
```
In [16]: runfile('F:/MSC IT/Practical/RIC/CODE/
prac1A.py', wdir='F:/MSC IT/Practical/RIC/CODE')
    Age  Rating
0    25    4.23
1    26    3.24
2    25    3.98
3    23    2.56
4    30    3.20
5    29    4.60
6    23    3.80
7    34    3.78
8    40    2.98
9    30    4.80
10   51    4.10
11   46    3.65
############ Sum ##########
Age       382.00
Rating     44.92
dtype: float64
############ Mean ##########
```



```
############ Mean ##########
Age         31.833333
Rating       3.743333
dtype: float64
############ Standard Deviation ####
Age         9.232682
Rating      0.661628
dtype: float64
############ Descriptive Statistics
             Age     Rating
count  12.000000  12.000000
mean   31.833333   3.743333
std     9.232682   0.661628
min    23.000000   2.560000
25%    25.000000   3.230000
50%    29.500000   3.790000
75%    35.500000   4.132500
max    51.000000   4.800000

Ninad Karlekar 22306A1012
```

Steps (EXCEL):

1. Open Excel file
2. Go to File -> Options -> Add-ins -> Click on Analysis Toolpack -> click on Go



View and manage Microsoft Office Add-ins.

Add-ins

Name	Location	Type
Active Application Add-ins		
Analysis ToolPak	C:\...32.XLL	Excel Add-in
Visual Studio Tools for Office Design-Time Adaptor for Excel	C:\...tor.dll	COM Add-in
Inactive Application Add-ins		

Add-in: Analysis ToolPak

Publisher: Microsoft Office

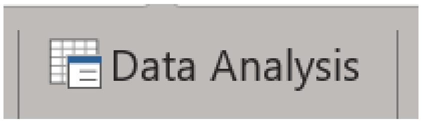
Compatibility: No compatibility information available

Location: C:\Program Files\Microsoft Office\root\Office16\Library\Analysis\ANALYS32.XLL

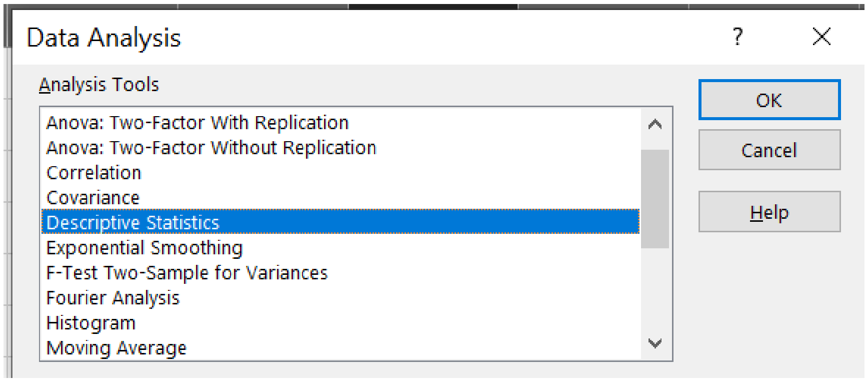
Description: Provides data analysis tools for statistical and engineering analysis

Manage: Excel Add-ins

3. Tick mark Analysis toolpack -> click on OK -> The Data Analysis option will be added in Data tab

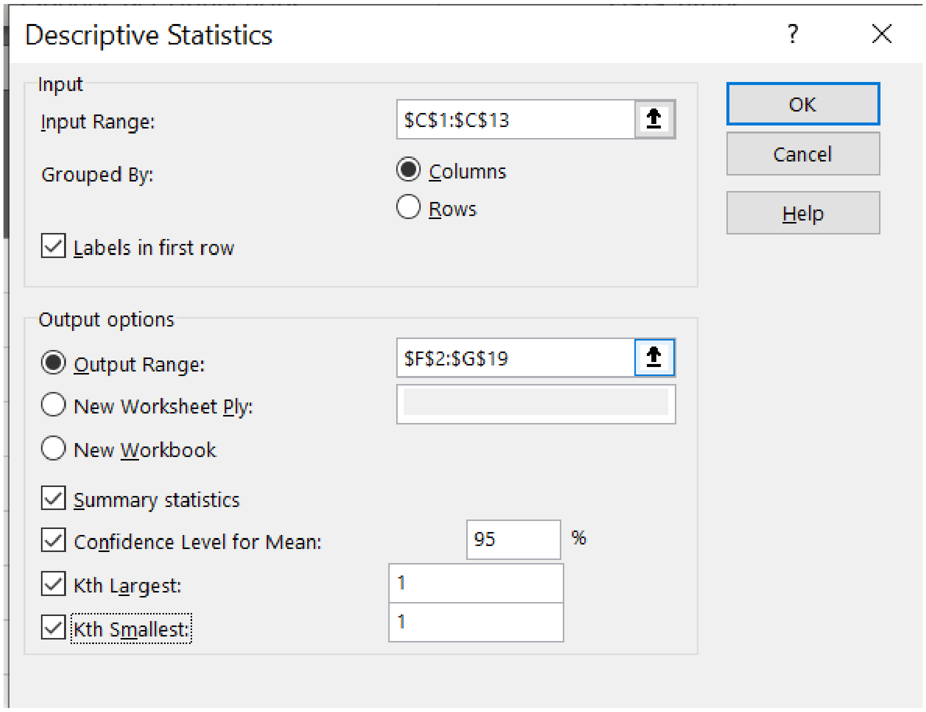


4. Now click on Data analysis -> Descriptive Statistiscs -> click on OK



5. Click on input range -> select **Age** column in input column -> and select any blank coloumn in output range -> Tick Mark on **Summery statistics, confidence level for mean(95%), kth Largest(1), kth**

smallest(1). -> click on Ok



Descriptive Statistics

Input

Input Range: $C$1:$C$13

Grouped By: ● Columns ○ Rows

☑ Labels in first row

Output options

● Output Range: $F$2:$G$19

○ New Worksheet Ply:

○ New Workbook

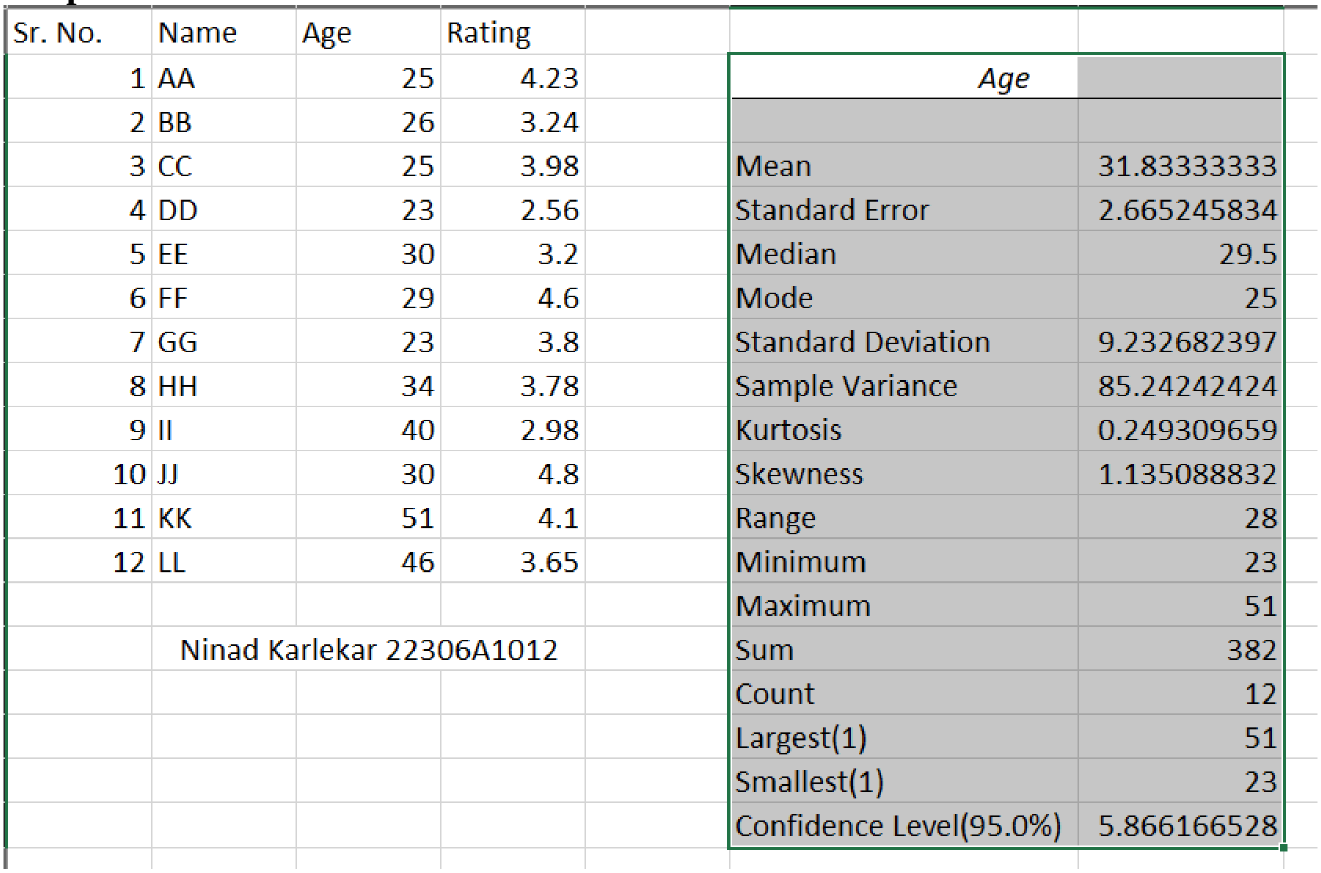
☑ Summary statistics

☑ Confidence Level for Mean: 95 %

☑ Kth Largest: 1

☑ Kth Smallest: 1

Output:



Sr. No.	Name	Age	Rating
1	AA	25	4.23
2	BB	26	3.24
3	CC	25	3.98
4	DD	23	2.56
5	EE	30	3.2
6	FF	29	4.6
7	GG	23	3.8
8	HH	34	3.78
9	II	40	2.98
10	JJ	30	4.8
11	KK	51	4.1
12	LL	46	3.65

Ninad Karlekar 22306A1012

Age	
Mean	31.83333333
Standard Error	2.665245834
Median	29.5
Mode	25
Standard Deviation	9.232682397
Sample Variance	85.24242424
Kurtosis	0.249309659
Skewness	1.135088832
Range	28
Minimum	23
Maximum	51
Sum	382
Count	12
Largest(1)	51
Smallest(1)	23
Confidence Level(95.0%)	5.866166528

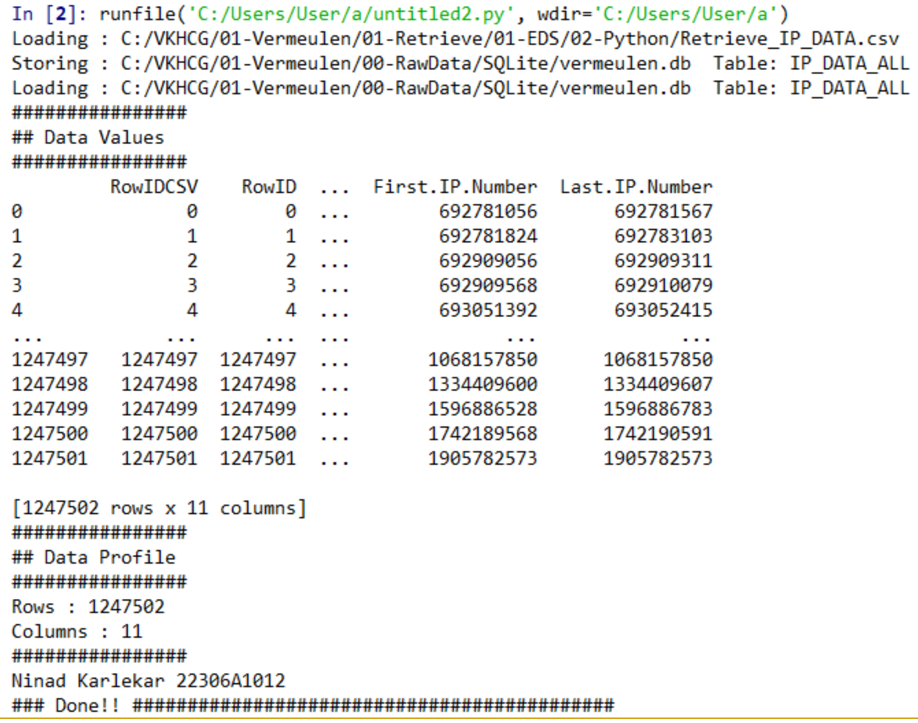
B. Import data from different data sources (from Excel, csv, mysql, sql server, oracle to R/Python/Excel)

From csv

Python Code:

```
import sqlite3 as sq
import pandas as pd
Base='C:/VKHCG'
sDatabaseName=Base + '/01-Vermeulen/00-RawData/SQLite/vermeulen.db'
conn = sq.connect(sDatabaseName)
sFileName='C:/VKHCG/01-Vermeulen/01-Retrieve/01-EDS/02-Python/Retrieve_IP_DATA.csv'
print('Loading :',sFileName)
IP_DATA_ALL_FIX=pd.read_csv(sFileName,header=0,low_memory=False)
IP_DATA_ALL_FIX.index.names = ['RowIDCSV']
sTable='IP_DATA_ALL'
print('Storing :',sDatabaseName,' Table:',sTable)
IP_DATA_ALL_FIX.to_sql(sTable, conn, if_exists="replace")
print('Loading :',sDatabaseName,' Table:',sTable)
TestData=pd.read_sql_query("select * from IP_DATA_ALL;", conn)
print('## Data Values')
print(TestData)
print('################')
print('## Data Profile')
print('################')
print('Rows :',TestData.shape[0])
print('Columns :',TestData.shape[1])
print("Ninad Karlekar 22306A1012")
print('### Done!! #############################################')
```

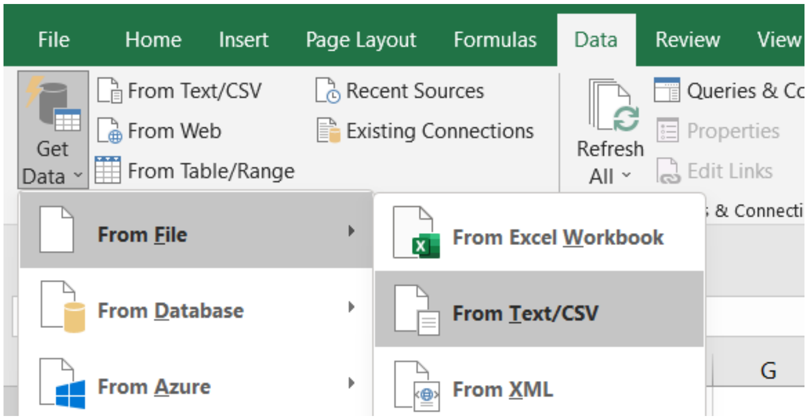
Output:



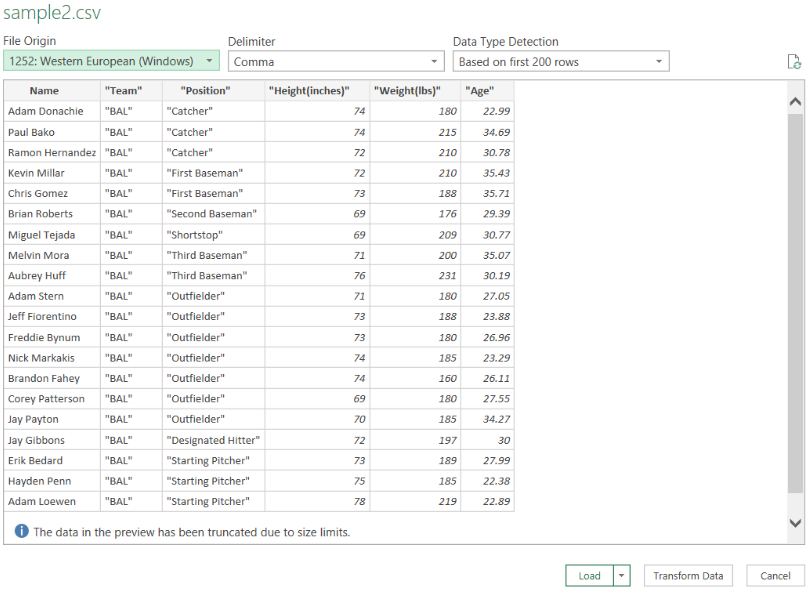
```
In [2]: runfile('C:/Users/User/a/untitled2.py', wdir='C:/Users/User/a')
Loading : C:/VKHCG/01-Vermeulen/01-Retrieve/01-EDS/02-Python/Retrieve_IP_DATA.csv
Storing : C:/VKHCG/01-Vermeulen/00-RawData/SQLite/vermeulen.db  Table: IP_DATA_ALL
Loading : C:/VKHCG/01-Vermeulen/00-RawData/SQLite/vermeulen.db  Table: IP_DATA_ALL
################
## Data Values
################
         RowIDCSV   RowID  ...  First.IP.Number  Last.IP.Number
0               0       0  ...        692781056       692781567
1               1       1  ...        692781824       692783103
2               2       2  ...        692909056       692909311
3               3       3  ...        692909568       692910079
4               4       4  ...        693051392       693052415
...           ...     ...  ...              ...             ...
1247497   1247497 1247497  ...       1068157850      1068157850
1247498   1247498 1247498  ...       1334409600      1334409607
1247499   1247499 1247499  ...       1596886528      1596886783
1247500   1247500 1247500  ...       1742189568      1742190591
1247501   1247501 1247501  ...       1905782573      1905782573

[1247502 rows x 11 columns]
################
## Data Profile
################
Rows : 1247502
Columns : 11
################
Ninad Karlekar 22306A1012
### Done!! #############################################
```

To Import data from csv file.

1. In Data tab, Click on Get Data
2. Select From file -> Select From Text/CSV



3. Select the csv file and Click on Import
4. Preview window will open→click on Load


sample2.csv

File Origin: 1252: Western European (Windows)

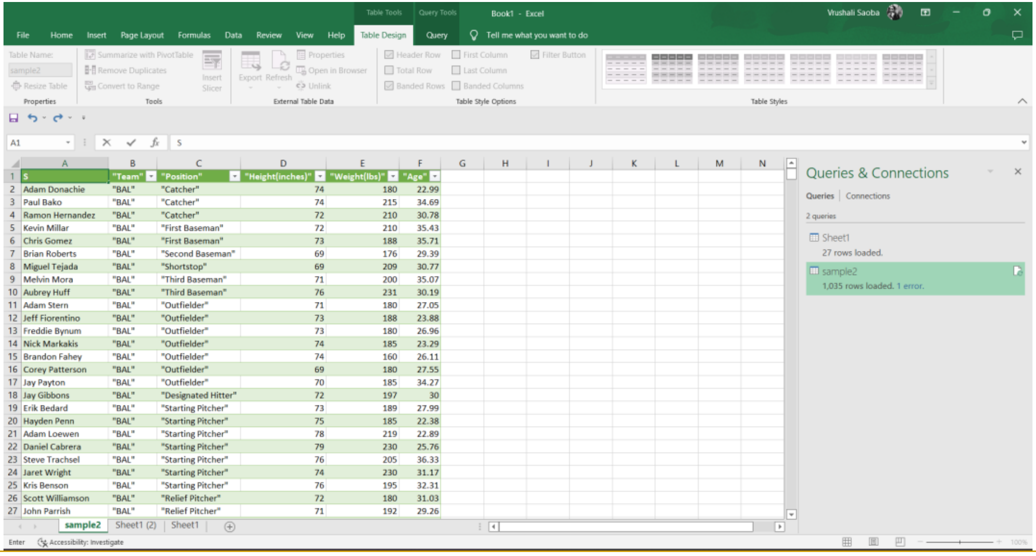
Delimiter: Comma

Data Type Detection: Based on first 200 rows

Name	"Team"	"Position"	"Height(inches)"	"Weight(lbs)"	"Age"
Adam Donachie	"BAL"	"Catcher"	74	180	22.99
Paul Bako	"BAL"	"Catcher"	74	215	34.69
Ramon Hernandez	"BAL"	"Catcher"	72	210	30.78
Kevin Millar	"BAL"	"First Baseman"	72	210	35.43
Chris Gomez	"BAL"	"First Baseman"	73	188	35.71
Brian Roberts	"BAL"	"Second Baseman"	69	176	29.39
Miguel Tejada	"BAL"	"Shortstop"	69	209	30.77
Melvin Mora	"BAL"	"Third Baseman"	71	200	35.07
Aubrey Huff	"BAL"	"Third Baseman"	76	231	30.19
Adam Stern	"BAL"	"Outfielder"	71	180	27.05
Jeff Fiorentino	"BAL"	"Outfielder"	73	188	23.88
Freddie Bynum	"BAL"	"Outfielder"	73	180	26.96
Nick Markakis	"BAL"	"Outfielder"	74	185	23.29
Brandon Fahey	"BAL"	"Outfielder"	74	160	26.11
Corey Patterson	"BAL"	"Outfielder"	69	180	27.55
Jay Payton	"BAL"	"Outfielder"	70	185	34.27
Jay Gibbons	"BAL"	"Designated Hitter"	72	197	30
Erik Bedard	"BAL"	"Starting Pitcher"	73	189	27.99
Hayden Penn	"BAL"	"Starting Pitcher"	75	185	22.38
Adam Loewen	"BAL"	"Starting Pitcher"	78	219	22.89

The data in the preview has been truncated due to size limits.

5. The File will be Imported and will be shown



	A	B	C	D	E	F
1	S	"Team"	"Position"	"Height(inches)"	"Weight(lbs)"	"Age"
2	Adam Donachie	"BAL"	"Catcher"	74	180	22.99
3	Paul Bako	"BAL"	"Catcher"	74	215	34.69
4	Ramon Hernandez	"BAL"	"Catcher"	72	210	30.78
5	Kevin Millar	"BAL"	"First Baseman"	72	210	35.43
6	Chris Gomez	"BAL"	"First Baseman"	73	188	35.71
7	Brian Roberts	"BAL"	"Second Baseman"	69	176	29.39
8	Miguel Tejada	"BAL"	"Shortstop"	69	209	30.77
9	Melvin Mora	"BAL"	"Third Baseman"	71	200	35.07
10	Aubrey Huff	"BAL"	"Third Baseman"	76	231	30.19
11	Adam Stern	"BAL"	"Outfielder"	71	180	27.05
12	Jeff Fiorentino	"BAL"	"Outfielder"	73	188	23.88
13	Freddie Bynum	"BAL"	"Outfielder"	73	180	26.96
14	Nick Markakis	"BAL"	"Outfielder"	74	185	23.29
15	Brandon Fahey	"BAL"	"Outfielder"	74	160	26.11
16	Corey Patterson	"BAL"	"Outfielder"	69	180	27.55
17	Jay Payton	"BAL"	"Outfielder"	70	185	34.27
18	Jay Gibbons	"BAL"	"Designated Hitter"	72	197	30
19	Erik Bedard	"BAL"	"Starting Pitcher"	73	189	27.99
20	Hayden Penn	"BAL"	"Starting Pitcher"	75	185	22.38
21	Adam Loewen	"BAL"	"Starting Pitcher"	78	219	22.89
22	Daniel Cabrera	"BAL"	"Starting Pitcher"	79	230	25.76
23	Steve Trachsel	"BAL"	"Starting Pitcher"	76	205	36.33
24	Jaret Wright	"BAL"	"Starting Pitcher"	74	230	31.17
25	Kris Benson	"BAL"	"Starting Pitcher"	76	195	32.31
26	Scott Williamson	"BAL"	"Relief Pitcher"	72	180	31.03
27	John Parrish	"BAL"	"Relief Pitcher"	71	192	29.26

Queries & Connections

Queries | Connections

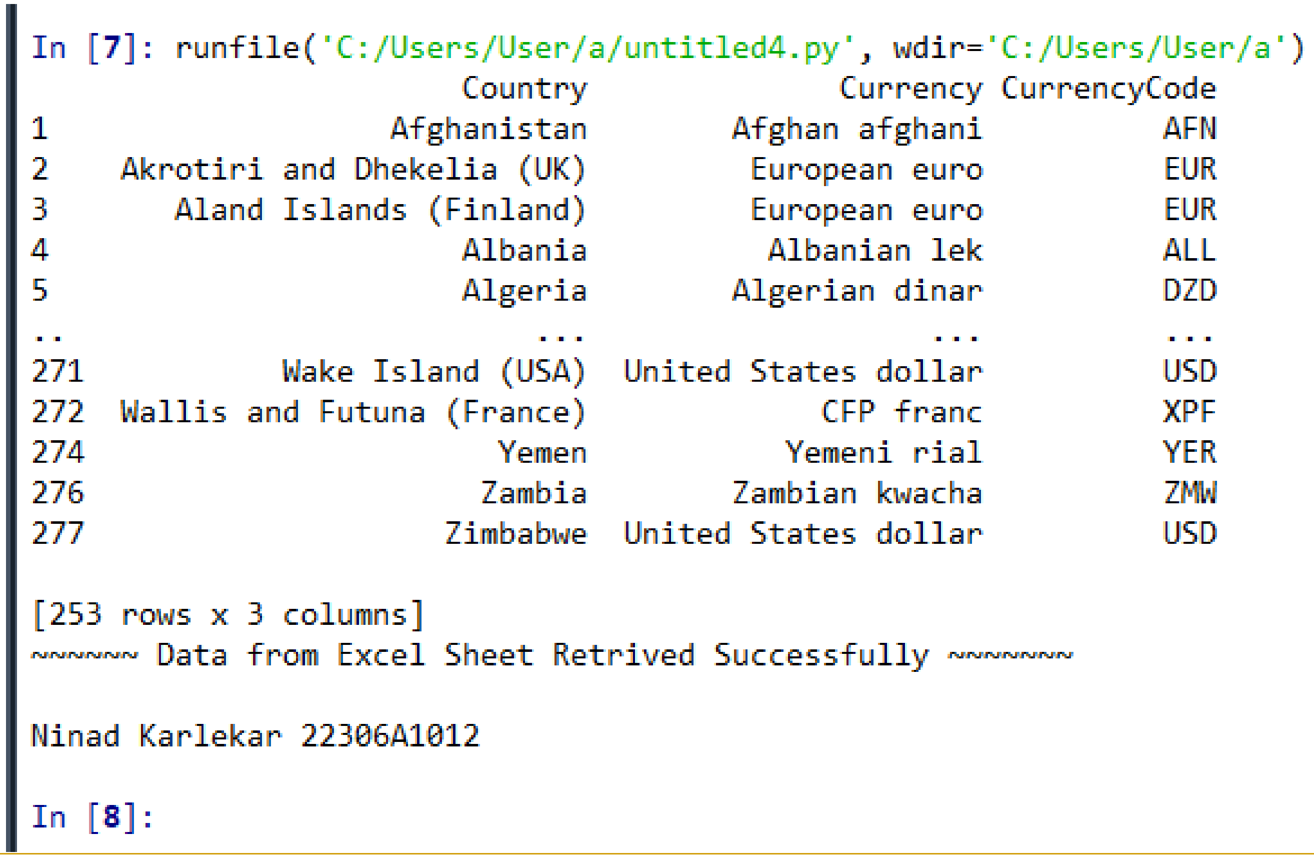
2 queries

Sheet1
27 rows loaded.

sample2
1,035 rows loaded. 1 error.

