A MINOR PROJECT

On

WARRANTY CLAIM FRAUD DETECTION USING DATA ANALYSIS

Dissertation submitted in the partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

By

DEPARTMENT OF INTERNSHIPS

TALARI TEJASWI

CSWDA133

Under the esteemed Guidance of

Er. Y V D CHANDRA SEKHAR

Founder & Chief Executive Officer

CS CODENZ



LEARN HERE, LEAD ANYWHERE

CS CODENZ

GUDIVADA – 521 323, ANDHRA PRADESH., INDIA

2022-2023

A MINOR PROJECT

On

WARRANTY CLAIM FRAUD DETECTION USING DATA ANALYSIS

Dissertation submitted in the partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

By

DEPARTMENT OF INTERNSHIPS

TALARI TEJASWI

CSWDA133

Under the esteemed Guidance of

Er. Y V D CHANDRA SEKHAR

Founder & Chief Executive Officer

CS CODENZ



LEARN HERE, LEAD ANYWHERE
CS CODENZ

GUDIVADA – 521 323, ANDHRA PRADESH., INDIA

2022-2023

CS CODENZ



CERTIFICATE

This is to certify that dissertation entitled "Warranty Claim Fraud Detection Using Data Analysis" submitted by TALARI TEJASWI (CSWDA133) in the partial fulfillment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY from CS CODENZ is a record of Bonafede work carried out by them under my guidance and supervision during the year 2022-2023. The result embodied in this dissertation have not been submitted by any other university or Institution for the award of any degree.

Signature of the Supervisor

Er. Y V D CHANDRA SEKHAR

Founder & CEO, CS CODENZ

DECLARATION

I TALARI TEJASWI (CSWDA133) declared that the dissertation report entitled "Warranty Claim Fraud Detection Using Data Analysis" is no more than 1,00,000 words in length including quotes and exclusive of tables, figures, bibliography, and references. This dissertation contains no material that has been submitted previously, in whole or in part, for the award of any other academic degree ordiploma. Except where otherwise indicated this dissertation in our own work.

Roll No	Name	Signature
CSWDA133	TALARI TEJASWI	

Date:

Place:

COs, POs and PSOs Mapping

Subject Name : Major Project

Subject Code : PY42223

AcademicYear : 2022 - 2023

Subject Code		Course Outcomes
	CO1	Formulate solutions to computing problems using latest technologies and tools
	CO2	Work effectively in teams to design and implement solutions to computational problems and socially relevant issues
PR4204	CO3	Recognize the social and ethical responsibilities of a professional working in the discipline
	CO4	Apply advanced algorithmic and mathematical concepts to the design and analysis of software
	CO5	Devise a communication strategy (language, content and medium) to deliver messages according to the situation and need of the audience.
	CO6	Deliver effective presentations, extemporaneous or impromptu oral presentations. Setting up technical reports using technical tools.

CO-PO-PSOs Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	2	1	2	2	-	1	-	-	-	1	1	3	-	-
CO 2	2	3	ı	2	2	ı	ı	ı	ı	-	ı	1	3	ı	-
CO 3	3	3	ı	2	2	1	ı	1	1	-	1	1	3	ı	-
CO 4	3	3	1	2	2	1	1	1	1	-	ı	ı	3	1	-
CO 5	2	3	1	2	2	1	ı	1	1	-	1	1	3	ı	-
CO 6	2	3	2	2	3	ı	1	1	2	2	2	2	3	ı	-
Avg	2.50	2.83	2.00	2.00	2.17	-	-		2.00	2.00	2.00	1.50	3.00	-	-

Note: 1 – Good , 2 – Average, 3 - Excellent

Signature of Student with Date

Signature of Guide with Date

ACKNOWLEDGEMENT

This report dissertation could not have been written without the support of our guide Er. Y V D Chandra Sekhar, Founder & CEO, CS CODENZ who not only served as our superior but also encouraged and challenged us throughout our academic program our foremost thanks goes to his. Without his this dissertation would not have been possible. We appreciate him vast knowledge in many areas, and his insights, suggestions and guidance that helped to shapeour research skills

It is needed with a great sense of pleasure and immense sense of gratitude that we acknowledge the help of these individuals. We owe many thanks to many people who helped and supported us during the writing of this report

We are thankful to our project coordinator **Er. Y V D Chandra Sekhar,** Founder & CEO, CS CODENZ, for his continuous support

We express our sincere thanks to our respected for bet valuable suggestion and constant motivation that greatly helped us in successful completion of project We also take the privilegeto express our heartfelt gratitude to Er. Y V D Chandra Sekhar, Founder & CEO,CS CODENZ

We are thankful to all faculty members for extending their kind cooperation and assistance Finally, we are extremely thankful to our parents and friends for their constant helped moral support

Table Of Contents

Abstract	(i)
Problem Statement	1
ER Diagram	2
Requirements	3
Description	4
Coding	5
Result	
Output	
Summary	51

ABSTRACT

The Primary objective of this "WARRANTY CLAIM FRAUD DETECTION USING DATA ANALYSIS" is to fulfill customer satisfaction and good trust along with challenges of the businesses. It includes data of historical warranty claims and real-time monitoring capabilities.

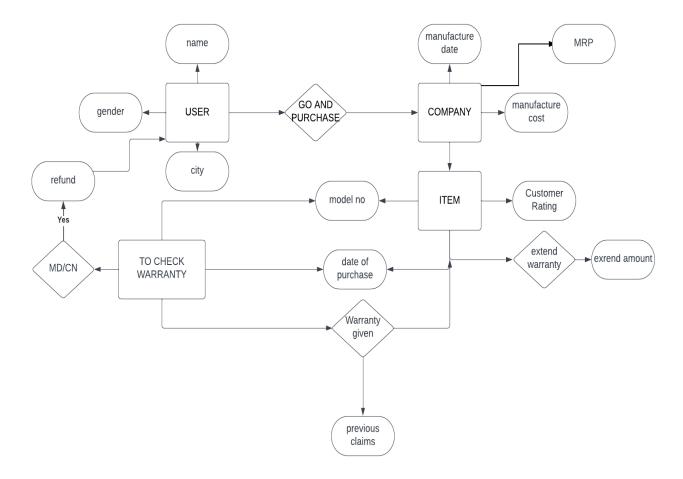
The project also focuses on achieving a high level of accuracy to find the frauds minimizing false positives. It will provide an interface for analysts to easily observe the frauds. It aims to provide a powerful tool to detect and combat warranty claim fraud. This is also help in reducing financial losses to the business providers.

PROBLEM STATEMENT

In today's competitive market, warranty claims are essential for maintaining customer trust and satisfaction. But warranty fraud remains a significant challenge for manufacturers and service providers, resulting in financial losses and eroding customer confidence.

The problem at hand is to develop an efficient and accurate warranty claim fraud detection system using data techniques.

E-R DIAGRAM



REQUIREMENTS

HARDWARE REQUIREMENTS

- Personal Computer / Laptop with minimum RAM (4 GB), ROM (128 GB) and Processor(i3)
- ➤ Good latency internet access

SOFTWARE REQUIREMENTS

- ➤ Basic Search Engine (Google)
- ➤ Google Colaboratory
- ➤ MICROSOFT WORD

FUNCTIONAL REQUIREMENTS

- > Calculation
- ➤ Help in manipulating data and easy process.
- > Graphical representation of Datasets

DESCRIPTION

Data Analytics is a process of scrutinizing the data to obtain accurate results. In data analytics the main purpose is extracting the original data from data. In this data analytics we need to perform the major that is data munging.

DATA MUNGING

It is a process of transferring unstructured data into structured format. The goal is to make the data more usable and valuable for analytics or other purposes.

STEPS OF DATA ANALYSIS

- 1. Defining the Question
- 2. Collecting the data
- 3. Cleaning the data
- 4. Analyzing the data
- 5. Sharing your results
- 6. Embracing your failures
- 7. Summary

CODE

1) DEFINING THE QUESTION

The first step in any data analysis process is to define your objective. In data analytics **jargon**, this is sometimes called the 'problem statement'. The problem at hand is to develop an efficient and accurate warranty claim fraud detection system.

2) COLLECTING THE DATA

Pandas - Helps to create a dataset and it is also a library in python.

Pandas Package – It's a group of Panal Data's which are used to analyze the labelled data and relational data.