From Excel

```
import os
import pandas as pd
Base='F:/tmp/practical-data-science/VKHCG'
sFileDir=Base + '/01-Vermeulen/01-Retrieve/01-EDS/02-Python'

CurrencyRawData = pd.read_excel('F:/tmp/practical-data-science/VKHCG/01-Vermeulen/00-
RawData/Country_Currency.xlsx')
sColumns = ['Country or territory', 'Currency', 'ISO-4217']
CurrencyData = CurrencyRawData[sColumns]
CurrencyData.rename(columns={'Country or territory': 'Country', 'ISO-4217':
'CurrencyCode'}, inplace=True)
CurrencyData.dropna(subset=['Currency'],inplace=True)
CurrencyData['Country'] = CurrencyData['Country'].map(lambda x: x.strip())
CurrencyData['Currency'] = CurrencyData['Currency'].map(lambda x:
x.strip())
CurrencyData['CurrencyCode'] = CurrencyData['CurrencyCode'].map(lambda x:
x.strip())
print(CurrencyData)

print('~~~~~~ Data from Excel Sheet Retrived Successfully ~~~~~~~ ')

print("\nNinad Karlekar 22306A1012")
```

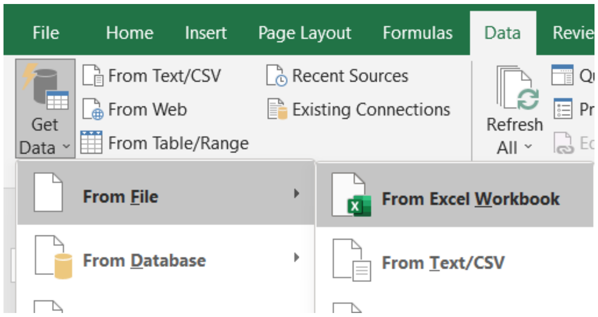


```
In [7]: runfile('C:/Users/User/a/untitled4.py', wdir='C:/Users/User/a')
                            Country              Currency CurrencyCode
1                       Afghanistan        Afghan afghani          AFN
2     Akrotiri and Dhekelia (UK)        European euro          EUR
3          Aland Islands (Finland)        European euro          EUR
4                           Albania          Albanian lek          ALL
5                           Algeria        Algerian dinar          DZD
..                              ...                   ...          ...
271               Wake Island (USA)  United States dollar          USD
272   Wallis and Futuna (France)             CFP franc          XPF
274                           Yemen           Yemeni rial          YER
276                          Zambia        Zambian kwacha          ZMW
277                        Zimbabwe  United States dollar          USD

[253 rows x 3 columns]
~~~~~~ Data from Excel Sheet Retrived Successfully ~~~~~~~