Series – A series is a method of pandas and labelled data. Series are nothing but columns in Excel sheet

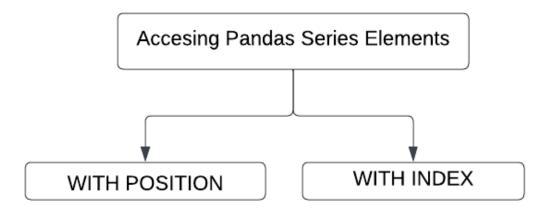
CREATING SERIES-

```
import pandas as pd
a=[10,20,30,'a',50]
b=pd.Series(a)
print(b)

0    10
1    20
2    30
3    a
4    50
dtype: object
```

```
import pandas as pd
a=[10,20,30,40,50]
b=pd.Series(a)
print(b)
```

```
0 10
1 20
2 30
3 40
4 50
dtype: int64
```



ACCESSING WITH POSITION-

```
#SLICING
import pandas as pd
a=[10,20,30,40,50]
b=pd.Series(a)
print(b[-2:])
```

3 40 4 50 dtype: int64

```
#SLICING
import pandas as pd
a=[10,20,30,40,50]
b=pd.Series(a)
print(b[-4:-2])
```

1 20 2 30 dtype: int64

```
import pandas as pd
a=[10,20,30,40,50]
b=pd.Series(a)
print(b[2:])
```

2 30 3 40 4 50 dtype: int64

ACCESSING WITH INDEX-

```
import pandas as pd
 a=[10,20,'CS',40,50]
b=pd.Series(a,index=['!','@','#','$','%'])
 print(b)
print("----")
print(b['@'])
    10
<u>a</u>
    20
#
    CS
$
    40
    50
dtype: object
20
```

DATAFRAME

A data frame is a 2D data structure in which we store data in the form of tables. [rows x columns] We can create a table via Data Frame i.e., known as DATASET.

CREATING A DATASET-

```
#creating empty data set
import pandas as pd
a=pd.DataFrame()
print(a)
```

```
Empty DataFrame
Columns: []
Index: []
```

Creating data set using list-

```
#creating dataframe by using list
import pandas as pd
a=[10,20,30,40,50]
b=pd.DataFrame(a)
print(b)
```

```
0
0 10
1 20
2 30
3 40
4 50
```

Creating data set using Dict-

```
#creating using DICT
import pandas as pd
x=[{'a':10,'b':20,'c':30}]
y=pd.DataFrame(x)
print(y)
```

```
a b c 0 10 20 30
```

```
#creating using DICT
import pandas as pd
x=[{'a':10,'b':20,'c':30}]
y=pd.DataFrame(x)
print(type(y))
```

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

Creating dataset using Series

```
import pandas as pd
a=[10,20,30,40]
b=pd.Series(a)
c=pd.DataFrame(b)
print(c)
```

- 0 10
- 1 20
- 2 30
- 3 40

CREATION OF LARGE DATASET-

```
import pandas as pd
a={'S.No':pd.Series([1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]),
    'Name':pd.Series(['vagesh','venkat','reddy','yashwanth','ram','kumar','abhi',
'bhavana', 'meghana', 'swapna', 'rani', 'keerthana', 'akash', 'siva', 'shankar', 'pradeep',
'sameena','samba','sai','vamsi']),
    'm','f','m','m','m']),
    'City':pd.Series(['hyderabad','pune','nuzvid','vijayawada','mumbai','kolkata',
      'bangalore', 'chennai', 'delhi', 'vizag', 'tirupati', 'nellore', 'kadapa', 'kurnool',
      'kakinada','srikakulam','nizamabad','noida','khammam','nalgonda',]),
    'DateOfPurchase':pd.Series(["01 jun 2020", "05 july 2020", "28 aug 2020",
      "25 sep 2020", "27 oct 2020", "31 nov 2020", "15 dec 2020", "17 dec 2020",
      "04 jan 2021","17 jan 2021","21 jan 2021","14 feb 2021","16 feb 2021",
      "25 feb 2021", "27 feb 2021", "01 mar 2021", "05 mar 2021", "10 mar 2021",
       "14 mar 2021","18 mar 2021"]),
   'Customer Rating(0-5)':([4,4,4,3,3,3,3,5,5,5,4,4,5,4,3,2,1,5,2,1]),
   'ManufactureDate':(["11 jun 2019","15 july 2019","18 aug 2019","15 sep 2019",
         "17 oct 2019", "21 nov 2019", "05 dec 2019", "07 dec 2019", "14 jan 2020",
         "07 jan 2020","11 jan 2019","04 feb 2019","06 feb 2019","15 feb 2019",
            "17 feb 2019","11 mar 2019","15 mar 2019","11 mar 2019",
             "04 mar 2019", "28 mar2019"]),
   'Condition':(['used','used','new','new','used','new','new','new','used',
               'used','new','used','new','new','new','used','used','new']),
   'ModelNo': ([101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,
              118,119,120]),
   'Mrp':([2380, 3472, 3127, 2704, 2575, 3210, 2999, 3866, 2750, 3665, 2455, 3766,
          3944, 2085, 3628, 2992, 3599, 2122, 2821, 2200]),
   'SoldPrice':([2975, 2536, 2199, 3515, 3089, 2288, 2399, 3966, 3575, 3765, 2946,
                1883, 3844, 2490, 2711, 3590, 2159, 2546, 3385,2100]),
   'Manufacturing Cost': ([230,340,310,270,250,320,290,380,270,360,240,370,390,200,
                         360,290,350,210,2800,220]),
   'Extend Amount':([23,34,31,27,25,32,29,38,27,36,24,37,39,20,36,29,35,21,28,22]),
   'Warranty periodLeft(in Months)':([0,2,4,0,6,8,9,4,7,5,8,0,0,6,5,4,2,2,1,1]),
   'MD/CN': ([0,0,0,0,1,0,1,1,1,0,1,1,1,1,1,1,1,1,0,0,0]),
   'Previous claims':([1,0,0,2,1,1,1,0,0,0,0,0,0,0,1,2,2,3,1,1]),
   'Refund Duration(in days)':([3,3,4,3,5,7,2,2,2,2,3,3,3,3,4,4,5,5,7,7])
b=pd.DataFrame(a)
print(b)
```