Ninad Karlekar 22306A1012

In [8]:
```

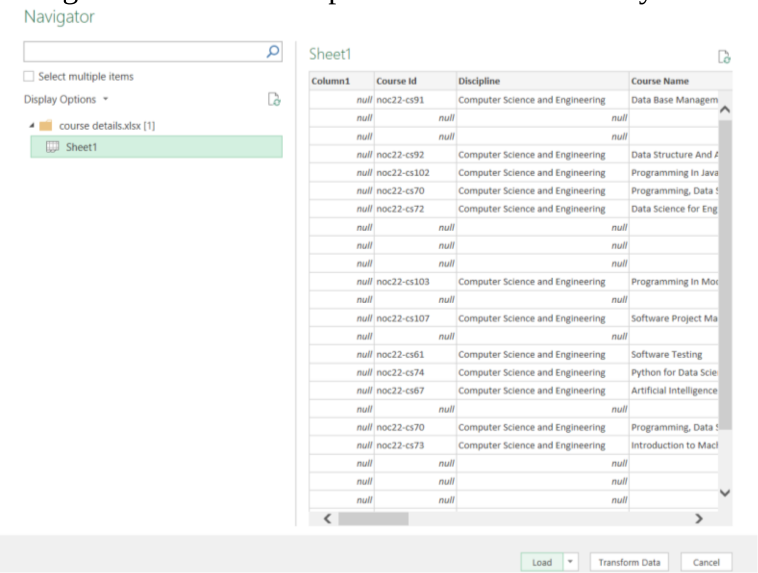
Procedure:-

To Import data from Excel sheet:

1. In Data tab, Click on Get Data
2. Select From file -> Select From Excel Workbook.



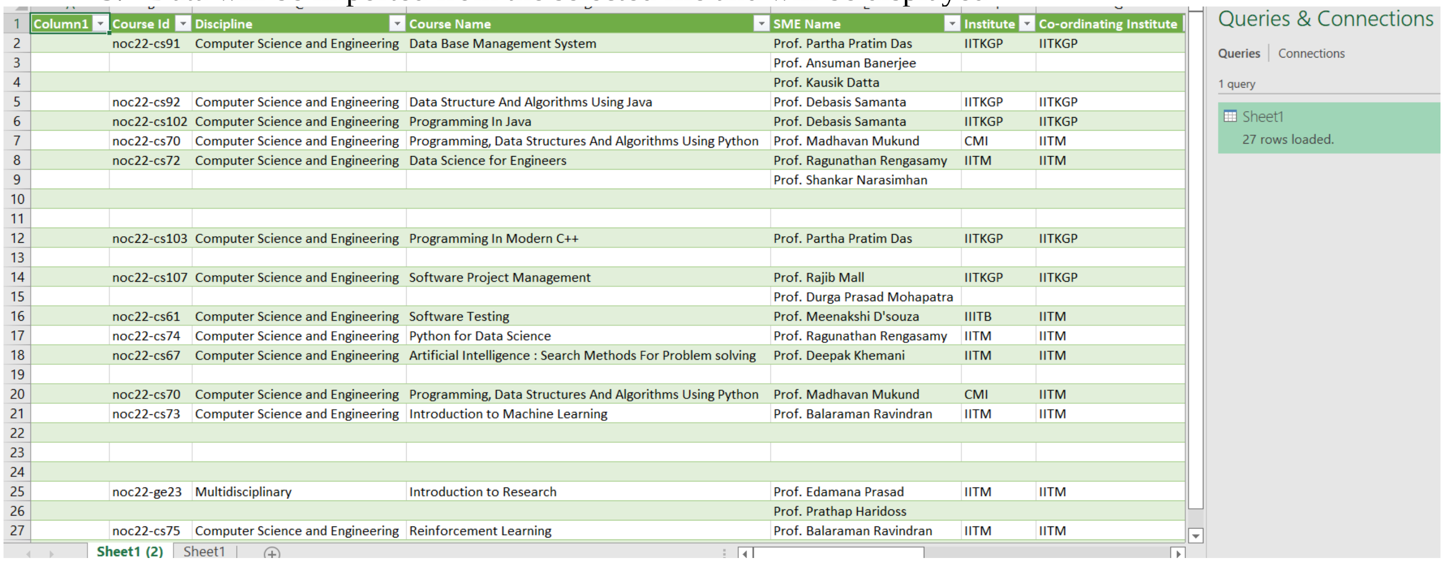
3. Select the Excel worksheet to import -> click on OK
4. Navigator Window will open -> Select the sheets you want to import -> Click on Load



Sheet1

Column1	Course Id	Discipline	Course Name
null	noc22-cs91	Computer Science and Engineering	Data Base Managem
null	null	null	
null	null	null	
null	noc22-cs92	Computer Science and Engineering	Data Structure And A
null	noc22-cs102	Computer Science and Engineering	Programming In Java
null	noc22-cs70	Computer Science and Engineering	Programming, Data S
null	noc22-cs72	Computer Science and Engineering	Data Science for Eng
null	null	null	
null	null	null	
null	null	null	
null	noc22-cs103	Computer Science and Engineering	Programming In Moc
null	null	null	
null	noc22-cs107	Computer Science and Engineering	Software Project Ma
null	null	null	
null	noc22-cs61	Computer Science and Engineering	Software Testing
null	noc22-cs74	Computer Science and Engineering	Python for Data Scie
null	noc22-cs67	Computer Science and Engineering	Artificial Intelligence
null	null	null	
null	noc22-cs70	Computer Science and Engineering	Programming, Data S
null	noc22-cs73	Computer Science and Engineering	Introduction to Macl
null	null	null	
null	null	null	
null	null	null	

5. Data will be imported from the selected file and will be displayed



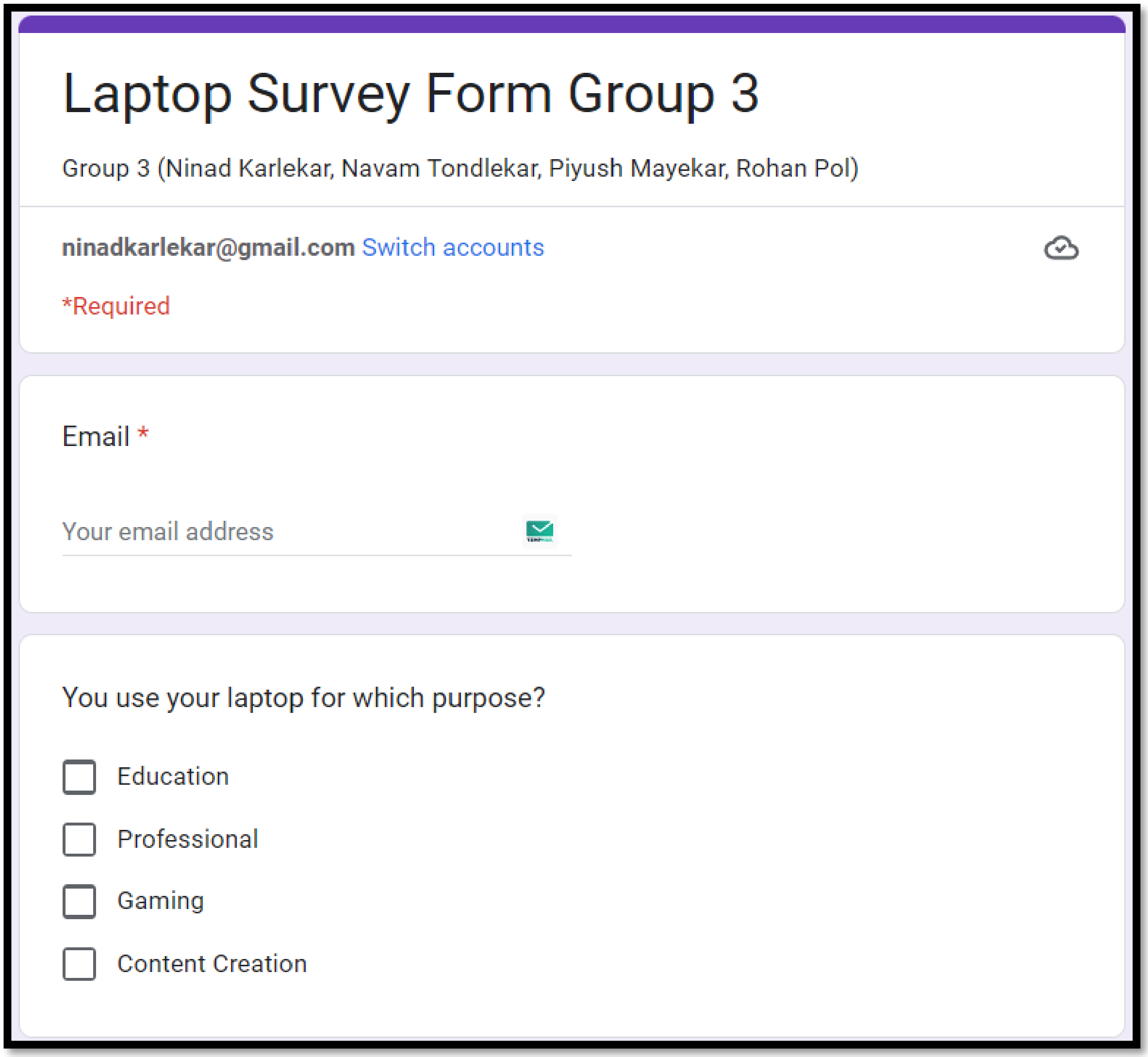
1	Column1	Course Id	Discipline	Course Name	SME Name	Institute	Co-ordinating Institute
2		noc22-cs91	Computer Science and Engineering	Data Base Management System	Prof. Partha Pratim Das	IITKGP	IITKGP
3					Prof. Ansuman Banerjee		
4					Prof. Kausik Datta		
5		noc22-cs92	Computer Science and Engineering	Data Structure And Algorithms Using Java	Prof. Debasis Samanta	IITKGP	IITKGP
6		noc22-cs102	Computer Science and Engineering	Programming In Java	Prof. Debasis Samanta	IITKGP	IITKGP
7		noc22-cs70	Computer Science and Engineering	Programming, Data Structures And Algorithms Using Python	Prof. Madhavan Mukund	CMI	IITM
8		noc22-cs72	Computer Science and Engineering	Data Science for Engineers	Prof. Ragunathan Rengasamy	IITM	IITM
9					Prof. Shankar Narasimhan		
10							
11							
12		noc22-cs103	Computer Science and Engineering	Programming In Modern C++	Prof. Partha Pratim Das	IITKGP	IITKGP
13							
14		noc22-cs107	Computer Science and Engineering	Software Project Management	Prof. Rajib Mall	IITKGP	IITKGP
15					Prof. Durga Prasad Mohapatra		
16		noc22-cs61	Computer Science and Engineering	Software Testing	Prof. Meenakshi D'souza	IIITB	IITM
17		noc22-cs74	Computer Science and Engineering	Python for Data Science	Prof. Ragunathan Rengasamy	IITM	IITM
18		noc22-cs67	Computer Science and Engineering	Artificial Intelligence : Search Methods For Problem solving	Prof. Deepak Khemani	IITM	IITM
19							
20		noc22-cs70	Computer Science and Engineering	Programming, Data Structures And Algorithms Using Python	Prof. Madhavan Mukund	CMI	IITM
21		noc22-cs73	Computer Science and Engineering	Introduction to Machine Learning	Prof. Balaraman Ravindran	IITM	IITM
22							
23							
24							
25		noc22-ge23	Multidisciplinary	Introduction to Research	Prof. Edamana Prasad	IITM	IITM
26					Prof. Prathap Haridoss		
27		noc22-cs75	Computer Science and Engineering	Reinforcement Learning	Prof. Balaraman Ravindran	IITM	IITM

Research in Computing
Practical # 2

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	Case study	**Batch**	1

A. Design a survey form for a given case study, collect the primary data and analyse it

https://forms.gle/SAxXR7HWY35MFf7o9



Laptop Survey Form Group 3

Group 3 (Ninad Karlekar, Navam Tondlekar, Piyush Mayekar, Rohan Pol)

ninadkarlekar@gmail.com Switch accounts

*Required

Email *

Your email address

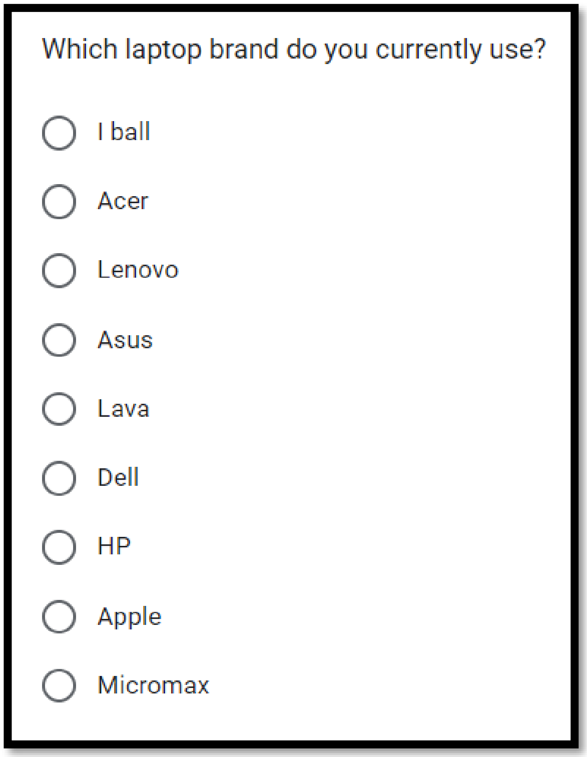
You use your laptop for which purpose?

□ Education

□ Professional

□ Gaming

□ Content Creation



Which laptop brand do you currently use?

○ I ball

○ Acer

○ Lenovo

○ Asus

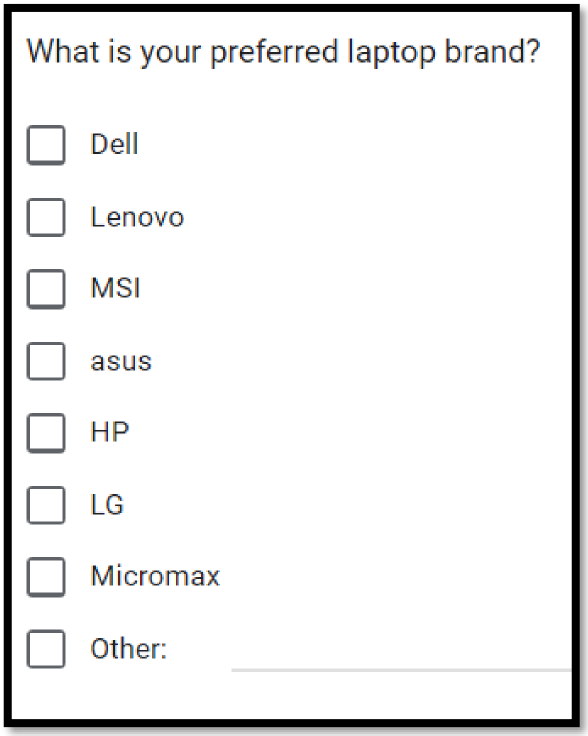
○ Lava

○ Dell

○ HP

○ Apple

○ Micromax



What is your preferred laptop brand?

□ Dell

□ Lenovo

□ MSI

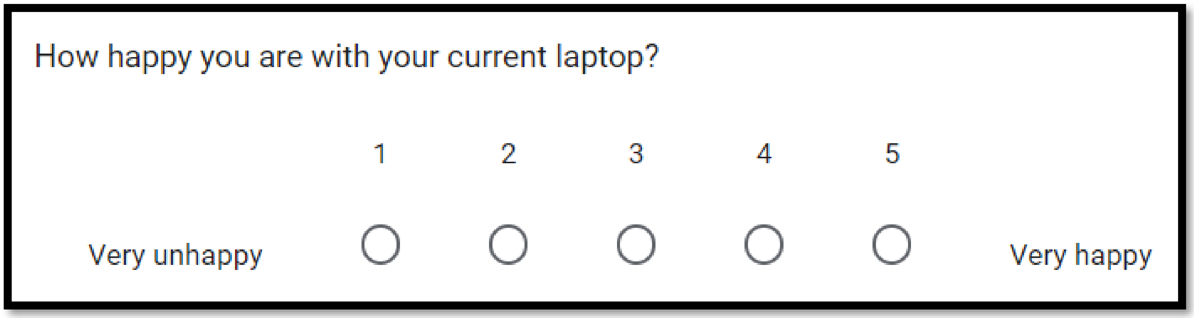
□ asus

□ HP

□ LG

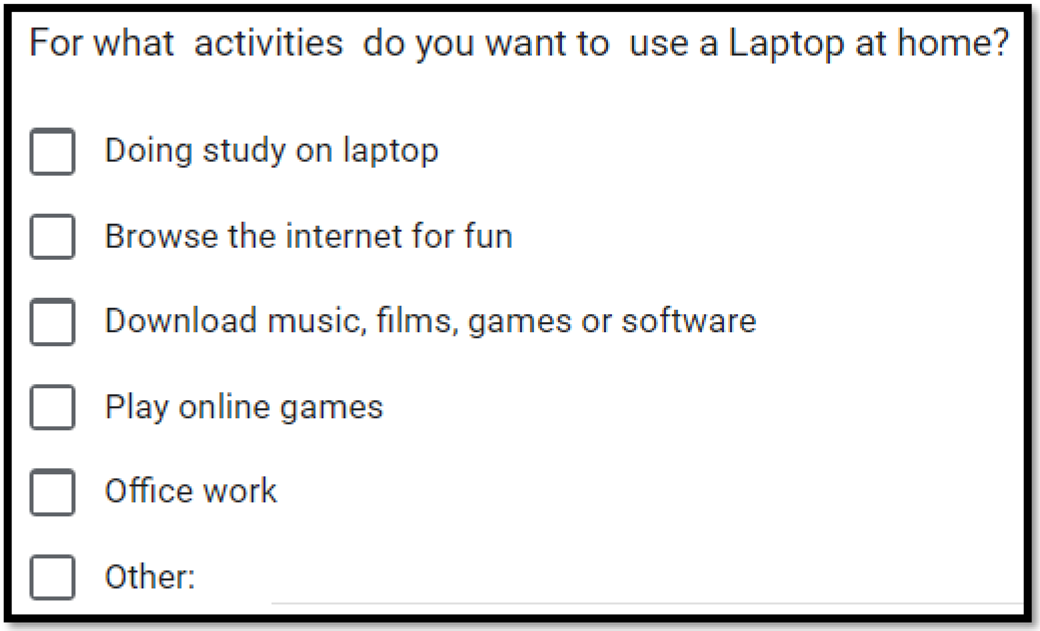
□ Micromax

□ Other: ______



How happy you are with your current laptop?

	1	2	3	4	5	
Very unhappy	○	○	○	○	○	Very happy



For what activities do you want to use a Laptop at home?

□ Doing study on laptop

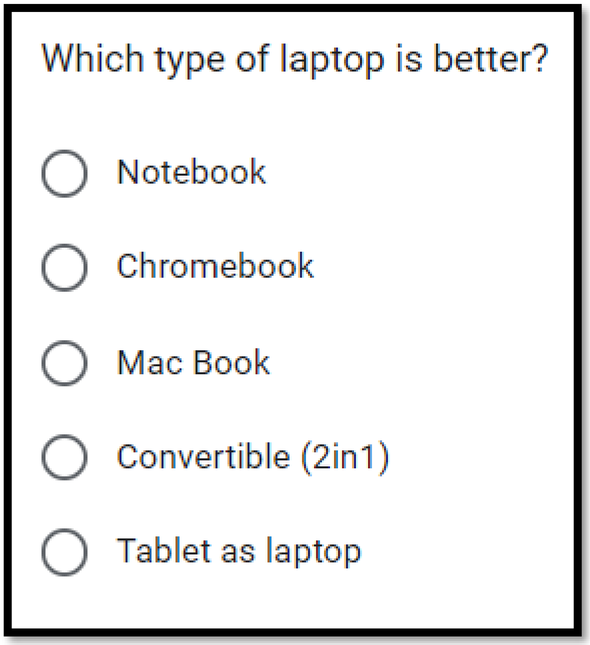
□ Browse the internet for fun

□ Download music, films, games or software

□ Play online games

□ Office work

□ Other: ______



Which type of laptop is better?

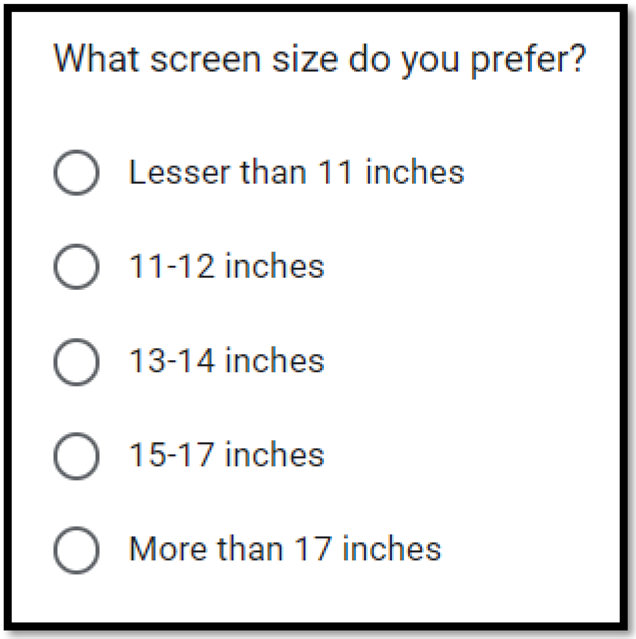
○ Notebook

○ Chromebook

○ Mac Book

○ Convertible (2in1)

○ Tablet as laptop



What screen size do you prefer?

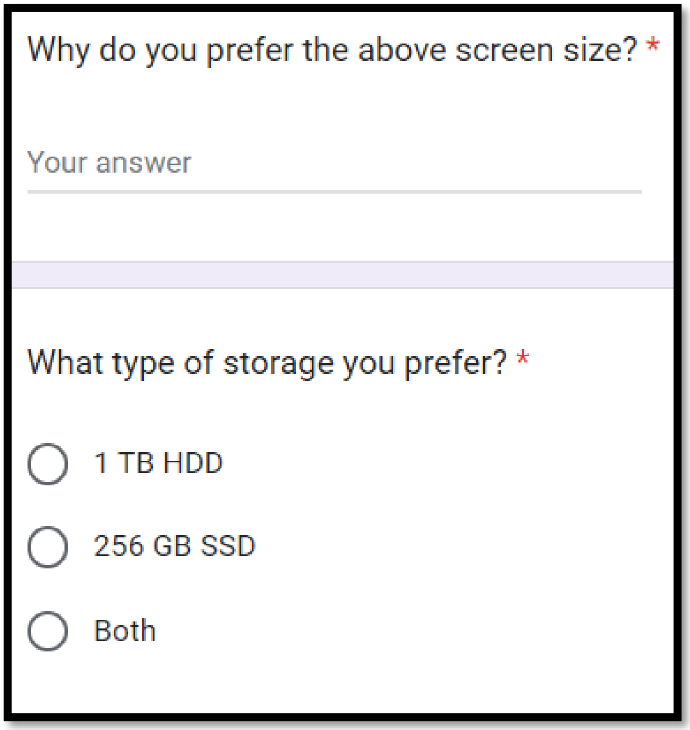
○ Lesser than 11 inches

○ 11-12 inches

○ 13-14 inches

○ 15-17 inches

○ More than 17 inches



Why do you prefer the above screen size? *

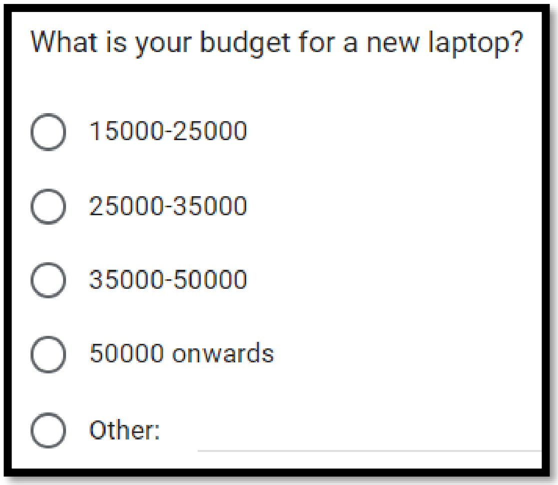
Your answer

What type of storage you prefer? *

○ 1 TB HDD

○ 256 GB SSD

○ Both



What is your budget for a new laptop?

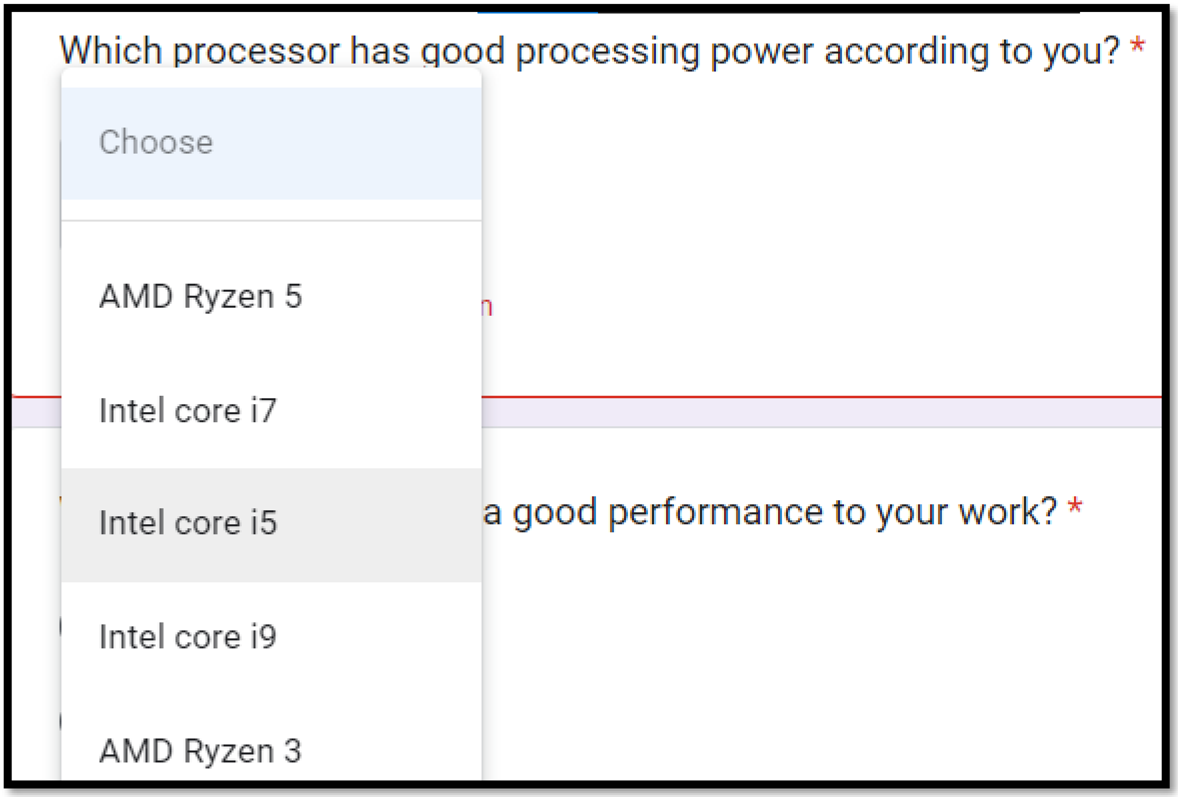
○ 15000-25000

○ 25000-35000

○ 35000-50000

○ 50000 onwards

○ Other:



Which processor has good processing power according to you? *

Choose

AMD Ryzen 5

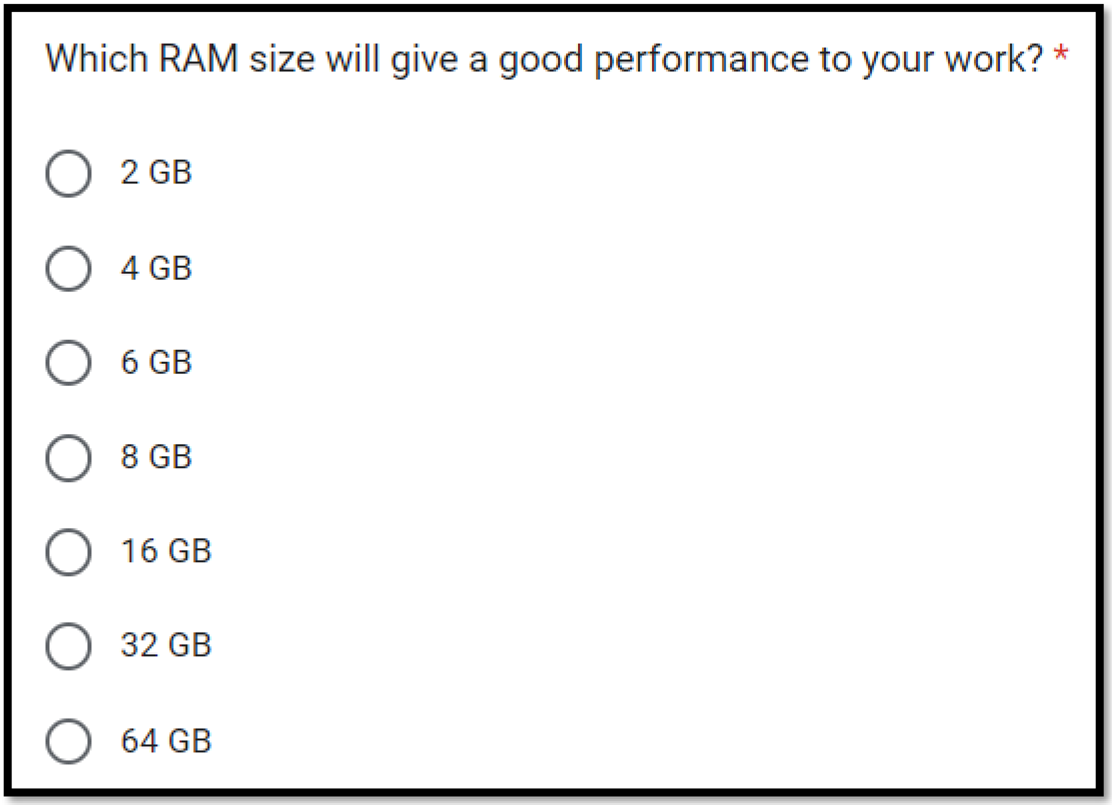
Intel core i7

Intel core i5

Intel core i9

AMD Ryzen 3

a good performance to your work? *



Which RAM size will give a good performance to your work? *

○ 2 GB

○ 4 GB

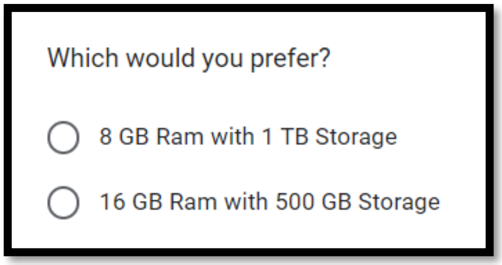
○ 6 GB

○ 8 GB

○ 16 GB

○ 32 GB

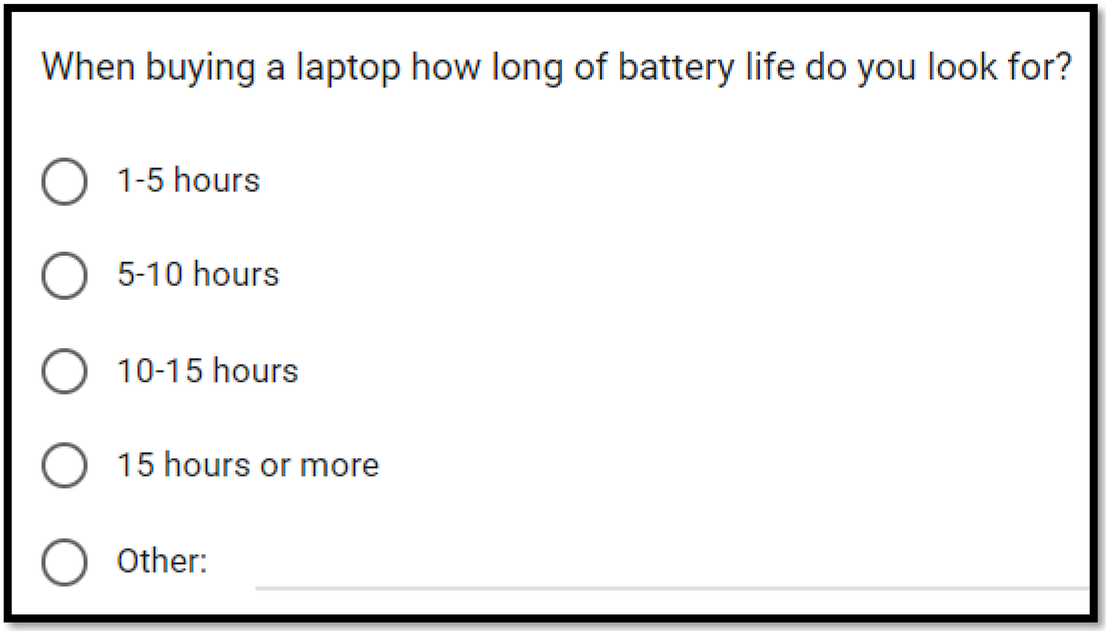
○ 64 GB



Which would you prefer?

○ 8 GB Ram with 1 TB Storage

○ 16 GB Ram with 500 GB Storage



When buying a laptop how long of battery life do you look for?

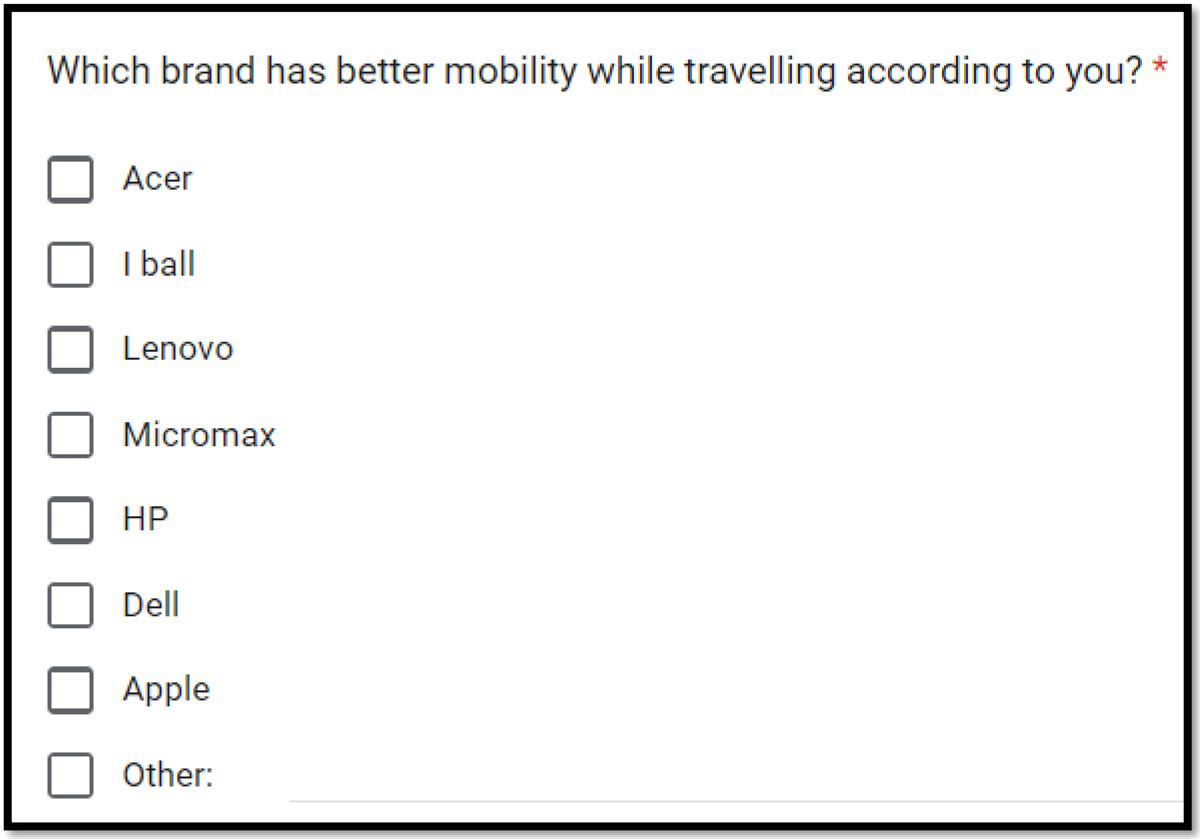
○ 1-5 hours

○ 5-10 hours

○ 10-15 hours

○ 15 hours or more

○ Other:



Which brand has better mobility while travelling according to you? *

□ Acer

□ I ball

□ Lenovo

□ Micromax

□ HP

□ Dell

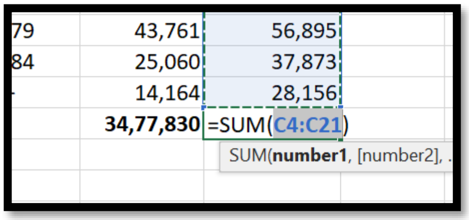
□ Apple

□ Other:

B. Perform analysis of given secondary data.

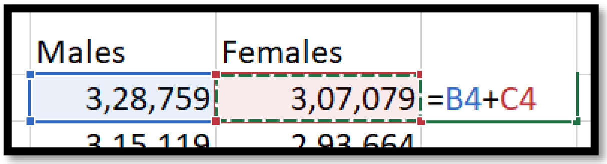
Steps:

1. Open World_population 2010 excel file.
2. Find the sum of Male ani Female Column.



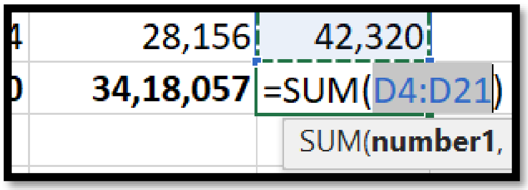
79	43,761	56,895
84	25,060	37,873
	14,164	28,156
	34,77,830	=SUM(C4:C21)
		SUM(**number1**, [number2], .

3. Create and find total of Male and Female coloumn (=B4+C4)



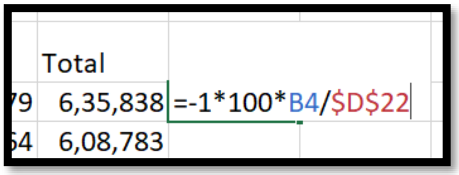
Males	Females	
3,28,759	3,07,079	=B4+C4
3,15,119	2,93,664	

4. Find Sum of all Total column values.



4	28,156	42,320
0	**34,18,057**	=SUM(D4:D21)
		SUM(**number1**,

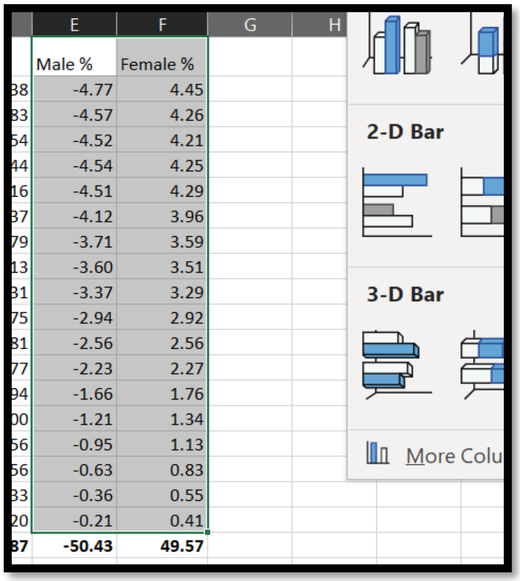
5. Find Percentage of Male (= -1*100*male column value/ sum of all total values) (=-1*100*B4/$D$22)



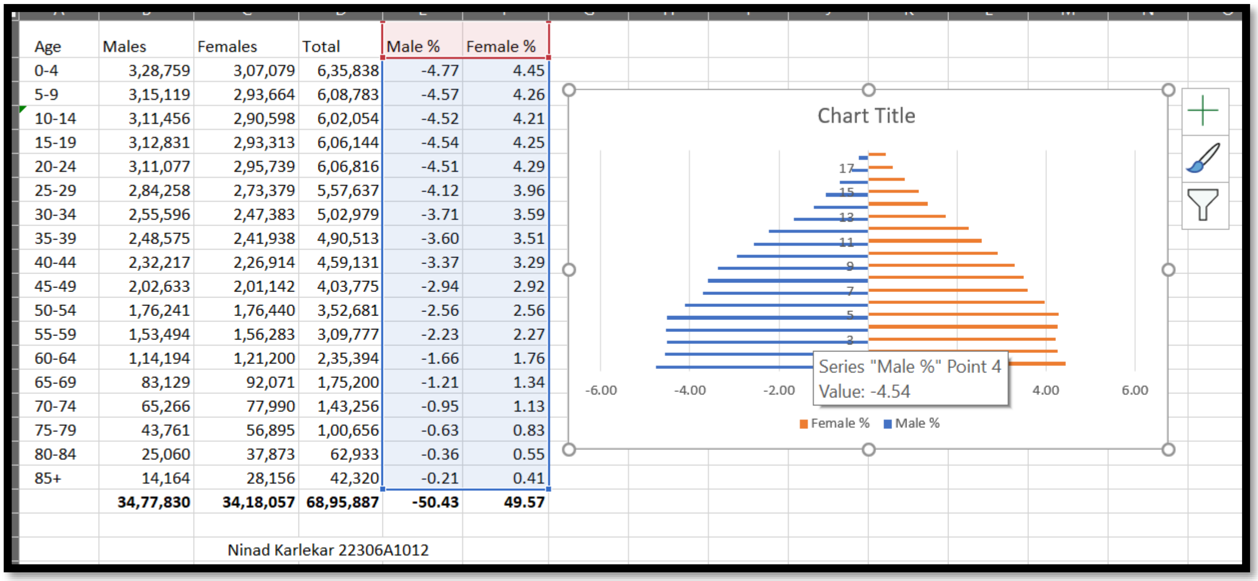
	Total	
79	6,35,838	=-1*100*B4/$D$22
54	6,08,783	

6. Find Percentage of Male (= 100*Female column value/ sum of all total values) (=100*C4/$D$22)
7. Find sum of both male% and female%

8. Select Male% and Female% -> insert -> clustered Bar



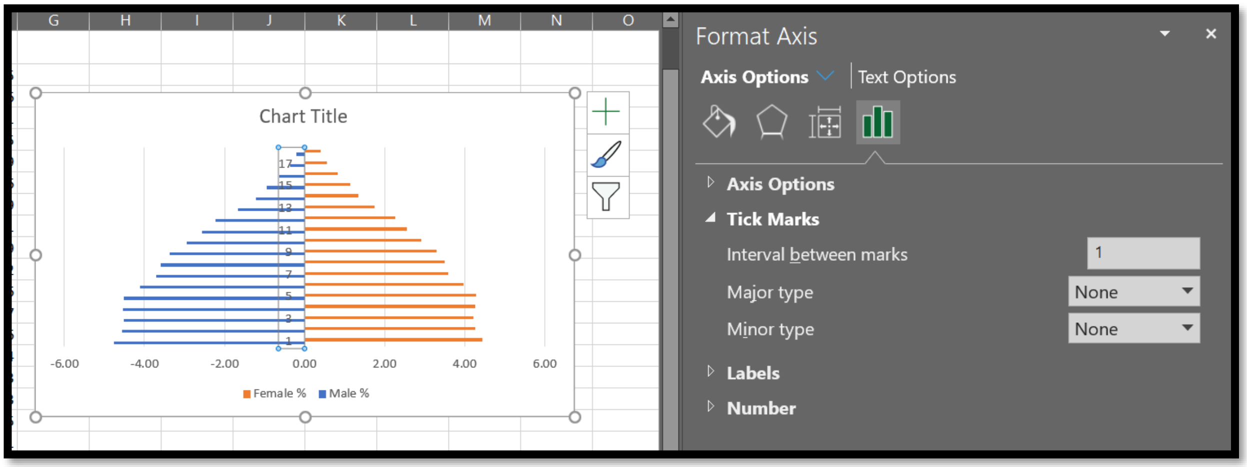
	Male %	Female %
38	-4.77	4.45
33	-4.57	4.26
54	-4.52	4.21
44	-4.54	4.25
16	-4.51	4.29
37	-4.12	3.96
79	-3.71	3.59
13	-3.60	3.51
31	-3.37	3.29
75	-2.94	2.92
81	-2.56	2.56
77	-2.23	2.27
94	-1.66	1.76
00	-1.21	1.34
56	-0.95	1.13
56	-0.63	0.83
33	-0.36	0.55
20	-0.21	0.41
87	**-50.43**	**49.57**



Age	Males	Females	Total	Male %	Female %
0-4	3,28,759	3,07,079	6,35,838	-4.77	4.45
5-9	3,15,119	2,93,664	6,08,783	-4.57	4.26
10-14	3,11,456	2,90,598	6,02,054	-4.52	4.21
15-19	3,12,831	2,93,313	6,06,144	-4.54	4.25
20-24	3,11,077	2,95,739	6,06,816	-4.51	4.29
25-29	2,84,258	2,73,379	5,57,637	-4.12	3.96
30-34	2,55,596	2,47,383	5,02,979	-3.71	3.59
35-39	2,48,575	2,41,938	4,90,513	-3.60	3.51
40-44	2,32,217	2,26,914	4,59,131	-3.37	3.29
45-49	2,02,633	2,01,142	4,03,775	-2.94	2.92
50-54	1,76,241	1,76,440	3,52,681	-2.56	2.56
55-59	1,53,494	1,56,283	3,09,777	-2.23	2.27
60-64	1,14,194	1,21,200	2,35,394	-1.66	1.76
65-69	83,129	92,071	1,75,200	-1.21	1.34
70-74	65,266	77,990	1,43,256	-0.95	1.13
75-79	43,761	56,895	1,00,656	-0.63	0.83
80-84	25,060	37,873	62,933	-0.36	0.55
85+	14,164	28,156	42,320	-0.21	0.41
	34,77,830	**34,18,057**	**68,95,887**	**-50.43**	**49.57**

Ninad Karlekar 22306A1012

9. Put the tip of your mouse arrow on the Y-axis (vertical axis) so it says “Category Axis”, right click and chose Format Axis



Format Axis

Axis Options | Text Options

Axis Options

Tick Marks

Interval between marks 1

Major type None

Minor type None

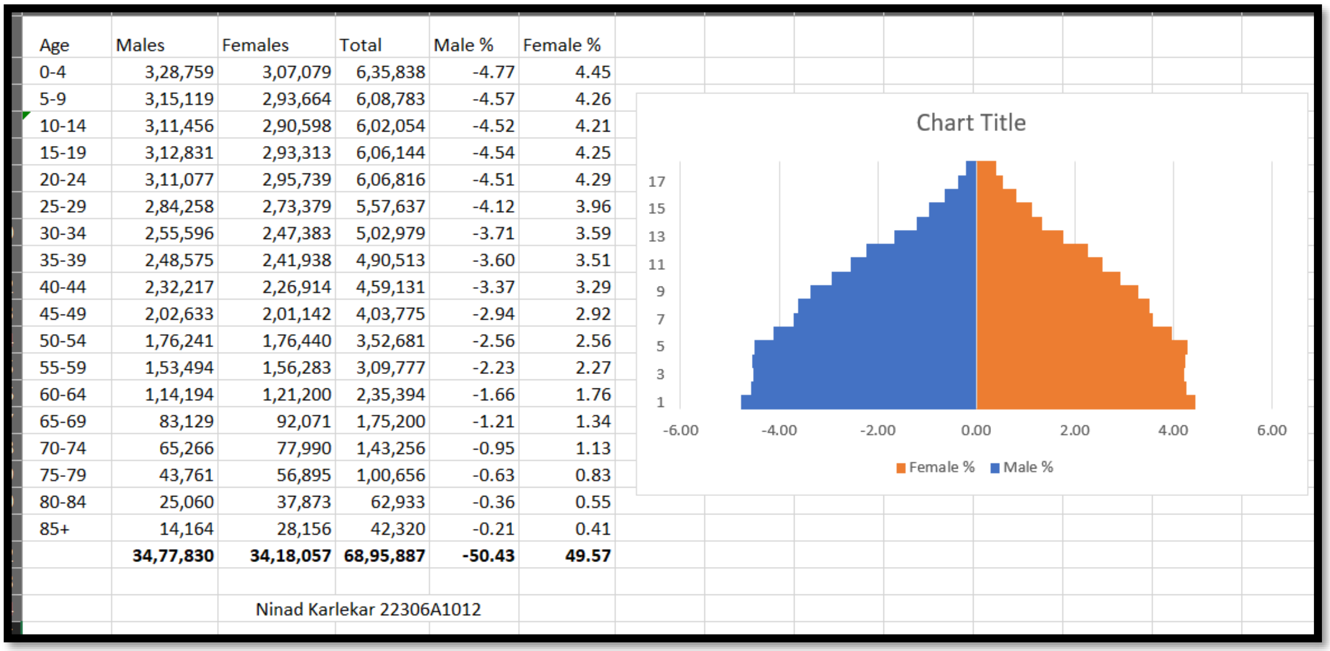
Labels

Number

10. Choose Axis options tab and set the major and minor tick mark type to None, Axis labels to Low, and click OK

11. Click on any of the bars in your pyramid, click right and select “format data series”. Set the **Overlap to 100** and **Gap Width to 0**. Click OK.

OUTPUT:



Age	Males	Females	Total	Male %	Female %
0-4	3,28,759	3,07,079	6,35,838	-4.77	4.45
5-9	3,15,119	2,93,664	6,08,783	-4.57	4.26
10-14	3,11,456	2,90,598	6,02,054	-4.52	4.21
15-19	3,12,831	2,93,313	6,06,144	-4.54	4.25
20-24	3,11,077	2,95,739	6,06,816	-4.51	4.29
25-29	2,84,258	2,73,379	5,57,637	-4.12	3.96
30-34	2,55,596	2,47,383	5,02,979	-3.71	3.59
35-39	2,48,575	2,41,938	4,90,513	-3.60	3.51
40-44	2,32,217	2,26,914	4,59,131	-3.37	3.29
45-49	2,02,633	2,01,142	4,03,775	-2.94	2.92
50-54	1,76,241	1,76,440	3,52,681	-2.56	2.56
55-59	1,53,494	1,56,283	3,09,777	-2.23	2.27
60-64	1,14,194	1,21,200	2,35,394	-1.66	1.76
65-69	83,129	92,071	1,75,200	-1.21	1.34
70-74	65,266	77,990	1,43,256	-0.95	1.13
75-79	43,761	56,895	1,00,656	-0.63	0.83
80-84	25,060	37,873	62,933	-0.36	0.55
85+	14,164	28,156	42,320	-0.21	0.41
	34,77,830	**34,18,057**	**68,95,887**	**-50.43**	**49.57**

Ninad Karlekar 22306A1012

VSIT

Research in Computing
Practical # 3

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	Testing Hypothesis	**Batch**	1

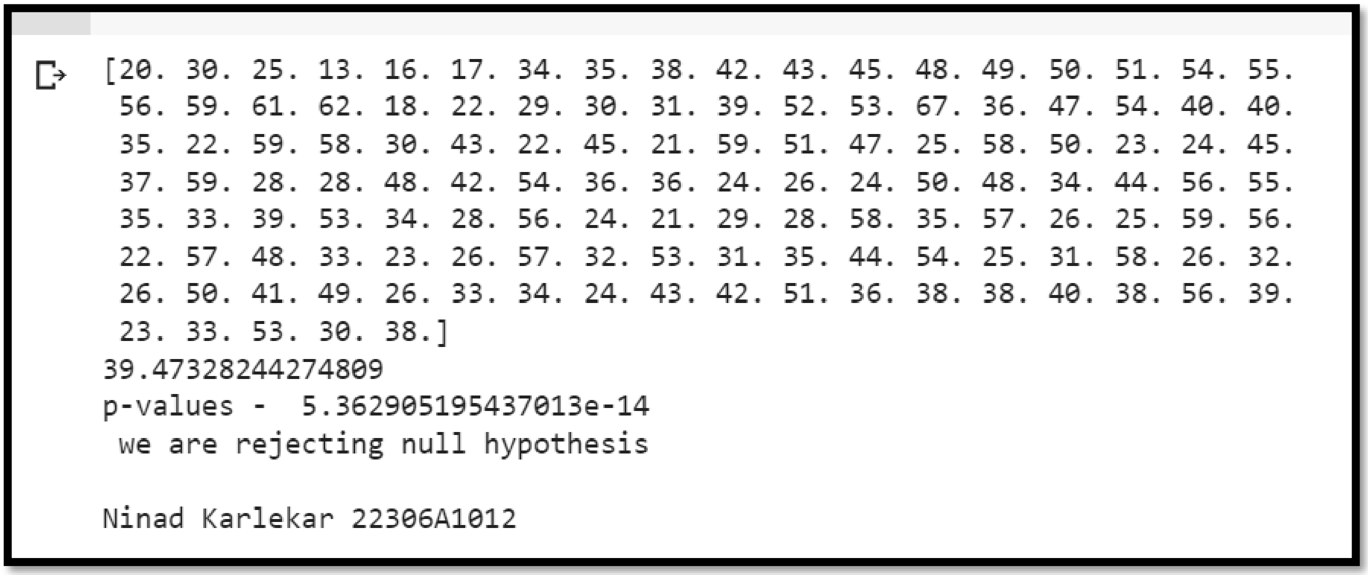
A. Perform testing of hypothesis using one sample t-test.

Description:-

One sample t-test : The One Sample t Test determines whether the sample mean is statistically different from a known or hypothesised population mean. The One Sample t Test is a parametric test.

Code:

```
from scipy.stats import ttest_1samp
import numpy as np
ages = np.genfromtxt('/content/ages.csv')
print(ages)
ages_mean = np.mean(ages)
print(ages_mean)
tset, pval = ttest_1samp(ages, 30)
print('p-values - ',pval)
if pval< 0.05: # alpha value is 0.05
  print(" we are rejecting null hypothesis")
else:
  print("we are accepting null hypothesis")

print("\nNinad Karlekar 22306A1012")
```

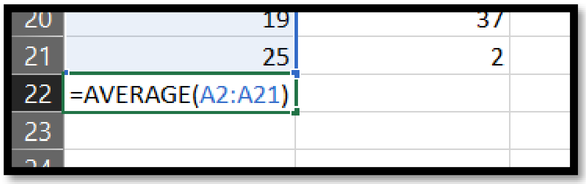


```
[20. 30. 25. 13. 16. 17. 34. 35. 38. 42. 43. 45. 48. 49. 50. 51. 54. 55.
 56. 59. 61. 62. 18. 22. 29. 30. 31. 39. 52. 53. 67. 36. 47. 54. 40. 40.
 35. 22. 59. 58. 30. 43. 22. 45. 21. 59. 51. 47. 25. 58. 50. 23. 24. 45.
 37. 59. 28. 28. 48. 42. 54. 36. 36. 24. 26. 24. 50. 48. 34. 44. 56. 55.
 35. 33. 39. 53. 34. 28. 56. 24. 21. 29. 28. 58. 35. 57. 26. 25. 59. 56.
 22. 57. 48. 33. 23. 26. 57. 32. 53. 31. 35. 44. 54. 25. 31. 58. 26. 32.
 26. 50. 41. 49. 26. 33. 34. 24. 43. 42. 51. 36. 38. 38. 40. 38. 56. 39.
 23. 33. 53. 30. 38.]
39.47328244274809
p-values -  5.362905195437013e-14
 we are rejecting null hypothesis

Ninad Karlekar 22306A1012
```

B. Write a program for t-test comparing two means for independent samples.

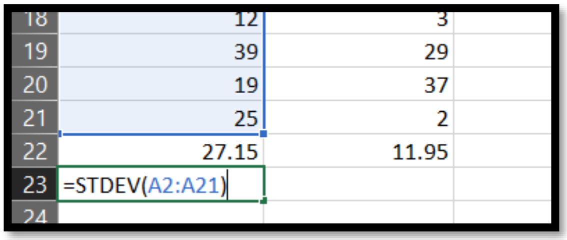
Steps(Excel):-

1. Open Excel file
2. Find the average(mean) of both Experimental and comparison columns



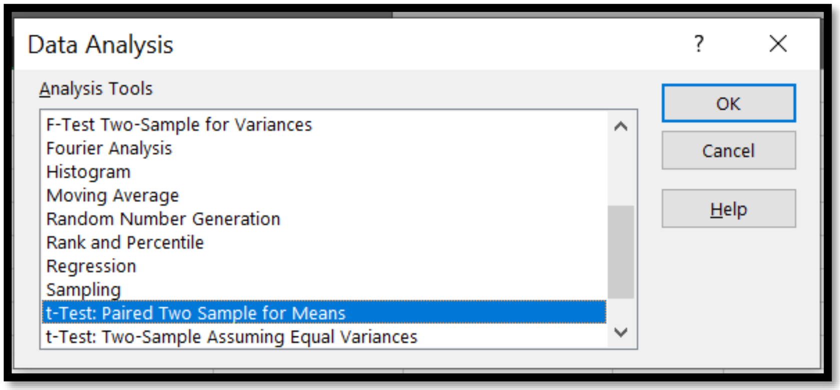
20	19	37
21	25	2
22	=AVERAGE(A2:A21)	
23		

3. Find the Standard deviation of both Experimental and comparison columns



18	12	3
19	39	29
20	19	37
21	25	2
22	27.15	11.95
23	=STDEV(A2:A21)	
24		

4. Go to Data analysis -> Select t-test: Paired Two Sample for Means -> OK

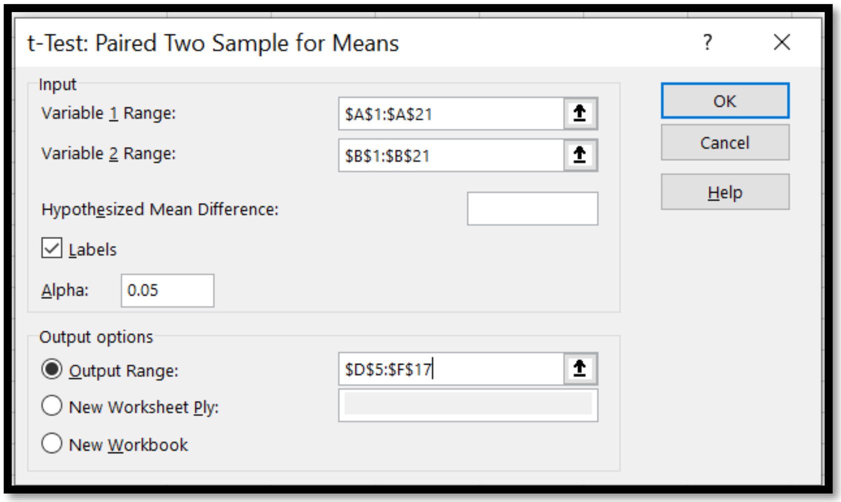


Data Analysis

Analysis Tools

F-Test Two-Sample for Variances
Fourier Analysis
Histogram
Moving Average
Random Number Generation
Rank and Percentile
Regression
Sampling
t-Test: Paired Two Sample for Means
t-Test: Two-Sample Assuming Equal Variances

5. For Variable 1 range(Experimental)= A1 to A21
   For Variable 2 range(Comparison)= B1 to B21
   For Output Range= D5 to F17



t-Test: Paired Two Sample for Means

Input

Variable 1 Range: $A$1:$A$21

Variable 2 Range: $B$1:$B$21

Hypothesized Mean Difference:

☑ Labels

Alpha: 0.05

Output options

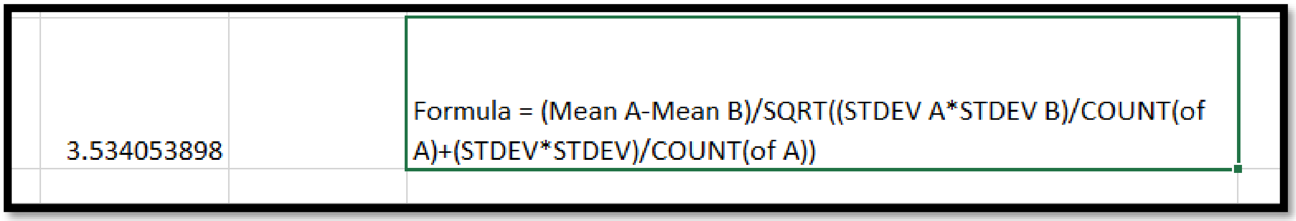
● Output Range: $D$5:$F$17

○ New Worksheet Ply:

○ New Workbook

6. Write 2 Hypothesis
   H0 - Difference in gain score is not likely the result of experiment.
   H1 - Difference in gain score is likely the result of experimental treatment and not the result of change variation

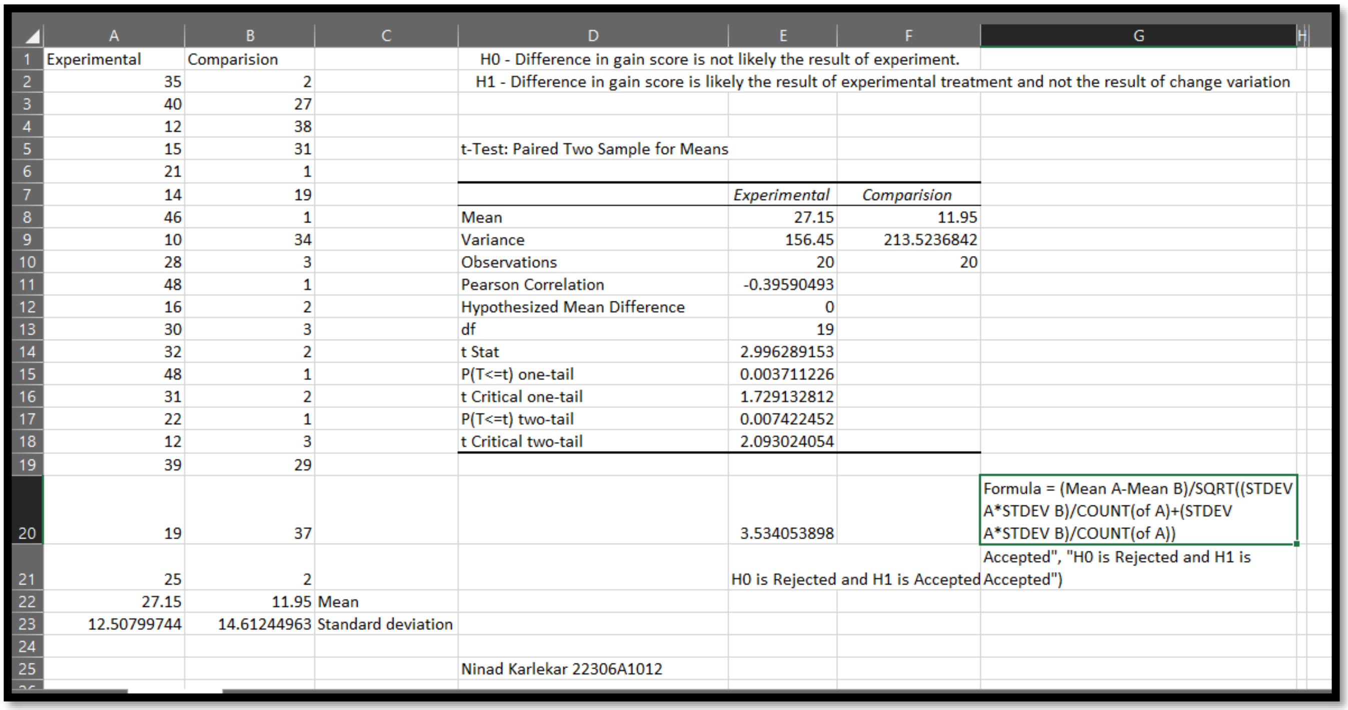
7. To calculate the T-Test square value go to cell E20 and type
   =(A22-B22)/SQRT((A23*A23)/COUNT(A2:A21)+(B23*B23)/COUNT(A2:A21))
   Formula=(Mean A-Mean B)/SQRT((STDEV A*STDEV B)/COUNT(of A) + (STDEV*STDEV)/COUNT(of A))



3.534053898		Formula = (Mean A-Mean B)/SQRT((STDEV A*STDEV B)/COUNT(of A)+(STDEV*STDEV)/COUNT(of A))

8. Now go to cell E21 and type
   =IF(E20<E12,"H0 is Accepted", "H0 is Rejected and H1 is Accepted")

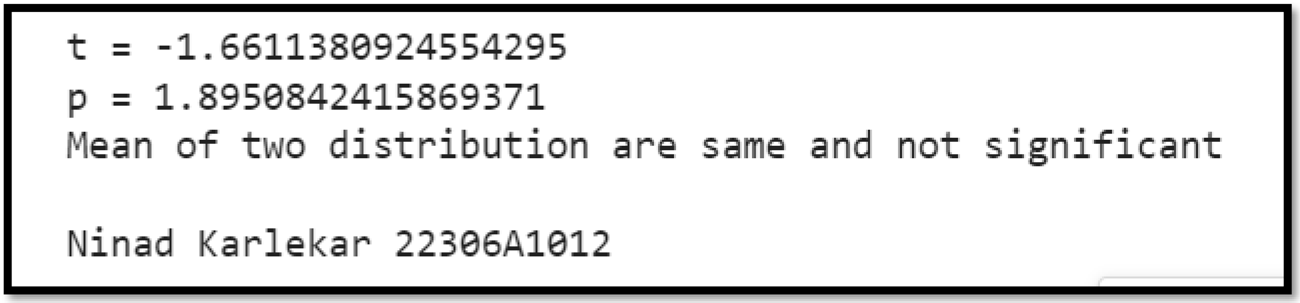
OUTPUT:



	A	B	C	D	E	F	G
1	Experimental	Comparision		H0 - Difference in gain score is not likely the result of experiment.			
2	35	2		H1 - Difference in gain score is likely the result of experimental treatment and not the result of change variation			
3	40	27					
4	12	38					
5	15	31		t-Test: Paired Two Sample for Means			
6	21	1					
7	14	19			*Experimental*	*Comparision*	
8	46	1		Mean	27.15	11.95	
9	10	34		Variance	156.45	213.5236842	
10	28	3		Observations	20	20	
11	48	1		Pearson Correlation	-0.39590493		
12	16	2		Hypothesized Mean Difference	0		
13	30	3		df	19		
14	32	2		t Stat	2.996289153		
15	48	1		P(T<=t) one-tail	0.003711226		
16	31	2		t Critical one-tail	1.729132812		
17	22	1		P(T<=t) two-tail	0.007422452		
18	12	3		t Critical two-tail	2.093024054		
19	39	29					
20	19	37			3.534053898		Formula = (Mean A-Mean B)/SQRT((STDEV A*STDEV B)/COUNT(of A)+(STDEV A*STDEV B)/COUNT(of A))
21	25	2			H0 is Rejected and H1 is Accepted		Accepted", "H0 is Rejected and H1 is Accepted")
22	27.15	11.95	Mean				
23	12.50799744	14.61244963	Standard deviation				
24							
25				Ninad Karlekar 22306A1012			

PYTHON:CODE:

```
import numpy as np
from scipy import stats
from numpy.random import randn
N = 20
a = 5 * randn(100) + 50
b = 5 * randn(100) + 51
var_a = a.var(ddof=1)
var_b = b.var(ddof=1)
s = np.sqrt((var_a + var_b)/2)
t = (a.mean() - b.mean())/(s*np.sqrt(2/N))
df = 2*N - 2
#p-value after comparison with the t
p = 1 - stats.t.cdf(t,df=df)
print("t = " + str(t))
print("p = " + str(2*p))
if t> p :
  print('Mean of two distribution are differnt and significant')
else:
  print('Mean of two distribution are same and not significant')
print('\nNinad Karlekar 22306A1012')
```



```
t = -1.6611380924554295
p = 1.8950842415869371
Mean of two distribution are same and not significant

Ninad Karlekar 22306A1012
```

C. Perform testing of hypothesis using paired t-test.

The paired sample t-test is also called dependent sample t-test. It’s an univariate test that tests for a significant difference between 2 related variables. An example of this is if you where to collect the blood pressure for an individual before and after some treatment, condition, or time point. The data set contains blood pressure readings before and after an intervention. These are variables “bp_before” and “bp_after”.

The hypothesis being test is:

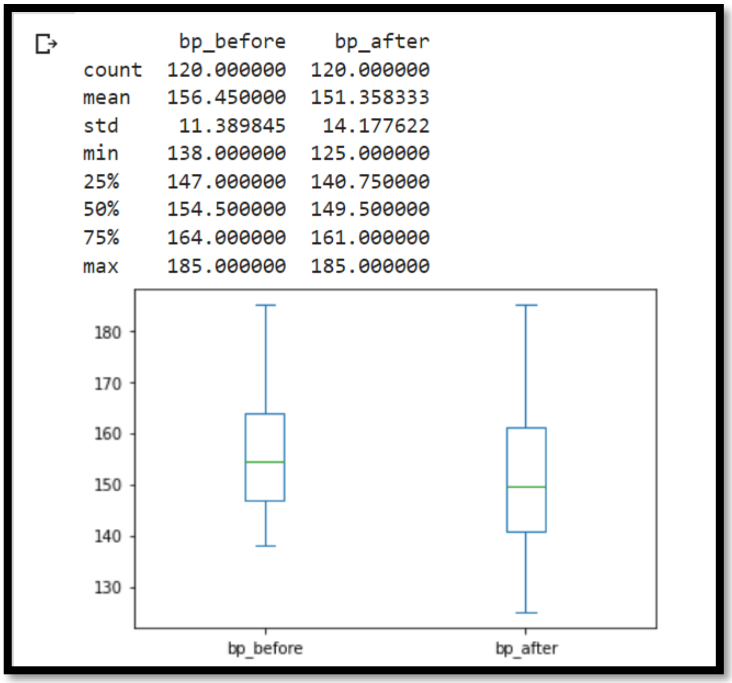
- H0 - The mean difference between sample 1 and sample 2 is equal to 0.
- H0 - The mean difference between sample 1 and sample 2 is not equal to 0.

Code & Output:

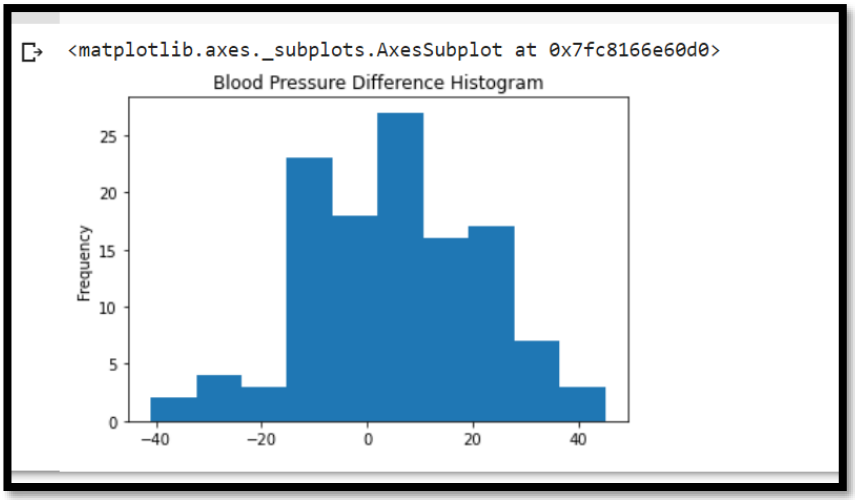
```
from scipy import stats
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("blood_pressure.csv")
print(df[['bp_before','bp_after']].describe())
#First let’s check for any significant outliers in
#each of the variables.
```

```
df[['bp_before', 'bp_after']].plot(kind='box')
# This saves the plot as a png file
plt.savefig('boxplot_outliers.png')
##################**###############################
# make a histogram to differences between the two scores.
df['bp_difference'] = df['bp_before'] - df['bp_after']
df['bp_difference'].plot(kind='hist', title= 'Blood Pressure Difference Histogram')
#Again, this saves the plot as a png file
##################**###############################
plt.savefig('blood pressure difference histogram.png')
stats.probplot(df['bp_difference'], plot= plt)
plt.title('Blood pressure Difference Q-Q Plot')
plt.savefig('blood pressure difference qq plot.png')
stats.shapiro(df['bp_difference'])
stats.ttest_rel(df['bp_before'], df['bp_after'])
```

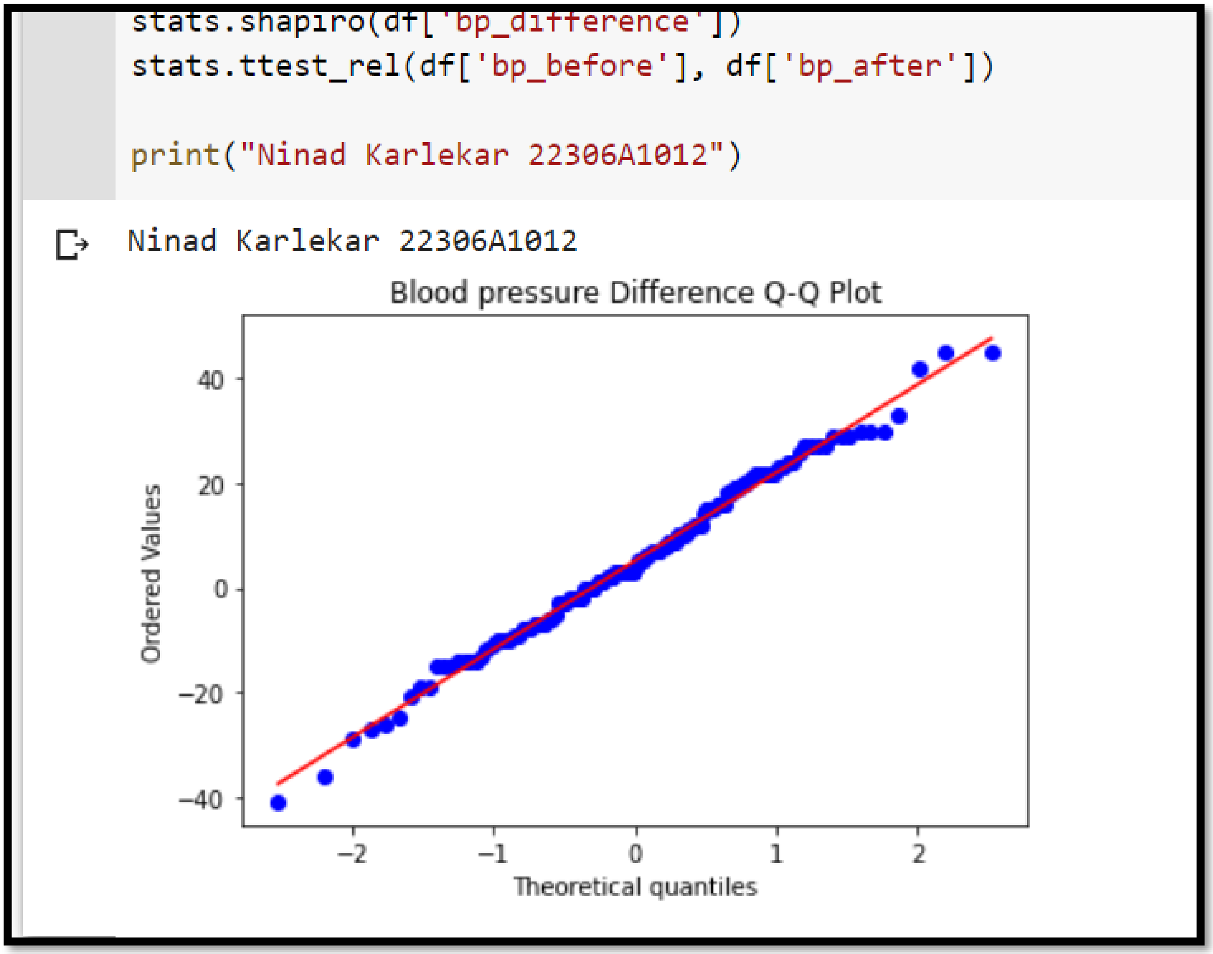
OUTPUT:



```
        bp_before   bp_after
count  120.000000  120.000000
mean   156.450000  151.358333
std     11.389845   14.177622
min    138.000000  125.000000
25%    147.000000  140.750000
50%    154.500000  149.500000
75%    164.000000  161.000000
max    185.000000  185.000000
```



```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc8166e60d0>
```



```
stats.shapiro(df['bp_difference'])
stats.ttest_rel(df['bp_before'], df['bp_after'])

print("Ninad Karlekar 22306A1012")
```

```
Ninad Karlekar 22306A1012
```

Reject Null Hypothesis
A paired sample t-test was used to analyse the blood pressure before and after the intervention to test if the intervention had a significant affect on the blood pressure.
The blood pressure before the intervention was higher (156.45 ± 11.39 units) compared to the blood pressure post intervention (151.36 ± 14.18 units); t
here was a statistically significant decrease in blood pressure (t(119)=3.34, p= 0.0011) of 5.09 units.

VSIT

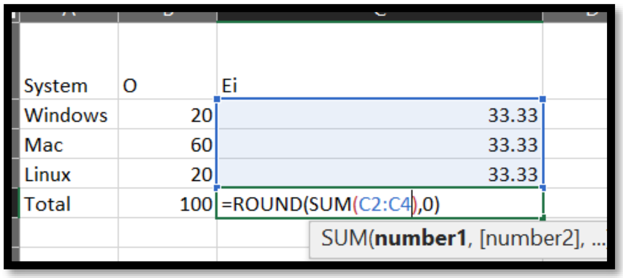
Research in Computing
Practical # 4

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	chi-squared	**Batch**	1

A. Perform testing of hypothesis using chi-squared goodness-of-fit test.

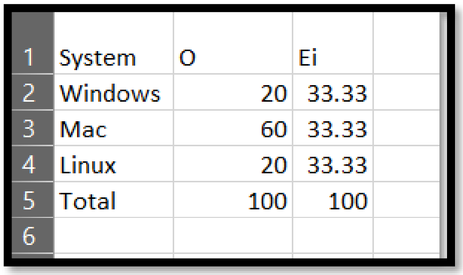
STEPS(EXCEL):-

1. Find total of both columns



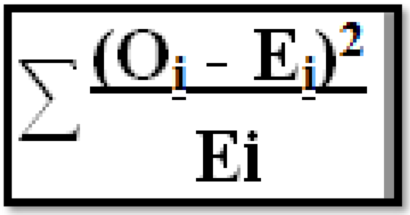
System	O	Ei
Windows	20	33.33
Mac	60	33.33
Linux	20	33.33
Total	100	=ROUND(SUM(C2:C4),0)

SUM(**number1**, [number2], ...

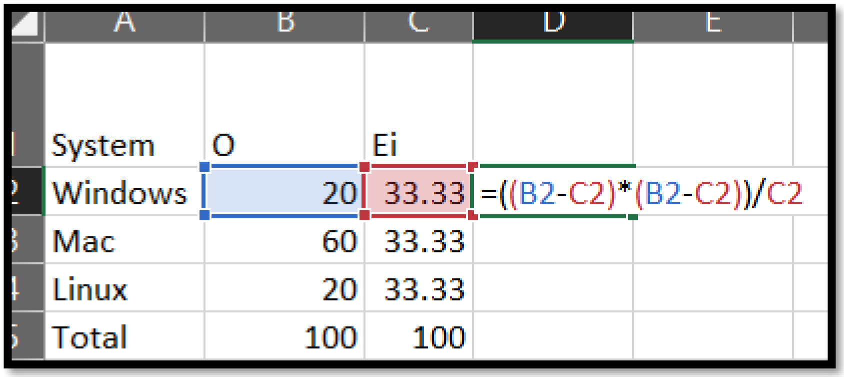


1	System	O	Ei
2	Windows	20	33.33
3	Mac	60	33.33
4	Linux	20	33.33
5	Total	100	100
6			

2. Formula

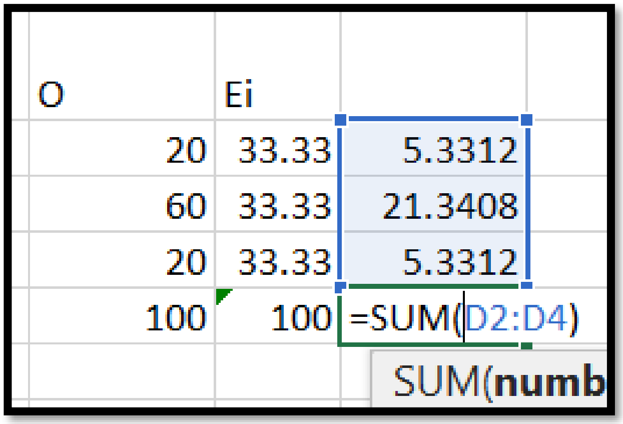


$$\sum \frac{(O_i - E_i)^2}{Ei}$$



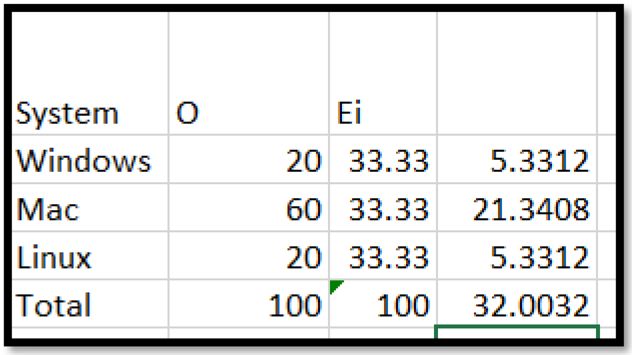
	A	B	C	D	E
1	System	O	Ei		
2	Windows	20	33.33	=((B2-C2)*(B2-C2))/C2	
3	Mac	60	33.33		
4	Linux	20	33.33		
5	Total	100	100		

3. Find the sum of all



O	Ei	
20	33.33	5.3312
60	33.33	21.3408
20	33.33	5.3312
100	100	=SUM(D2:D4)

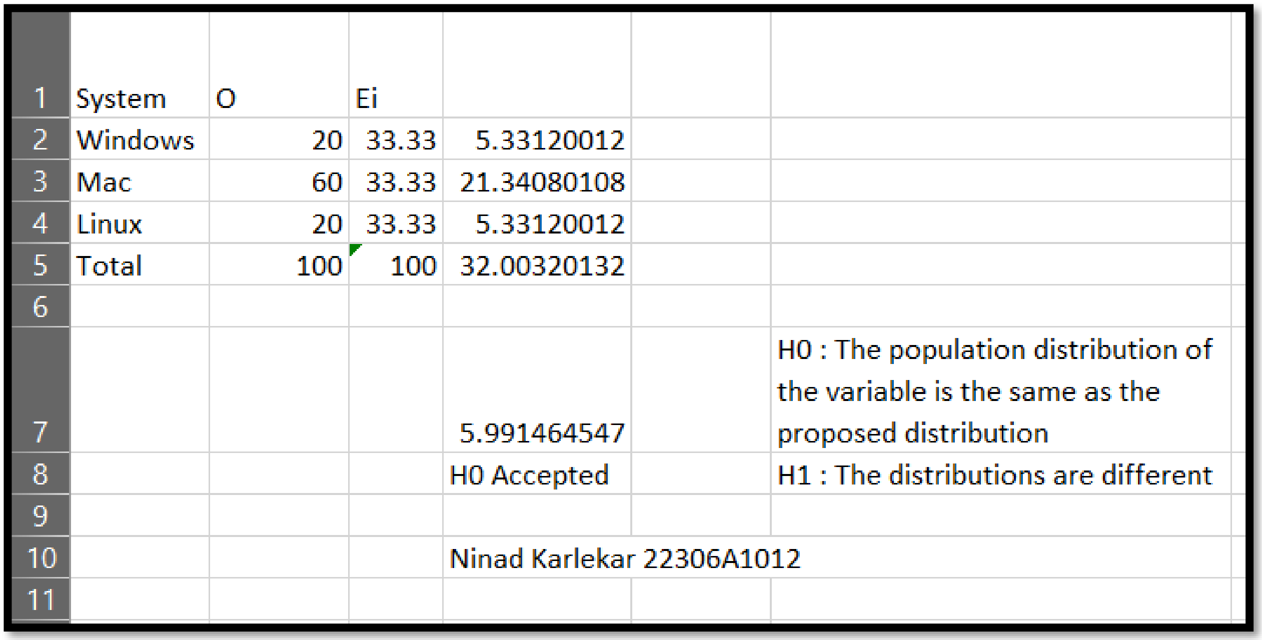
SUM(**numb**



System	O	Ei	
Windows	20	33.33	5.3312
Mac	60	33.33	21.3408
Linux	20	33.33	5.3312
Total	100	100	32.0032

4. At cell D8 type =IF(D5>D7, "H0 Accepted","H0 Rejected")

OUTPUT:

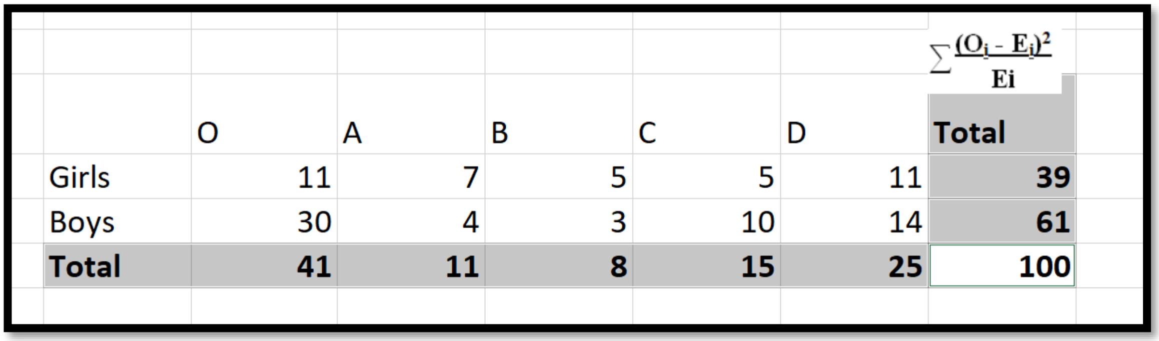


1	System	O	Ei			
2	Windows	20	33.33	5.33120012		
3	Mac	60	33.33	21.34080108		
4	Linux	20	33.33	5.33120012		
5	Total	100	100	32.00320132		
6						
7				5.991464547		H0 : The population distribution of the variable is the same as the proposed distribution
8				H0 Accepted		H1 : The distributions are different
9						
10				Ninad Karlekar 22306A1012		
11						

B. Perform testing of hypothesis using chi-squared test of independence.

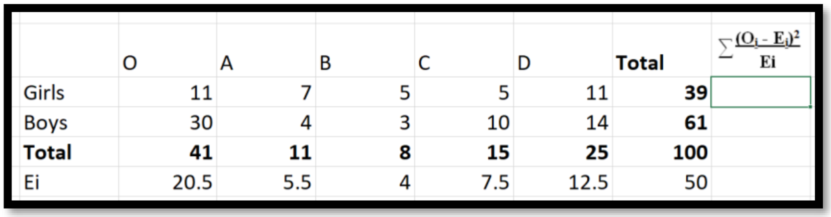
Steps:

1. Find the total for all columns and rows



						$\sum \frac{(O_i - E_i)^2}{Ei}$
	O	A	B	C	D	**Total**
Girls	11	7	5	5	11	**39**
Boys	30	4	3	10	14	**61**
Total	**41**	**11**	**8**	**15**	**25**	**100**

2. To calculate the expected value Ei
   Go to Cell N9 and type =N8/2
   Go to Cell O9 and type =O8/2
   Go to Cell P9 and type =P8/2
   Go to Cell Q9 and type =Q8/2
   Go to Cell R9 and type =R8/2

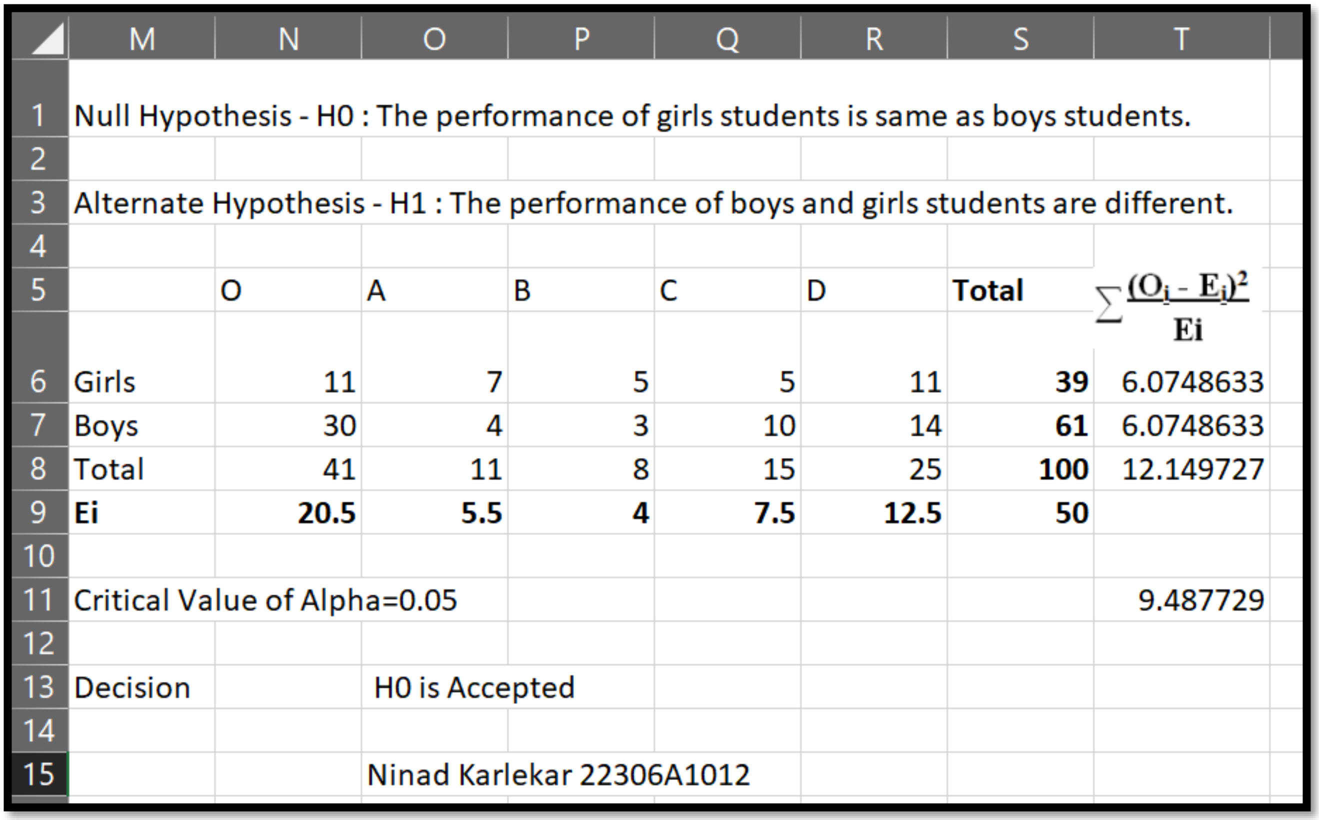


	O	A	B	C	D	Total	$\sum \frac{(O_i - E_i)^2}{Ei}$
Girls	11	7	5	5	11	**39**	
Boys	30	4	3	10	14	**61**	
Total	**41**	**11**	**8**	**15**	**25**	**100**	
Ei	20.5	5.5	4	7.5	12.5	50	

3. Now Calculate $\sum\frac{(O_i - E_i)^2}{Ei}$

Go to cell **T6** and type
=SUM((N6-$N$9)^2/$N$9,(O6-$O$9)^2/$O$9,(P6-$P$9)^2/$P$9,(Q6-Q$9)^2/$Q$9, (R6-$R$9)^2/$R$9)
Go to cell **T7** and type
=SUM((N7-$N$9)^2/$N$9,(O7-$O$9)^2/$O$9,(P7-$P$9)^2/$P$9,(Q7-Q$9)^2/$Q$9, (R7-$R$9)^2/$R$9)

To get the table value go to cell T11 and type =CHIINV(0.05,4)
Go to cell O13 and type =IF(T8>=T11," H0 is Accepted", "H0 is Rejected")



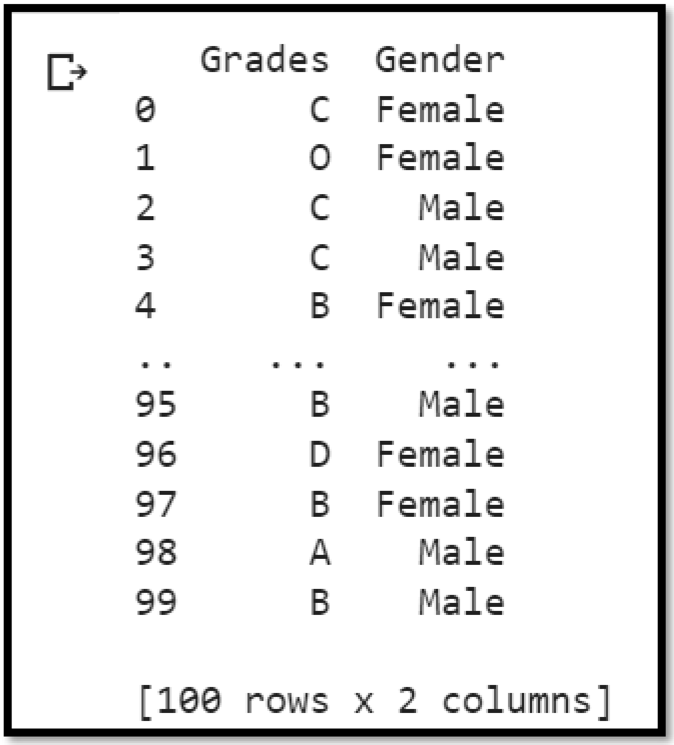
	M	N	O	P	Q	R	S	T
1	Null Hypothesis - H0 : The performance of girls students is same as boys students.							
2								
3	Alternate Hypothesis - H1 : The performance of boys and girls students are different.							
4								
5		O	A	B	C	D	**Total**	$\sum\frac{(O_i - E_i)^2}{Ei}$
6	Girls	11	7	5	5	11	**39**	6.0748633
7	Boys	30	4	3	10	14	**61**	6.0748633
8	Total	41	11	8	15	25	**100**	12.149727
9	**Ei**	**20.5**	**5.5**	**4**	**7.5**	**12.5**	**50**	
10								
11	Critical Value of Alpha=0.05							9.487729
12								
13	Decision		H0 is Accepted					
14								
15			Ninad Karlekar 22306A1012					

Using Python

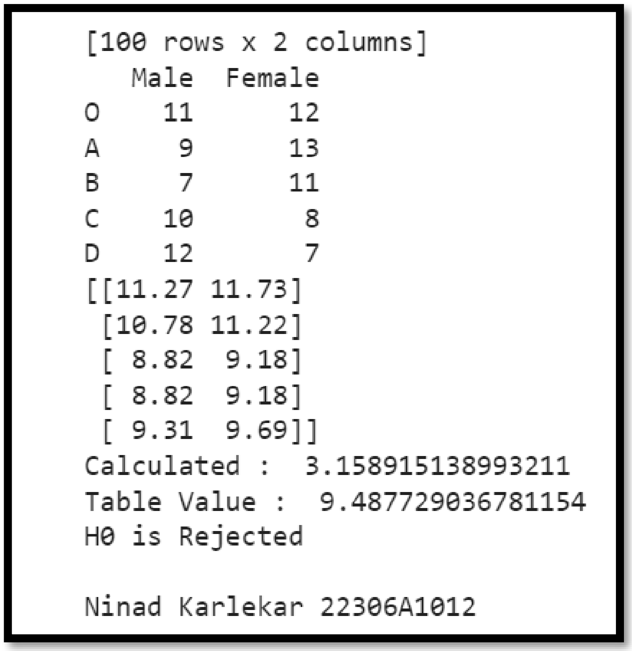
Code:

```
import numpy as np
import pandas as pd
import scipy.stats as stats
np.random.seed(10)
stud_grade = np.random.choice(a=["O","A","B","C","D"],
p=[0.20, 0.20 ,0.20, 0.20, 0.20], size=100)
stud_gen = np.random.choice(a=["Male","Female"], p=[0.5, 0.5], size=100)
mscpart1 = pd.DataFrame({"Grades":stud_grade, "Gender":stud_gen})
print(mscpart1)
stud_tab = pd.crosstab(mscpart1.Grades, mscpart1.Gender, margins=True)
stud_tab.columns = ["Male", "Female", "row_totals"]
stud_tab.index = ["O", "A", "B", "C", "D", "col_totals"]
observed = stud_tab.iloc[0:5, 0:2 ]
print(observed)
```

```
expected = np.outer(stud_tab["row_totals"][0:5], stud_tab.loc["col_totals"][0:2]) / 100
print(expected)
chi_squared_stat = (((observed-expected)**2)/expected).sum().sum()
print('Calculated : ',chi_squared_stat)
crit = stats.chi2.ppf(q=0.95, df=4)
print('Table Value : ',crit)
if chi_squared_stat>= crit:
  print('H0 is Accepted ')
else:
  print('H0 is Rejected ')
print("\nNinad Karlekar 22306A1012")
```



```
     Grades  Gender
0         C  Female
1         O  Female
2         C    Male
3         C    Male
4         B  Female
..      ...     ...
95        B    Male
96        D  Female
97        B  Female
98        A    Male
99        B    Male

[100 rows x 2 columns]
```



```
[100 rows x 2 columns]
   Male  Female
O    11      12
A     9      13
B     7      11
C    10       8
D    12       7
[[11.27 11.73]
 [10.78 11.22]
 [ 8.82  9.18]
 [ 8.82  9.18]
 [ 9.31  9.69]]
Calculated :  3.158915138993211
Table Value :  9.487729036781154
H0 is Rejected

Ninad Karlekar 22306A1012
```

Research in Computing
Practical # 5

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	Perform testing of hypothesis using Z-test.	**Batch**	1

A. Perform testing of hypothesis using Z-test.

Description:-

Define Hypothesis:
Hypothesis is a strong, short statement that forms the basis of your research.
The purpose of Hypothesis is to predict the findings, conclusions and data. It is a educated guess based on your observation and Environment around you.
Define Null hypothesis:
Null hypothesis is a type of hypothesis that is presumed to be true until it is invalidated by testing.
What is hypothesis Testing?
Hypothesis testing provides a way to verify whether the results of an experiment are valid. It is a type of tool
What is Z Test?
Z test is a statistical test that is conducted on data that approximately follows a normal distribution. The z test can be performed on one sample, two samples, or on proportions for hypothesis testing. It checks if the means of two large samples are different or not when the population variance is known.

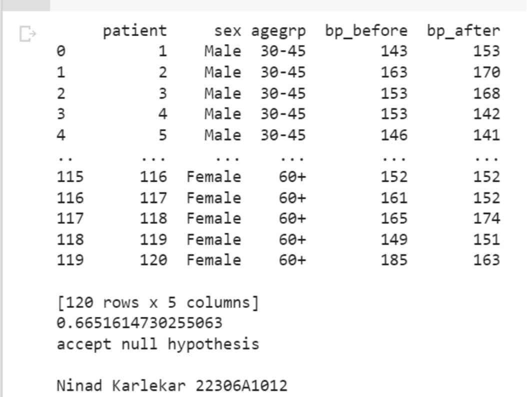
Procedure:-
Use a Z test if:

- Your sample size is greater than 30. Otherwise, use a t test.
- Data points should be independent from each other. In other words, one data point isn't related or doesn't affect another data point.
- Your data should be normally distributed. However, for large sample sizes (over 30) this doesn't always matter.
- Your data should be randomly selected from a population, where each item has an equal chance of being selected.
- Sample sizes should be equal if at all possible.

H0 - Blood pressure has a mean of 156 units
Dataset- blood_pressure.csv

Code:-

```
from statsmodels.stats import weightstats as stests
import pandas as pd
from scipy import stats
df = pd.read_csv("blood_pressure.csv")
df[['bp_before','bp_after']].describe()
print(df)
ztest ,pval = stests.ztest(df['bp_before'], x2=None, value=156)
print(float(pval))
if pval<0.05:
  print("reject null hypothesis")
else:
  print("accept null hypothesis")
print("\nNinad Karlekar 22306A1012")
```



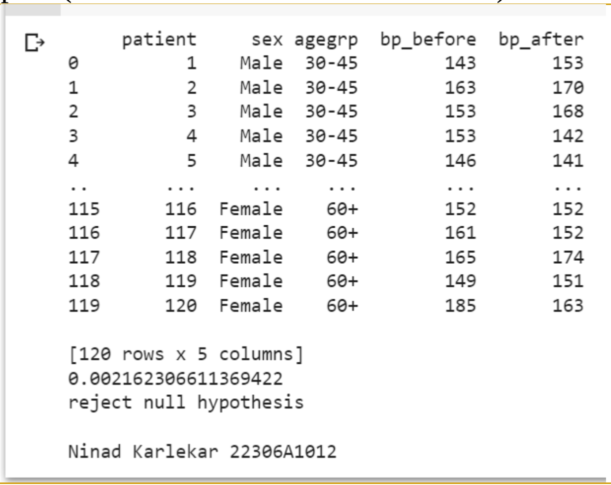
```
     patient     sex agegrp  bp_before  bp_after
0          1    Male  30-45        143       153
1          2    Male  30-45        163       170
2          3    Male  30-45        153       168
3          4    Male  30-45        153       142
4          5    Male  30-45        146       141
..       ...     ...    ...        ...       ...
115      116  Female    60+        152       152
116      117  Female    60+        161       152
117      118  Female    60+        165       174
118      119  Female    60+        149       151
119      120  Female    60+        185       163

[120 rows x 5 columns]
0.6651614730255063
accept null hypothesis

Ninad Karlekar 22306A1012
```

B. Two-Sample Z test

Two-sample Z test - In two sample z-test , similar to t-test here we are checking two independent data groups and deciding whether sample mean of two group is equal or not.
H0 : Mean of two group is 0
H1 : Mean of two group is not 0

Code:-