	S.No	Na	me	Gender	Citv	Date0	fPuro	chase	<pre>Customer_Rating(0-5) \</pre>
0	1	vage		m	hyderabad			2020	4
1	2	venk		m	pune		_	2020	4
2	3	red	ldy	m	nuzvid	-	-	2020	4
3	4	yashwar	-	m	vijayawada			2020	3
4	5	•	am	m	mumbai		-	2020	3
5	6	kun	ıar	m	kolkata	31	nov	2020	3
6	7	ab	hi	f	bangalore	15	dec	2020	3
7	8	bhava	na	f	chennai	17	dec	2020	3
8	9	megha	na	f	delhi	04	jan	2021	5
9	10	swap	na	f	vizag	17	jan	2021	5
10	11	ra	ni	f	tirupati	21	jan	2021	4
11	12	keertha	na	f	nellore	14	feb	2021	4
12	13	aka	sh	m	kadapa	16	feb	2021	5
13	14	si	lva	m	kurnool	25	feb	2021	4
14	15	shank	car	m	kakinada	27	feb	2021	3
15	16	prade	ер	m	srikakulam	01	mar	2021	2
16	17	samee	ena	f	nizamabad	05	mar	2021	1
17	18	san	ıba	m	noida	10	mar	2021	5
18	19	9	ai	m	khammam	14	mar	2021	2
19	20	van	nsi	m	nalgonda	18	mar	2021	1
	Manufa	ctureDat	-0 (Condition	ModelNo	Mnn	Sold	dDni ca	Manufacturing Cost \
a				Condition		Mrp	Solo	dPrice	Manufacturing_Cost \
0	11	jun 201	L9	used	101	2380	Solo	2975	230
1	11 15	jun 201 july 201	L9 L9	used used	101 102	2380 3472	Solo	2975 2536	230 340
1 2	11 15 18	jun 201 july 201 aug 201	L9 L9 L9	used used used	101 102 103	2380 3472 3127	Solo	2975 2536 2199	230 340 310
1 2 3	11 15 18 15	jun 201 july 201 aug 201 sep 201	L9 L9 L9	used used used new	101 102 103 104	2380 3472 3127 2704	Solo	2975 2536 2199 3515	230 340 310 270
1 2 3 4	11 15 18 15 17	jun 201 july 201 aug 201 sep 201 oct 201	19 19 19 19	used used used new new	101 102 103 104 105	2380 3472 3127 2704 2575	Solo	2975 2536 2199 3515 3089	230 340 310 270 250
1 2 3 4 5	11 15 18 15 17 21	jun 201 july 201 aug 201 sep 201 oct 201 nov 201	19 19 19 19	used used used new new used	101 102 103 104 105 106	2380 3472 3127 2704 2575 3210	Solo	2975 2536 2199 3515 3089 2288	230 340 310 270 250 320
1 2 3 4 5 6	11 15 18 15 17 21 05	jun 201 july 201 aug 201 sep 201 oct 201 nov 201 dec 201	19 19 19 19 19	used used used new new used new	101 102 103 104 105 106 107	2380 3472 3127 2704 2575 3210 2999	Solo	2975 2536 2199 3515 3089 2288 2399	230 340 310 270 250 320 290
1 2 3 4 5	11 15 18 15 17 21 05	jun 201 july 201 aug 201 sep 201 oct 201 nov 201 dec 201 dec 201	19 19 19 19 19	used used used new new used new	101 102 103 104 105 106 107 108	2380 3472 3127 2704 2575 3210 2999 3866	Solo	2975 2536 2199 3515 3089 2288 2399 3966	230 340 310 270 250 320 290 380
1 2 3 4 5 6 7 8	11 15 18 15 17 21 05 07	jun 201 july 201 aug 201 sep 201 oct 201 nov 201 dec 201 dec 201	19 19 19 19 19 19	used used used new new used new new	101 102 103 104 105 106 107 108 109	2380 3472 3127 2704 2575 3210 2999 3866 2750	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575	230 340 310 270 250 320 290 380 270