```
import pandas as pd
from statsmodels.stats import weightstats as stests
df = pd.read_csv("blood_pressure.csv")
df[['bp_before','bp_after']].describe()
print(df)
ztest ,pval = stests.ztest(df['bp_before'], x2=df['bp_after'], value=0,alternative='two-sided')
print(float(pval))
if pval<0.05:
  print("reject null hypothesis")
else:
  print("accept null hypothesis")
print("\nNinad Karlekar 22306A1012")
```



```
     patient     sex agegrp  bp_before  bp_after
0          1    Male  30-45        143       153
1          2    Male  30-45        163       170
2          3    Male  30-45        153       168
3          4    Male  30-45        153       142
4          5    Male  30-45        146       141
..       ...     ...    ...        ...       ...
115      116  Female    60+        152       152
116      117  Female    60+        161       152
117      118  Female    60+        165       174
118      119  Female    60+        149       151
119      120  Female    60+        185       163

[120 rows x 5 columns]
0.002162306611369422
reject null hypothesis

Ninad Karlekar 22306A1012
```

VSIT

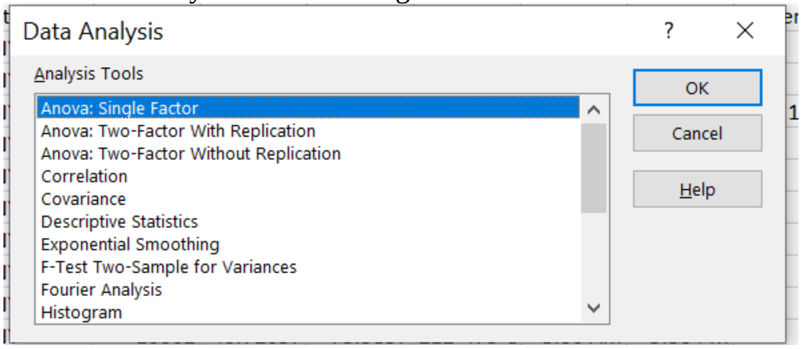
Research in Computing
Practical # 6

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	Hypothesis using ANOVA	**Batch**	1

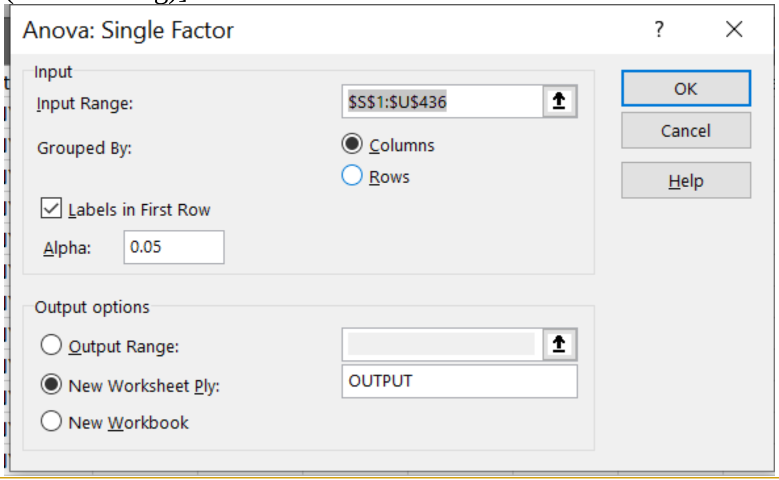
A. Perform testing of hypothesis using one-way ANOVA

Steps(EXCEL):

1. Open scores.csv file
2. Go to Data analysis -> Anova single factor -> ok



3. Select input range as all values from [Average Score (SAT Math), Average Score (SAT Reading), Average Score (SAT Writing)] columns



Input

Input Range: $S$1:$U$436

Grouped By: ● Columns ○ Rows

☑ Labels in First Row

Alpha: 0.05

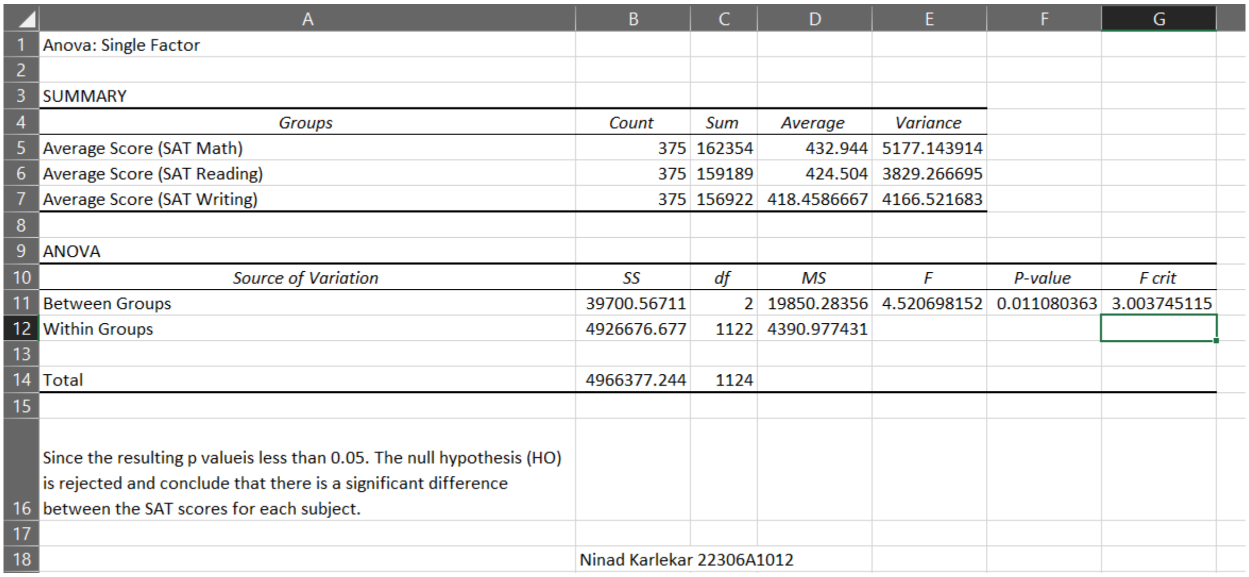
Output options

○ Output Range:

● New Worksheet Ply: OUTPUT

○ New Workbook

4. OUTPUT



	A	B	C	D	E	F	G
1	Anova: Single Factor						
2							
3	SUMMARY						
4	*Groups*	*Count*	*Sum*	*Average*	*Variance*		
5	Average Score (SAT Math)	375	162354	432.944	5177.143914		
6	Average Score (SAT Reading)	375	159189	424.504	3829.266695		
7	Average Score (SAT Writing)	375	156922	418.4586667	4166.521683		
8							
9	ANOVA						
10	*Source of Variation*	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
11	Between Groups	39700.56711	2	19850.28356	4.520698152	0.011080363	3.003745115
12	Within Groups	4926676.677	1122	4390.977431			
13							
14	Total	4966377.244	1124				
15							
16	Since the resulting p valueis less than 0.05. The null hypothesis (HO) is rejected and conclude that there is a significant difference between the SAT scores for each subject.						
17							
18		Ninad Karlekar 22306A1012					

B. Perform testing of hypothesis using two-way ANOVA.

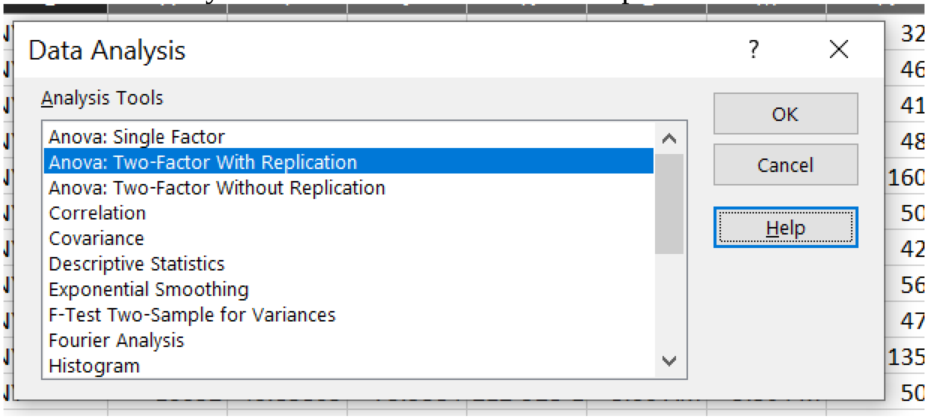
Description:

ANOVA (Analysis of Variance) is a statistical test used to analyses the difference between the means of more than two groups.

A two-way ANOVA is used to estimate how the mean of a quantitative variable changes according to the levels of two categorical variables. Use a two-way ANOVA when you want to know how two independent variables, in combination, affect a dependent variable.

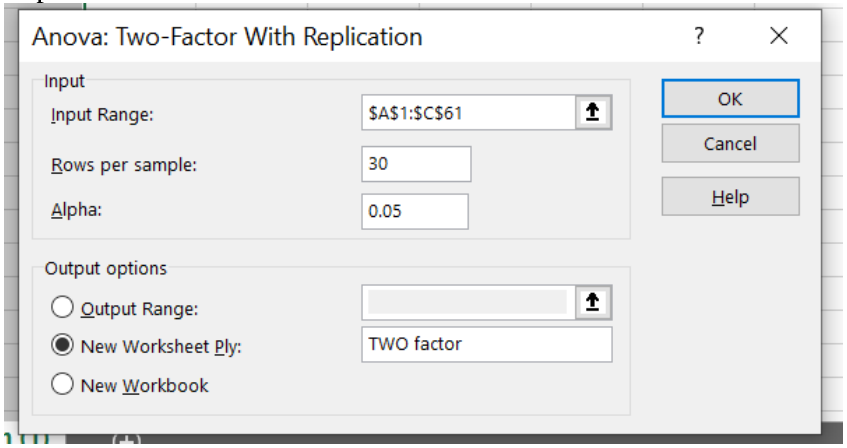
Steps

1. Open ToothGrowth.csv file
2. Go to Data analysis -> Anova two factor with replication-> ok



3. Select all cell in input range ,
   Rows per sample=30

Alpha=0.05



Anova: Two-Factor With Replication ? ×

Input

Input Range: $A$1:$C$61

Rows per sample: 30

Alpha: 0.05

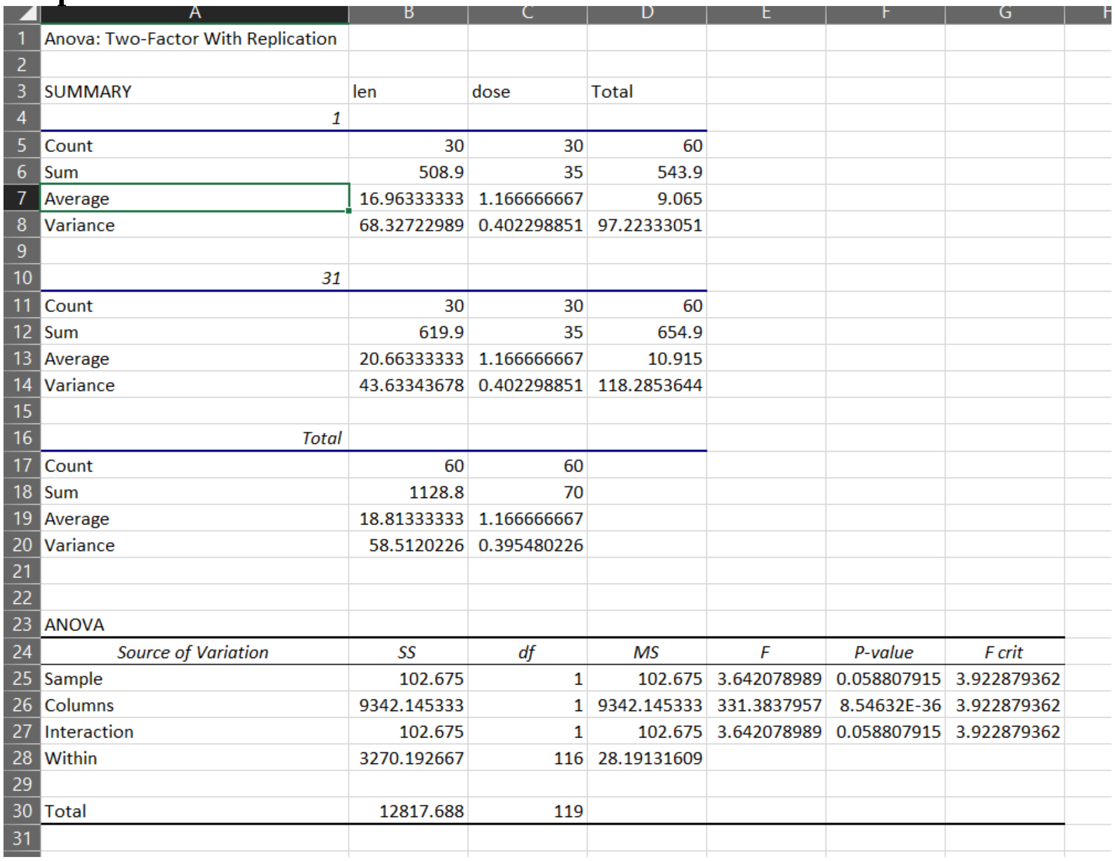
Output options

○ Output Range:

● New Worksheet Ply: TWO factor

○ New Workbook

Output:



	A	B	C	D	E	F	G
1	Anova: Two-Factor With Replication						
2							
3	SUMMARY	len	dose	Total			
4	*1*						
5	Count	30	30	60			
6	Sum	508.9	35	543.9			
7	Average	16.96333333	1.166666667	9.065			
8	Variance	68.32722989	0.402298851	97.22333051			
9							
10	*31*						
11	Count	30	30	60			
12	Sum	619.9	35	654.9			
13	Average	20.66333333	1.166666667	10.915			
14	Variance	43.63343678	0.402298851	118.2853644			
15							
16	*Total*						
17	Count	60	60				
18	Sum	1128.8	70				
19	Average	18.81333333	1.166666667				
20	Variance	58.5120226	0.395480226				
21							
22							
23	ANOVA						
24	*Source of Variation*	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
25	Sample	102.675	1	102.675	3.642078989	0.058807915	3.922879362
26	Columns	9342.145333	1	9342.145333	331.3837957	8.54632E-36	3.922879362
27	Interaction	102.675	1	102.675	3.642078989	0.058807915	3.922879362
28	Within	3270.192667	116	28.19131609			
29							
30	Total	12817.688	119				
31							

C. Perform testing of hypothesis using multivariate ANOVA (MANOVA)

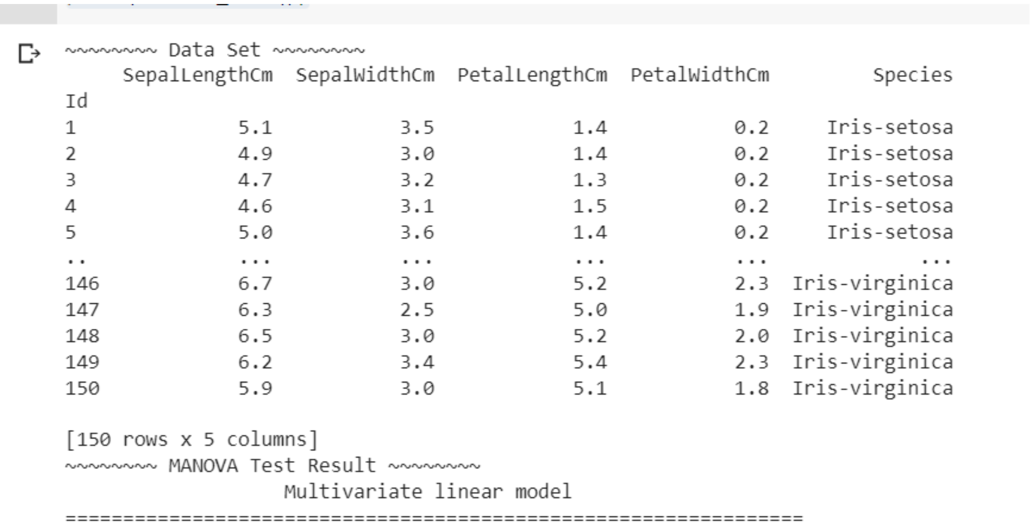
Description:

The Multivariate analysis of variance (MANOVA) procedure provides regression analysis and analysis of variance for multiple dependent variables by one or more factor variables or covariates. The factor variables divide the population into groups. Using this general linear model procedure, you can test null hypotheses about the effects of factor variables on the means of various groupings of a joint distribution of dependent variables. You can investigate interactions between factors as well as the effects of individual factors. In addition, the effects of covariates and covariate interactions with factors can be included. For regression analysis, the independent (predictor) variables are specified as covariates.

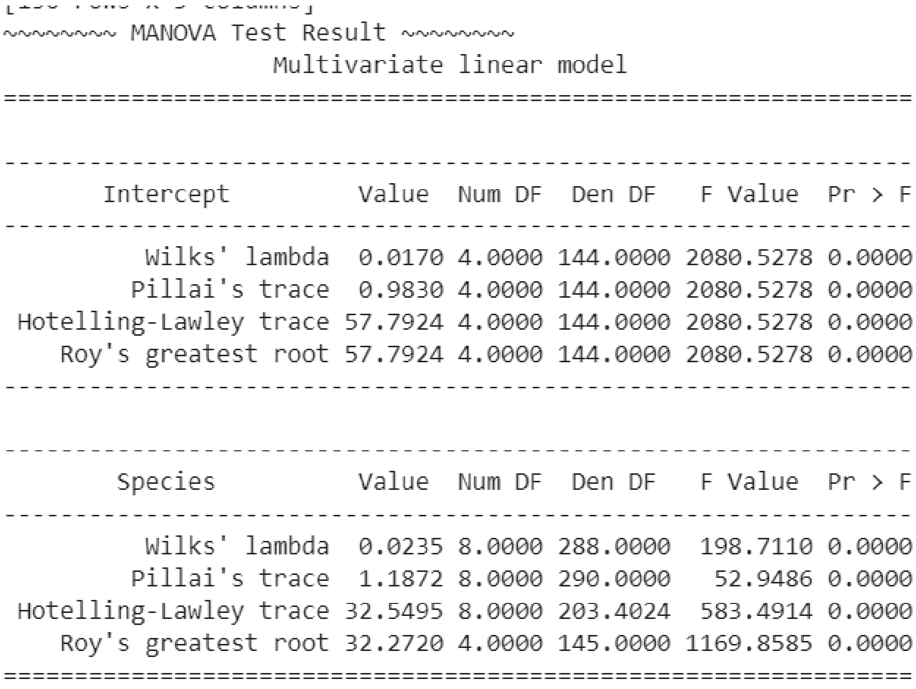
PYTHON CODE:

```
import pandas as pd
from statsmodels.multivariate.manova import MANOVA
df = pd.read_csv('Iris.csv', index_col=0)
df.columns = df.columns.str.replace(".", "_")
df.head()
print('~~~~~~~~ Data Set ~~~~~~~~')
print(df)
maov = MANOVA.from_formula('SepalLengthCm + SepalWidthCm + \
PetalLengthCm + PetalWidthCm ~ Species', data=df)
print('~~~~~~~~ MANOVA Test Result ~~~~~~~~')
print(maov.mv_test())
```

OUTPUT:



```
~~~~~~~~ Data Set ~~~~~~~~
     SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm         Species
Id
1              5.1           3.5            1.4           0.2     Iris-setosa
2              4.9           3.0            1.4           0.2     Iris-setosa
3              4.7           3.2            1.3           0.2     Iris-setosa
4              4.6           3.1            1.5           0.2     Iris-setosa
5              5.0           3.6            1.4           0.2     Iris-setosa
..             ...           ...            ...           ...             ...
146            6.7           3.0            5.2           2.3  Iris-virginica
147            6.3           2.5            5.0           1.9  Iris-virginica
148            6.5           3.0            5.2           2.0  Iris-virginica
149            6.2           3.4            5.4           2.3  Iris-virginica
150            5.9           3.0            5.1           1.8  Iris-virginica

[150 rows x 5 columns]
~~~~~~~~ MANOVA Test Result ~~~~~~~~
                  Multivariate linear model
================================================================
```



```
~~~~~~~~ MANOVA Test Result ~~~~~~~~
                  Multivariate linear model
================================================================
```

```

----------------------------------------------------------------
       Intercept         Value  Num DF  Den DF   F Value  Pr > F
----------------------------------------------------------------
          Wilks' lambda  0.0170 4.0000 144.0000 2080.5278 0.0000
         Pillai's trace  0.9830 4.0000 144.0000 2080.5278 0.0000
 Hotelling-Lawley trace 57.7924 4.0000 144.0000 2080.5278 0.0000
    Roy's greatest root 57.7924 4.0000 144.0000 2080.5278 0.0000
----------------------------------------------------------------

----------------------------------------------------------------
        Species          Value  Num DF  Den DF   F Value  Pr > F
----------------------------------------------------------------
          Wilks' lambda  0.0235 8.0000 288.0000  198.7110 0.0000
         Pillai's trace  1.1872 8.0000 290.0000   52.9486 0.0000
 Hotelling-Lawley trace 32.5495 8.0000 203.4024  583.4914 0.0000
    Roy's greatest root 32.2720 4.0000 145.0000 1169.8585 0.0000
================================================================
```

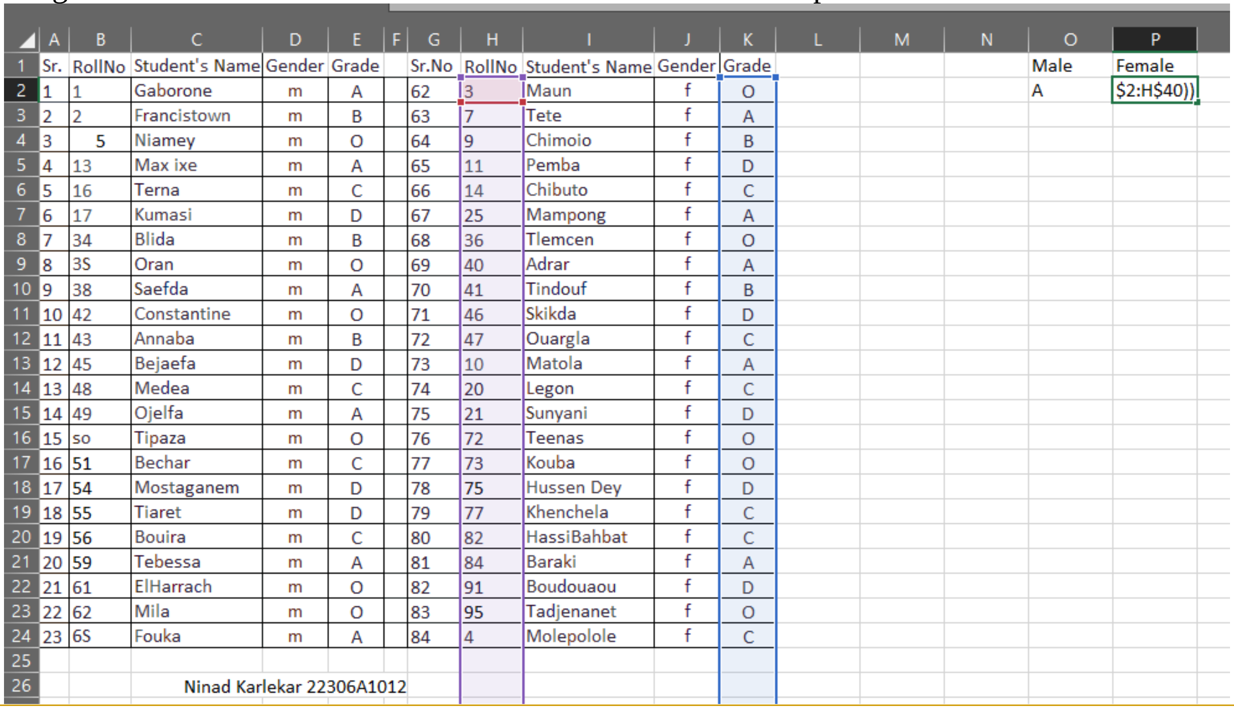
Research in Computing
Practical # 7

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	Perform the Random sampling | Perform the Stratified sampling	**Batch**	1

A. Perform the Random sampling for the given data and analyse it.

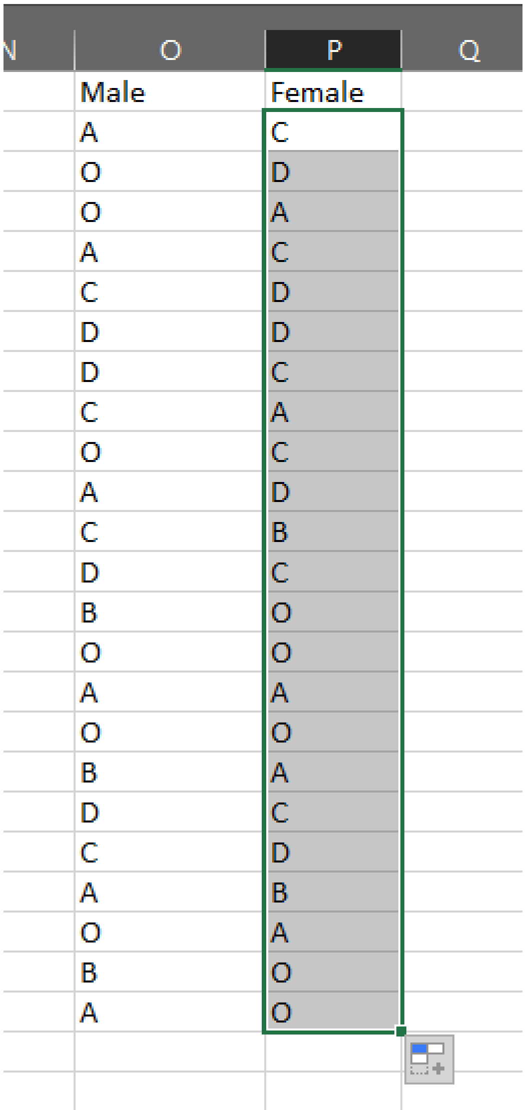
Example 1: From a population of 10 women and 10 men as given in the table in Figure 1 on the left below, create a random sample of 6 people for Group 1 and a periodic sample consisting of every 3rd woman for Group 2.
You need to run the sampling data analysis tool twice, once to create Group 1 and again to create Group 2. For Group 1 you select all 20 population cells as the Input Range and Random as the Sampling Method with 6 for the Random Number of Samples. For Group 2 you select the 10 cells in the Women column as Input Range and Periodic with Period 3.

1. Open existing excel sheet with population data
   Sample Sheet looks as given below:
   Set Cell O1 = Male and Cell O2 = Female
2. To generate a random sample for male students from given population go to Cell O1 and type
   =INDEX(E$2:E$62,RANK(B2,B$2:B$62))
   Drag the formula to the desired no of cell to select random sample.
3. Now, to generate a random sample for female students go to cell P1 and type
   =INDEX(K$2:K$40,RANK(H2,H$2:H$40))
   Drag the formula to the desired no of cell to select random sample.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Sr.	RollNo	Student's Name	Gender	Grade		Sr.No	RollNo	Student's Name	Gender	Grade				Male	Female
2	1	1	Gaborone	m	A		62	3	Maun	f	O				A	$2:H$40))
3	2	2	Francistown	m	B		63	7	Tete	f	A					
4	3	5	Niamey	m	O		64	9	Chimoio	f	B					
5	4	13	Max ixe	m	A		65	11	Pemba	f	D					
6	5	16	Terna	m	C		66	14	Chibuto	f	C					
7	6	17	Kumasi	m	D		67	25	Mampong	f	A					
8	7	34	Blida	m	B		68	36	Tlemcen	f	O					
9	8	3S	Oran	m	O		69	40	Adrar	f	A					
10	9	38	Saefda	m	A		70	41	Tindouf	f	B					
11	10	42	Constantine	m	O		71	46	Skikda	f	D					
12	11	43	Annaba	m	B		72	47	Ouargla	f	C					
13	12	45	Bejaefa	m	D		73	10	Matola	f	A					
14	13	48	Medea	m	C		74	20	Legon	f	C					
15	14	49	Ojelfa	m	A		75	21	Sunyani	f	D					
16	15	so	Tipaza	m	O		76	72	Teenas	f	O					
17	16	51	Bechar	m	C		77	73	Kouba	f	O					
18	17	54	Mostaganem	m	D		78	75	Hussen Dey	f	D					
19	18	55	Tiaret	m	D		79	77	Khenchela	f	C					
20	19	56	Bouira	m	C		80	82	HassiBahbat	f	C					
21	20	59	Tebessa	m	A		81	84	Baraki	f	A					
22	21	61	ElHarrach	m	O		82	91	Boudouaou	f	D					
23	22	62	Mila	m	O		83	95	Tadjenanet	f	O					
24	23	6S	Fouka	m	A		84	4	Molepolole	f	C					
25																
26			Ninad Karlekar 22306A1012													

OUTPUT:



Male	Female
A	C
O	D
O	A
A	C
C	D
D	D
D	C
C	A
O	C
A	D
C	B
D	C
B	O
O	O
A	A
O	O
B	A
D	C
C	D
A	B
O	A
B	O
A	O

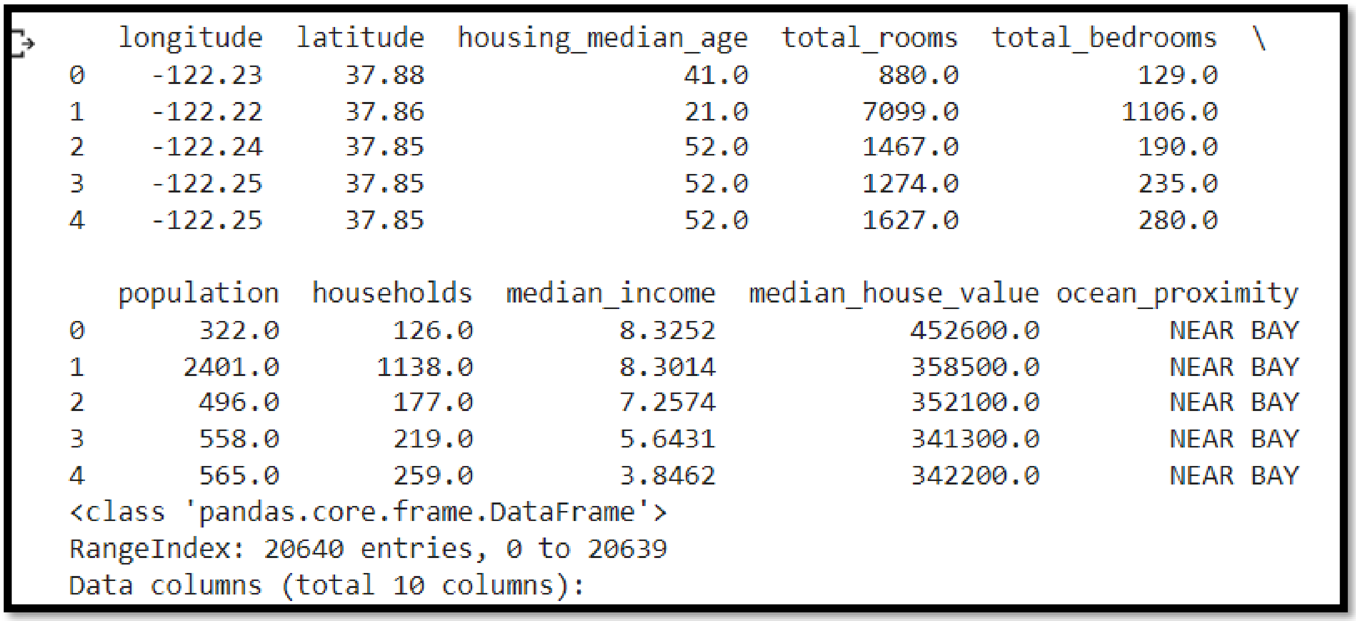
B. Perform the Stratified sampling for the given data and analyse it.

we are to carry out a hypothetical housing quality survey across Lagos state, Nigeria. And we looking at a total of 5000 houses (hypothetically). We don't just go to one local government and select 5000 houses, rather we ensure that the 5000 houses are a representative of the whole 20 local government areas Lagos state is comprised of. This is called stratified sampling. The population is divided into homogenous strata and the right number of instances is sampled from each stratum to guarantee that the test-set (which in this case is the 5000 houses) is a representative of the overall population. If we used random sampling, there would be a significant chance of having bias in the survey results.

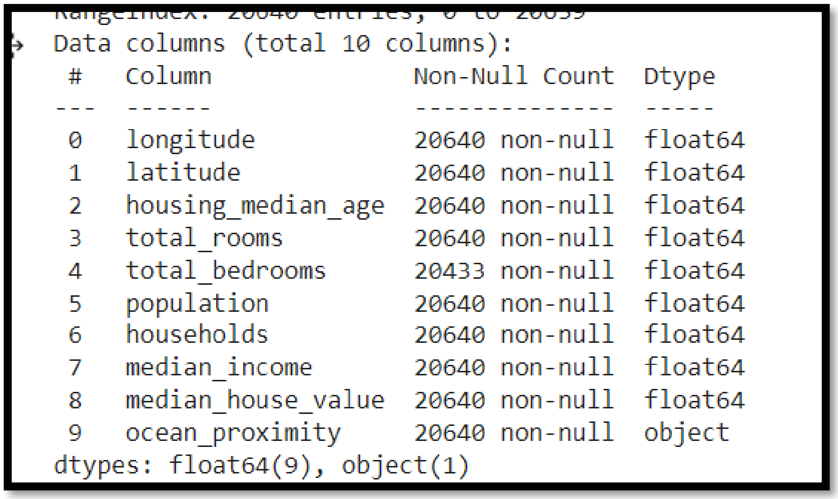
Program Code:

```
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
plt.rcParams['axes.labelsize'] = 14
plt.rcParams['xtick.labelsize'] = 12
plt.rcParams['ytick.labelsize'] = 12
import seaborn as sns
color = sns.color_palette()
sns.set_style('darkgrid')
import sklearn
from sklearn.model_selection import train_test_split
housing =pd.read_csv('housing.csv')
print(housing.head())
print(housing.info())
#creating a heatmap of the attributes in the dataset

correlation_matrix = housing.corr()
plt.subplots(figsize=(8,6))
sns.heatmap(correlation_matrix, center=0, annot=True, linewidths=.3)
corr =housing.corr()
print(corr['median_house_value'].sort_values(ascending=False))
sns.distplot(housing.median_income)
plt.show()
```

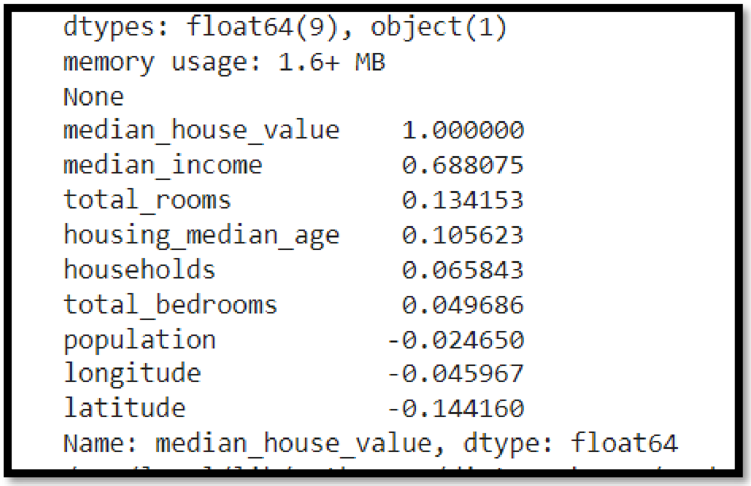
output:



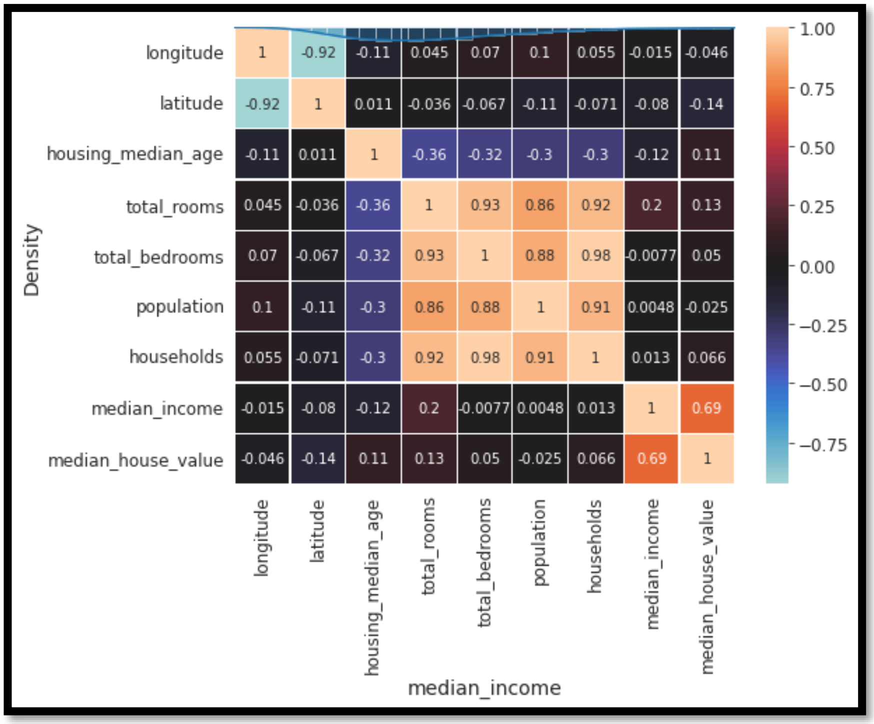
```
   longitude  latitude  housing_median_age  total_rooms  total_bedrooms  \
0    -122.23     37.88                41.0        880.0           129.0   
1    -122.22     37.86                21.0       7099.0          1106.0   
2    -122.24     37.85                52.0       1467.0           190.0   
3    -122.25     37.85                52.0       1274.0           235.0   
4    -122.25     37.85                52.0       1627.0           280.0   

   population  households  median_income  median_house_value ocean_proximity  
0       322.0       126.0         8.3252            452600.0        NEAR BAY  
1      2401.0      1138.0         8.3014            358500.0        NEAR BAY  
2       496.0       177.0         7.2574            352100.0        NEAR BAY  
3       558.0       219.0         5.6431            341300.0        NEAR BAY  
4       565.0       259.0         3.8462            342200.0        NEAR BAY  
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
```



```
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   longitude           20640 non-null  float64
 1   latitude            20640 non-null  float64
 2   housing_median_age  20640 non-null  float64
 3   total_rooms         20640 non-null  float64
 4   total_bedrooms      20433 non-null  float64
 5   population          20640 non-null  float64
 6   households          20640 non-null  float64
 7   median_income       20640 non-null  float64
 8   median_house_value  20640 non-null  float64
 9   ocean_proximity     20640 non-null  object 
dtypes: float64(9), object(1)
```



```
dtypes: float64(9), object(1)
memory usage: 1.6+ MB
None
median_house_value    1.000000
median_income         0.688075
total_rooms           0.134153
housing_median_age    0.105623
households            0.065843
total_bedrooms        0.049686
population           -0.024650
longitude            -0.045967
latitude             -0.144160
Name: median_house_value, dtype: float64
```



	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
longitude	1	-0.92	-0.11	0.045	0.07	0.1	0.055	-0.015	-0.046
latitude	-0.92	1	0.011	-0.036	-0.067	-0.11	-0.071	-0.08	-0.14
housing_median_age	-0.11	0.011	1	-0.36	-0.32	-0.3	-0.3	-0.12	0.11
total_rooms	0.045	-0.036	-0.36	1	0.93	0.86	0.92	0.2	0.13
total_bedrooms	0.07	-0.067	-0.32	0.93	1	0.88	0.98	-0.0077	0.05
population	0.1	-0.11	-0.3	0.86	0.88	1	0.91	0.0048	-0.025
households	0.055	-0.071	-0.3	0.92	0.98	0.91	1	0.013	0.066
median_income	-0.015	-0.08	-0.12	0.2	-0.0077	0.0048	0.013	1	0.69
median_house_value	-0.046	-0.14	0.11	0.13	0.05	-0.025	0.066	0.69	1

VSIT

Research in Computing
Practical # 8

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	Compute different types of correlation.	**Batch**	1

Write a program for computing different correlation

A. Positive Correlation:

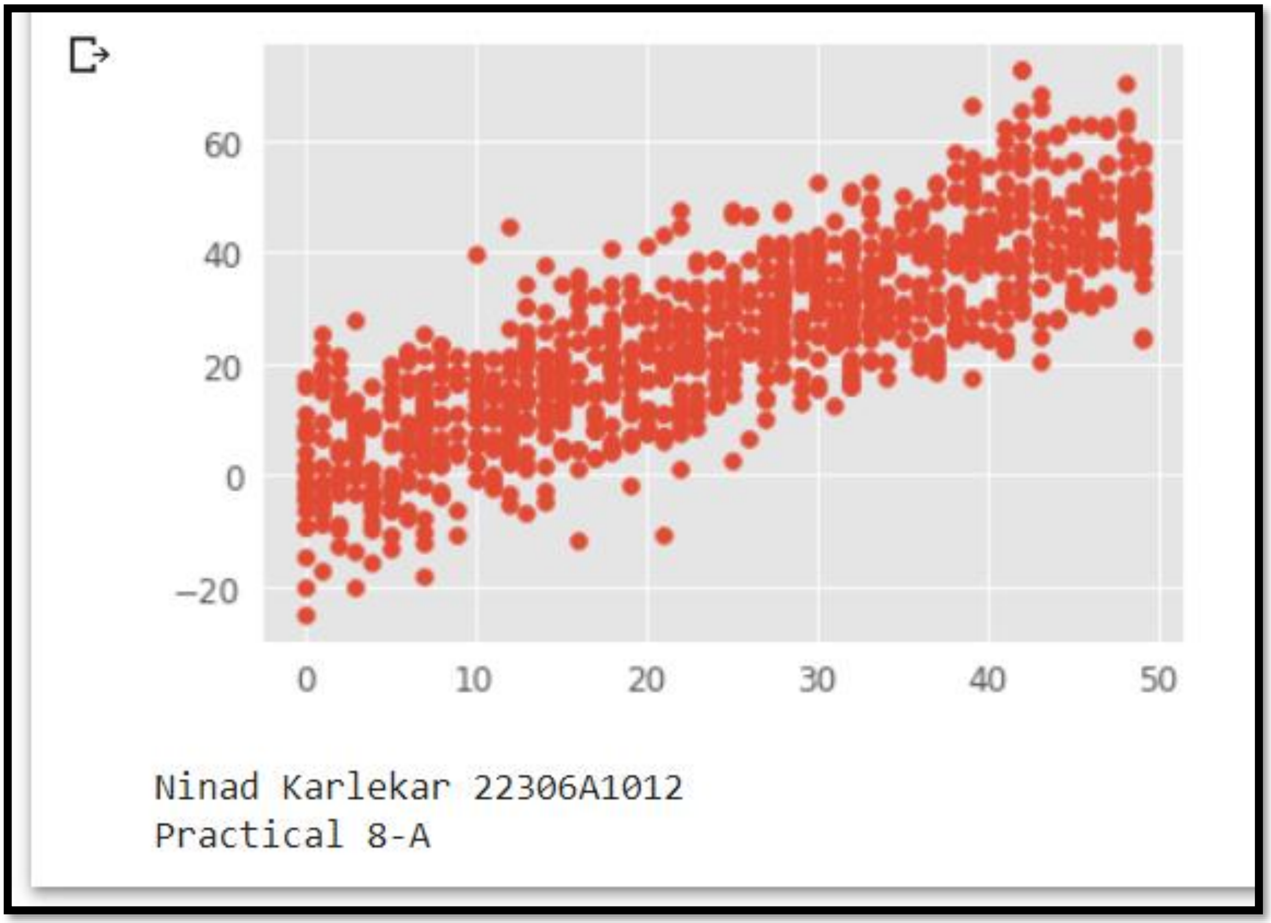
Positive Correlation:

Let’s take a look at a positive correlation. Numpy implements a corrcoef() function that returns a matrix of correlations of x with x, x with y, y with x and y with y. We’re interested in the values of correlation of x with y (so position (1, 0) or (0, 1)).

Code:

```
import numpy as np

import matplotlib.pyplot as plt

np.random.seed(1)

# 1000 random integers between 0 and 50

x = np.random.randint(0, 50, 1000)

# Positive Correlation with some noise

y = x + np.random.normal(0, 10, 1000)

np.corrcoef(x, y)

matplotlib.style.use('ggplot')

plt.scatter(x, y)

plt.show()

print("\nNinad Karlekar 22306A1012")
```

Output:



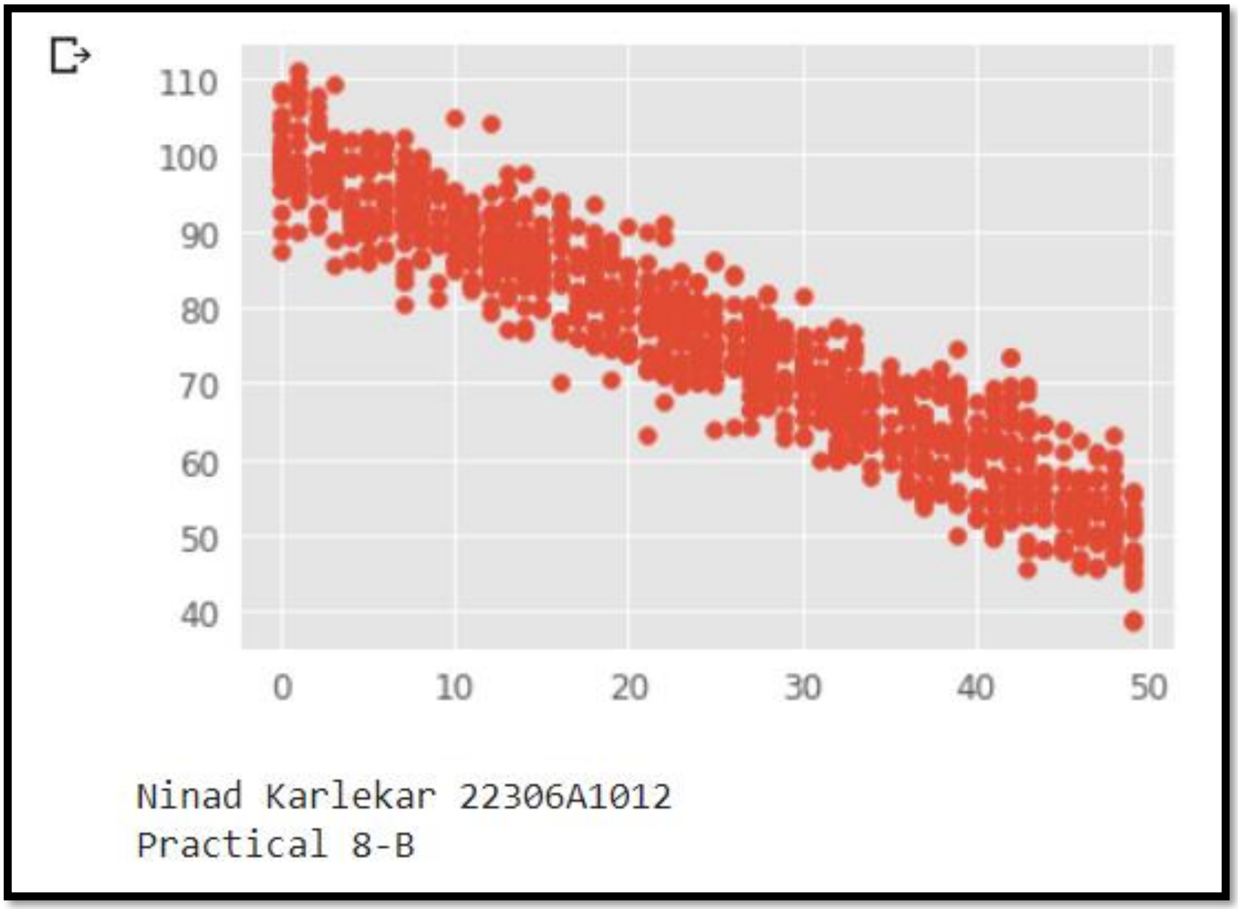
```
Ninad Karlekar 22306A1012
Practical 8-A
```

B. Negative Correlation:

CODE:

```
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(1)
# 1000 random integers between 0 and 50
x = np.random.randint(0, 50, 1000)
# Negative Correlation with some noise
y = 100 - x + np.random.normal(0, 5, 1000)
np.corrcoef(x, y)
plt.scatter(x, y)
plt.show()
print("\nNinad Karlekar 22306A1012")
print("Practical 8-B")
```

OUTPUT:



```
Ninad Karlekar 22306A1012
Practical 8-B
```

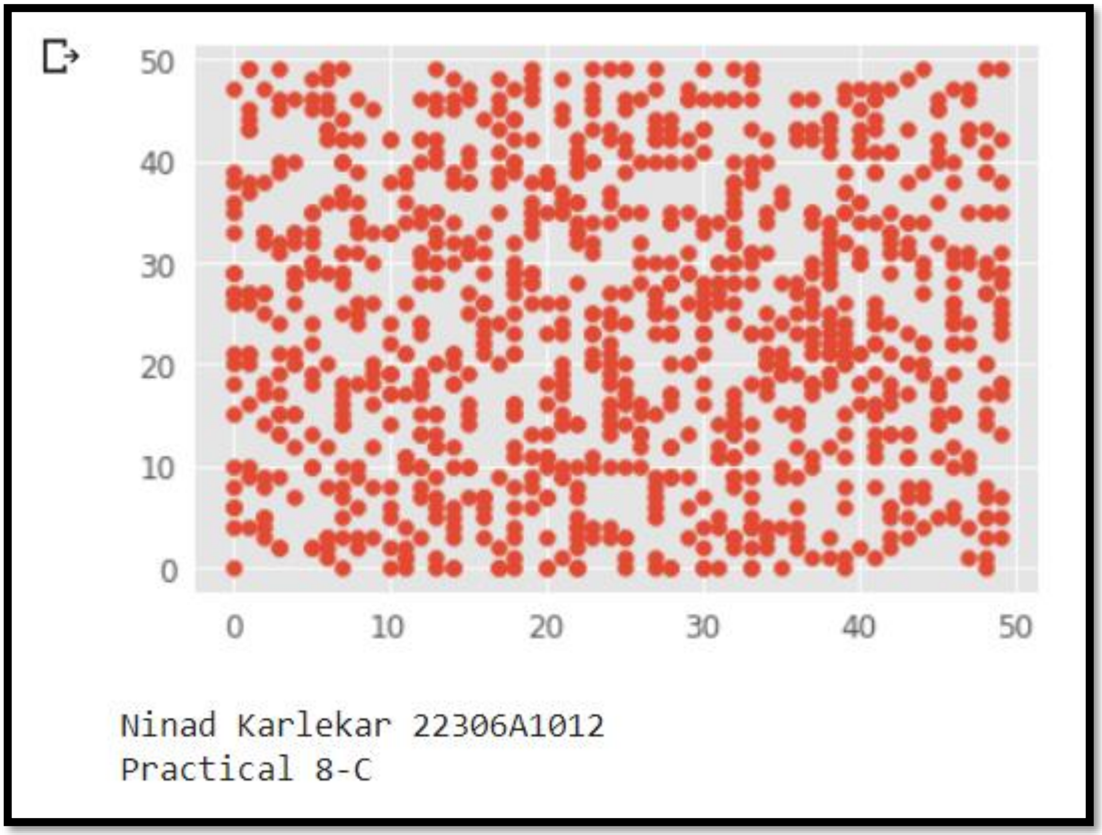
C. No/Weak Correlation:

CODE:

```
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(1)
x = np.random.randint(0, 50, 1000)
y = np.random.randint(0, 50, 1000)
np.corrcoef(x, y)
plt.scatter(x, y)
plt.show()
```

```
print("\nNinad Karlekar 22306A1012")
print("Practical 8-C")
```

OUTPUT:



```
Ninad Karlekar 22306A1012
Practical 8-C
```

Research in Computing
Practical # 9

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Research in Computing Practical	**Class**	M.Sc. IT – Sem I
Topic	Linear regression for prediction. | Polynomial regression for prediction.	**Batch**	1

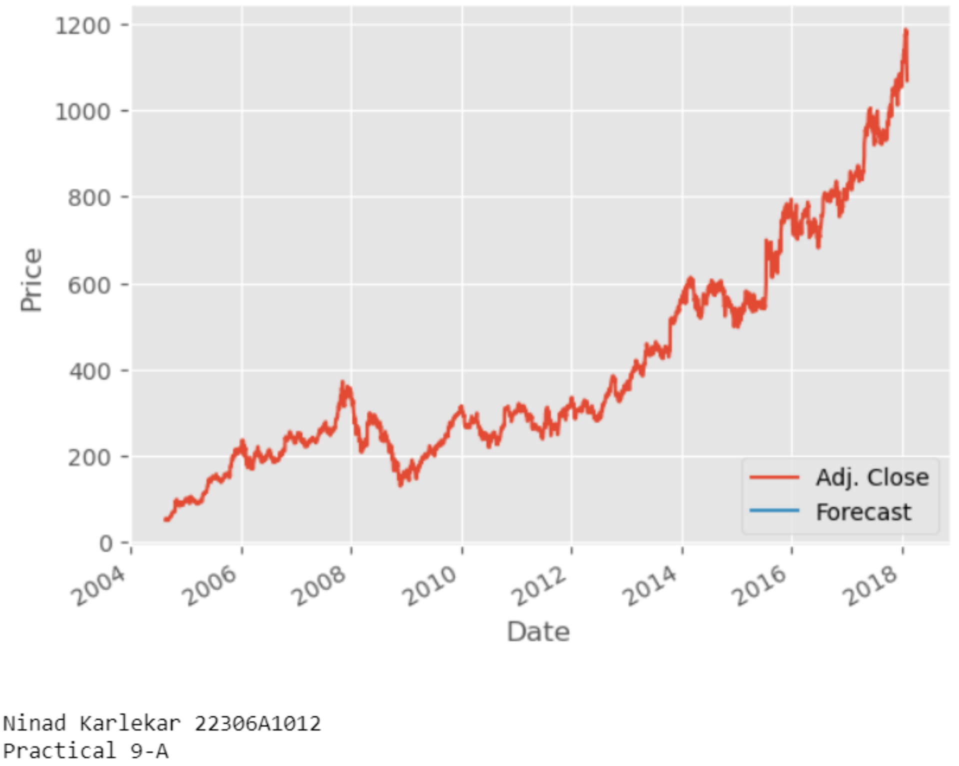
A. Write a program to Perform linear regression for prediction.

CODE:

```
#PRAC 9A #Jupyter
import quandl, math
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn import svm
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
from matplotlib import style
import datetime
style.use('ggplot')
df = quandl.get("WIKI/GOOGL")
df = df[['Adj. Open', 'Adj. High', 'Adj. Low', 'Adj. Close', 'Adj. Volume']]
df['HL_PCT'] = (df['Adj. High'] - df['Adj. Low']) / df['Adj. Close'] * 100.0
df['PCT_change'] = (df['Adj. Close'] - df['Adj. Open']) / df['Adj. Open'] * 100.0
df = df[['Adj. Close', 'HL_PCT', 'PCT_change', 'Adj. Volume']]
forecast_col = 'Adj. Close'
df.fillna(value=-99999, inplace=True)
forecast_out = int(math.ceil(0.01 * len(df)))
df['label'] = df[forecast_col].shift(-forecast_out)
X = np.array(df.drop(['label'], 1))
X = preprocessing.scale(X)
X_lately = X[-forecast_out:]
X = X[:-forecast_out]
df.dropna(inplace=True)
y = np.array(df['label'])
X_train, X_test, y_train, y_test = sklearn.model_selection.train_test_split(X, y, test_size=0.2)
clf = LinearRegression(n_jobs=-1)
clf.fit(X_train, y_train)
confidence = clf.score(X_test, y_test)
forecast_set = clf.predict(X_lately)
df['Forecast'] = np.nan
last_date = df.iloc[-1].name
last_unix = last_date.timestamp()
one_day = 86400
```

```
next_unix = last_unix + one_day
for i in forecast_set:
  next_date = datetime.datetime.fromtimestamp(next_unix)
  next_unix += 86400
df.loc[next_date] = [np.nan for _ in range(len(df.columns)-1)]+[i]
df['Adj. Close'].plot()
df['Forecast'].plot()
plt.legend(loc=4)
plt.xlabel('Date')
plt.ylabel('Price')
plt.show()
print("\nNinad Karlekar 22306A1012")
print("Practical 9-A")
```

OUTPUT:



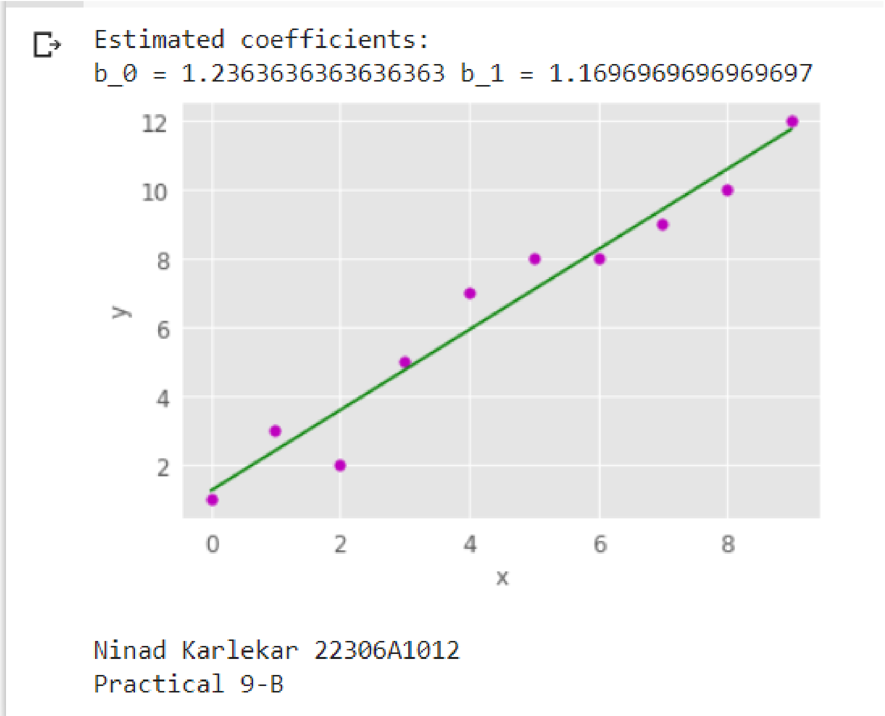
```
Ninad Karlekar 22306A1012
Practical 9-A
```

B. Perform polynomial regression for prediction.

CODE:

```
import numpy as np
import matplotlib.pyplot as plt
def estimate_coef(x, y):
  # number of observations/points
  n = np.size(x)
  # mean of x and y vector
  m_x, m_y = np.mean(x), np.mean(y)
  # calculating cross-deviation and deviation about x
  SS_xy = np.sum(y*x) - n*m_y*m_x
  SS_xx = np.sum(x*x) - n*m_x*m_x
  # calculating regression coefficients
  b_1 = SS_xy / SS_xx
  b_0 = m_y - b_1*m_x
  return(b_0, b_1)
def plot_regression_line(x, y, b):
  # plotting the actual points as scatter plot
  plt.scatter(x, y, color = "m",marker = "o", s = 30)
  # predicted response vector
  y_pred = b[0] + b[1]*x
  # plotting the regression line
  plt.plot(x, y_pred, color = "g")
  # putting labels
  plt.xlabel('x')
  plt.ylabel('y')
  # function to show plot
  plt.show()
def main():
  # observations
  x = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
  y = np.array([1, 3, 2, 5, 7, 8, 8, 9, 10, 12])
  # estimating coefficients
  b = estimate_coef(x, y)
  print("Estimated coefficients:\nb_0 = {} b_1 = {}".format(b[0], b[1]))
  # plotting regression line
  plot_regression_line(x, y, b)
if __name__ == "__main__":
   main()
print("\nNinad Karlekar 22306A1012")
print("Practical 9-B")
```

OUTPUT:

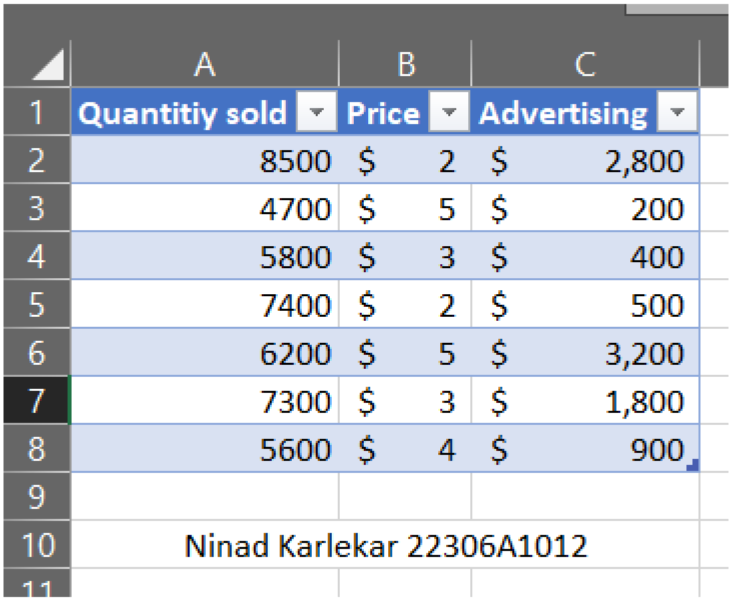


```
Estimated coefficients:
b_0 = 1.2363636363636363 b_1 = 1.1696969696969697
```

```
Ninad Karlekar 22306A1012
Practical 9-B
```

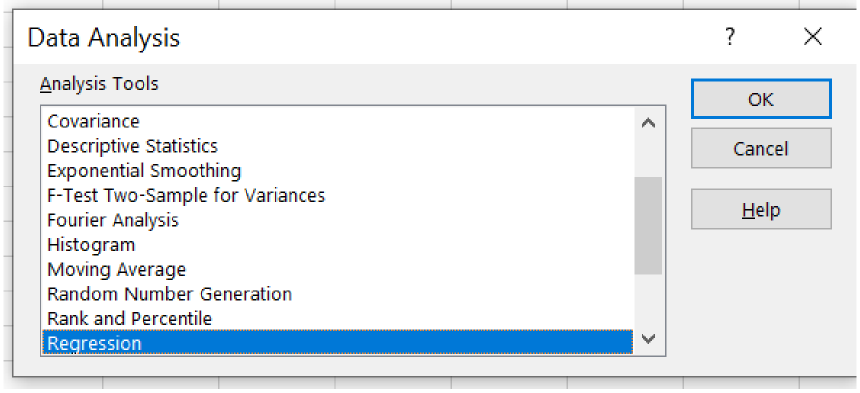
By Excel Steps

1. Insert the data as follows



	A	B	C
1	Quantitiy sold	Price	Advertising
2	8500	$ 2	$ 2,800
3	4700	$ 5	$ 200
4	5800	$ 3	$ 400
5	7400	$ 2	$ 500
6	6200	$ 5	$ 3,200
7	7300	$ 3	$ 1,800
8	5600	$ 4	$ 900
9			
10	Ninad Karlekar 22306A1012		

2. Go to Data -> Data Analysis -> Regression

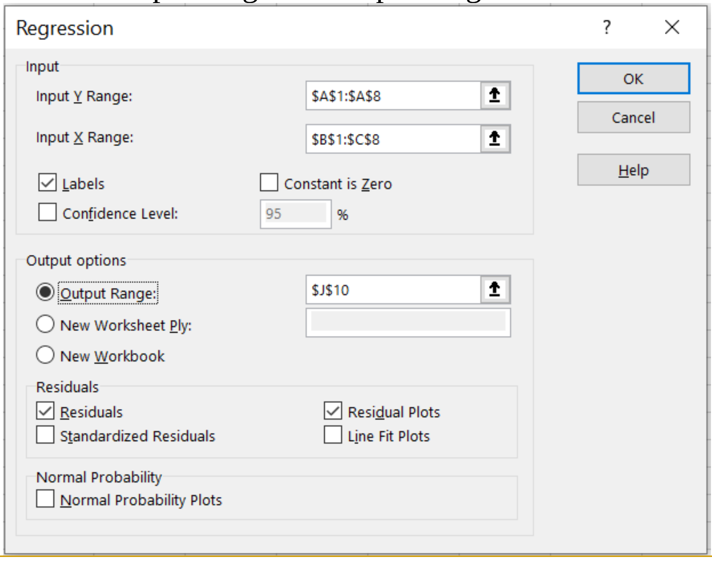


Data Analysis

Analysis Tools

- Covariance
- Descriptive Statistics
- Exponential Smoothing
- F-Test Two-Sample for Variances
- Fourier Analysis
- Histogram
- Moving Average
- Random Number Generation
- Rank and Percentile
- Regression

3. Enter the input range and output range



Regression

Input

Input Y Range: $A$1:$A$8

Input X Range: $B$1:$C$8

☑ Labels ☐ Constant is Zero

☐ Confidence Level: 95 %

Output options

● Output Range: $J$10

○ New Worksheet Ply:

○ New Workbook

Residuals

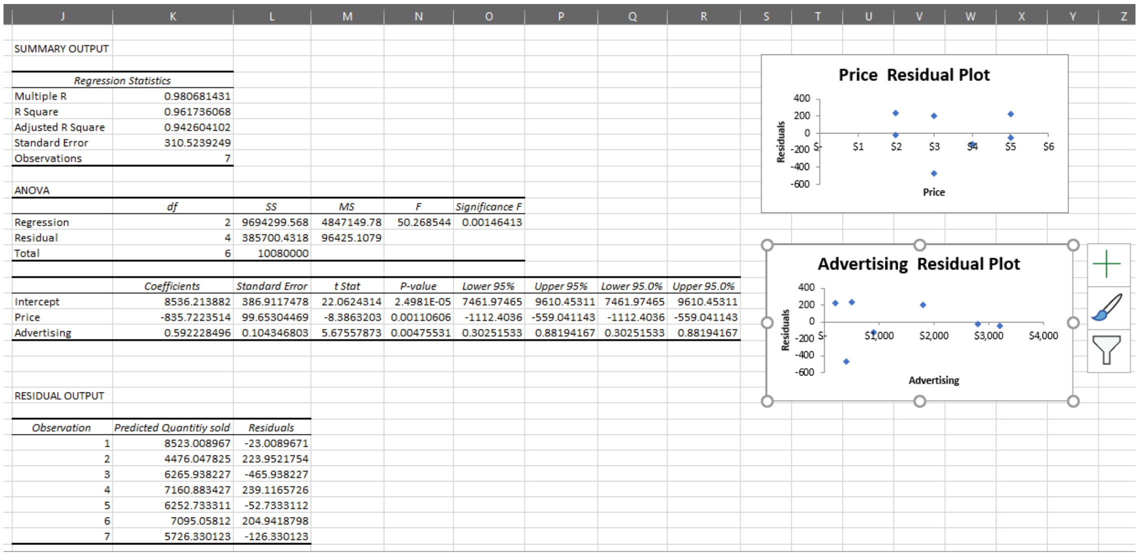
☑ Residuals ☑ Residual Plots

☐ Standardized Residuals ☐ Line Fit Plots

Normal Probability

☐ Normal Probability Plots

4. Click on OK



SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.980681431
R Square	0.961736068
Adjusted R Square	0.942604102
Standard Error	310.5239249
Observations	7

ANOVA

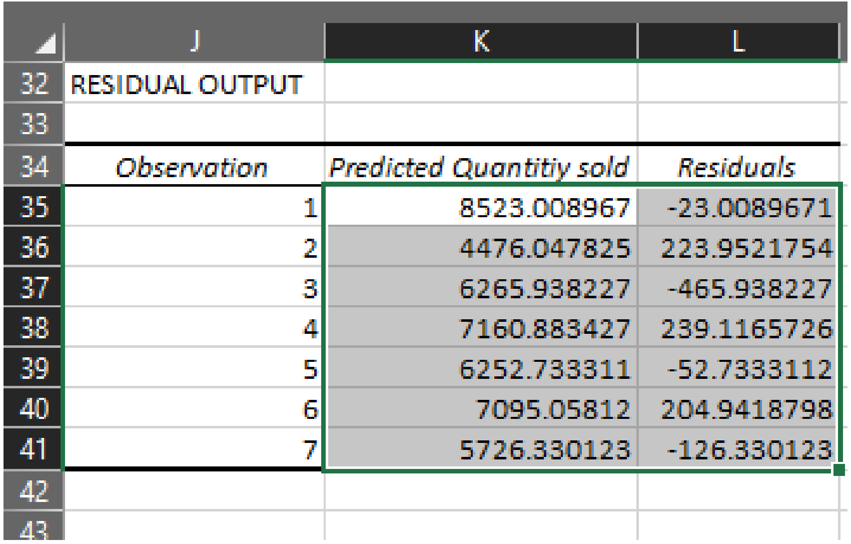
	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	9694299.568	4847149.78	50.268544	0.00146413
Residual	4	385700.4318	96425.1079		
Total	6	10080000			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
Intercept	8536.213882	386.9117478	22.0624314	2.4981E-05	7461.97465	9610.45311	7461.97465	9610.45311
Price	-835.7223514	99.65304469	-8.3863203	0.00110606	-1112.4036	-559.041143	-1112.4036	-559.041143
Advertising	0.592228496	0.104346803	5.67557873	0.00475531	0.30251533	0.88194167	0.30251533	0.88194167

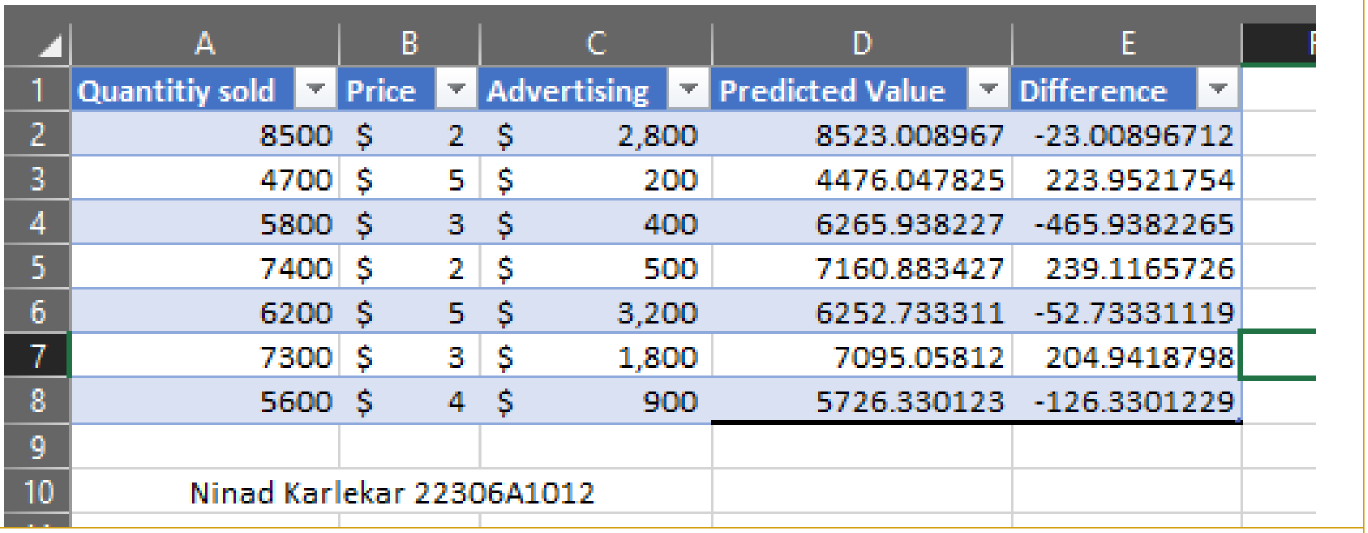
RESIDUAL OUTPUT

Observation	*Predicted Quantitiy sold*	*Residuals*
1	8523.008967	-23.0089671
2	4476.047825	223.9521754
3	6265.938227	-465.938227
4	7160.883427	239.1165726
5	6252.733311	-52.7333112
6	7095.05812	204.9418798
7	5726.330123	-126.330123

5. Select the PREDICTED QUANTITY SOLD and RESIDUALS column and paste on above table

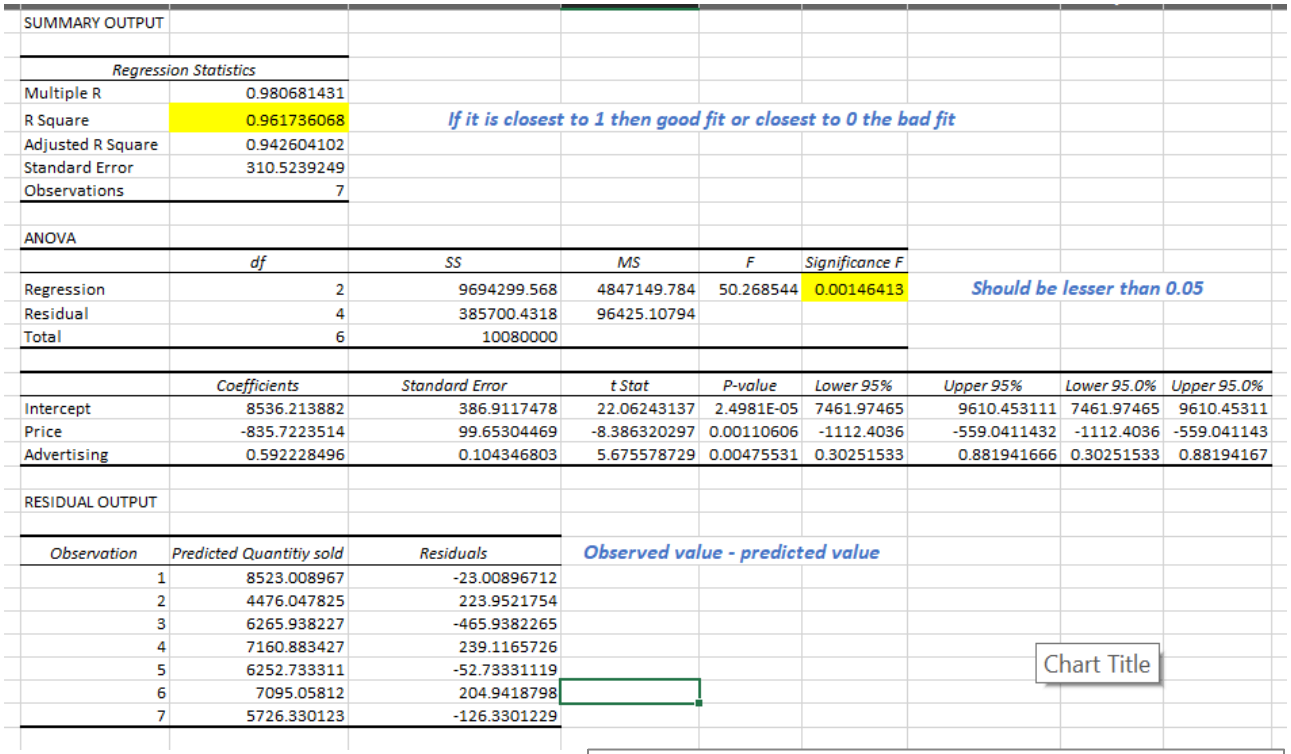


	J	K	L
32	RESIDUAL OUTPUT		
33			
34	*Observation*	*Predicted Quantitiy sold*	*Residuals*
35	1	8523.008967	-23.0089671
36	2	4476.047825	223.9521754
37	3	6265.938227	-465.938227
38	4	7160.883427	239.1165726
39	5	6252.733311	-52.7333112
40	6	7095.05812	204.9418798
41	7	5726.330123	-126.330123
42			
43			



	A	B	C	D	E
1	Quantitiy sold	Price	Advertising	Predicted Value	Difference
2	8500	$ 2	$ 2,800	8523.008967	-23.00896712
3	4700	$ 5	$ 200	4476.047825	223.9521754
4	5800	$ 3	$ 400	6265.938227	-465.9382265
5	7400	$ 2	$ 500	7160.883427	239.1165726
6	6200	$ 5	$ 3,200	6252.733311	-52.73331119
7	7300	$ 3	$ 1,800	7095.05812	204.9418798
8	5600	$ 4	$ 900	5726.330123	-126.3301229
9					
10	Ninad Karlekar 22306A1012				

OUTPUT:



SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.980681431
R Square	0.961736068
Adjusted R Square	0.942604102
Standard Error	310.5239249
Observations	7

If it is closest to 1 then good fit or closest to 0 the bad fit

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	9694299.568	4847149.784	50.268544	0.00146413
Residual	4	385700.4318	96425.10794		
Total	6	10080000			

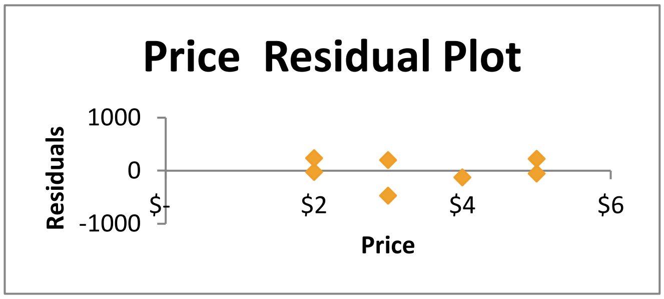
Should be lesser than 0.05

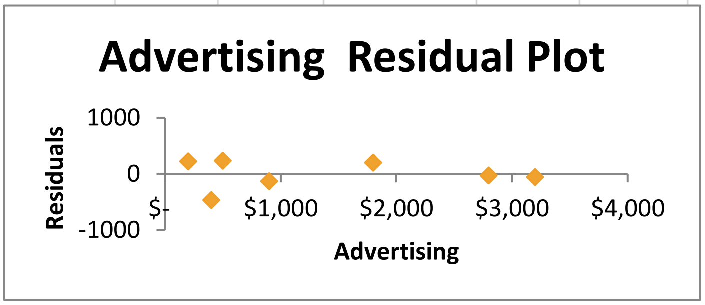
	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
Intercept	8536.213882	386.9117478	22.06243137	2.4981E-05	7461.97465	9610.453111	7461.97465	9610.45311
Price	-835.7223514	99.65304469	-8.386320297	0.00110606	-1112.4036	-559.0411432	-1112.4036	-559.041143
Advertising	0.592228496	0.104346803	5.675578729	0.00475531	0.30251533	0.881941666	0.30251533	0.88194167

RESIDUAL OUTPUT

Observation	*Predicted Quantitiy sold*	*Residuals*
1	8523.008967	-23.00896712
2	4476.047825	223.9521754
3	6265.938227	-465.9382265
4	7160.883427	239.1165726
5	6252.733311	-52.73331119
6	7095.05812	204.9418798
7	5726.330123	-126.3301229

Observed value - predicted value





Result:
R square equals 0.962, which is a very good fit. 6% of the variation in Qunatity Sold is explained by the independent variables Price and Advertising. The closer to 1, the better the regression line (read on) fits the data.
Significance F is 0.001464128 which is less than 0.05 (good fit).