1 2 3 4 5 6 7 8	11 15 18 15 17 21 05 07 14	jun 201 july 201 aug 201 sep 201 oct 201 nov 201 dec 201 dec 201 jan 202	19 19 19 19 19 19	used used new new used new new used	101 102 103 104 105 106 107 108 109 110	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765	230 340 310 270 250 320 290 380 270 360
1 2 3 4 5 6 7 8 9 10	11 15 18 15 17 21 05 07 14 07	jun 201 july 201 aug 201 sep 201 oct 201 dec 201 dec 201 jan 202 jan 202	19 19 19 19 19 19 19	used used used new used new new used used	101 102 103 104 105 106 107 108 109 110	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946	230 340 310 270 250 320 290 380 270 360 240
1 2 3 4 5 6 7 8 9 10	11 15 18 15 17 21 05 07 14 07 11	jun 201 july 201 aug 201 sep 201 nov 201 dec 201 jan 202 jan 201 feb 201	19 19 19 19 19 19 19 19	used used used new used new new used new used used used	101 102 103 104 105 106 107 108 109 110 111	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765	230 340 310 270 250 320 290 380 270 360 240 370
1 2 3 4 5 6 7 8 9 10	11 15 18 15 17 21 05 07 14 07 11 04	jun 201 july 201 aug 201 sep 201 oct 201 dec 201 dec 201 jan 202 jan 202	19 19 19 19 19 19 19 19	used used used new used new new used used	101 102 103 104 105 106 107 108 109 110 111 112 113	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883	230 340 310 270 250 320 290 380 270 360 240
1 2 3 4 5 6 7 8 9 10 11 12	11 15 18 15 17 21 05 07 14 07 11 04	jun 201 july 201 aug 201 sep 201 nov 201 dec 201 jan 202 jan 202 jan 201 feb 201	19 19 19 19 19 19 19 19 19	used used used new used new new used used used used	101 102 103 104 105 106 107 108 109 110 111 112 113	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844	230 340 310 270 250 320 290 380 270 360 240 370 390
1 2 3 4 5 6 7 8 9 10 11 12 13	11 15 18 15 17 21 05 07 14 07 11 04 06 15	jun 201 july 201 aug 201 sep 201 oct 201 dec 201 jan 202 jan 202 jan 201 feb 201 feb 201	19 19 19 19 19 19 19 19 19 19	used used used new used new new used used used new used	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2490	230 340 310 270 250 320 290 380 270 360 240 370 390 200
1 2 3 4 5 6 7 8 9 10 11 12 13 14	11 15 18 15 17 21 05 07 14 07 11 04 06 15	jun 201 july 201 aug 201 sep 201 nov 201 dec 201 jan 202 jan 202 jan 201 feb 201 feb 201 feb 201	19 19 19 19 19 19 19 19 19 19	used used used new used new used used used new used new used	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2490 2711	230 340 310 270 250 320 290 380 270 360 240 370 390 200 360
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17	jun 201 july 201 aug 201 sep 201 oct 201 dec 201 jan 202 jan 202 jan 201 feb 201 feb 201 feb 201 feb 201	19 19 19 19 19 19 19 19 19 19	used used used new used new used used used new used new used	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2490 2711 3590	230 340 310 270 250 320 290 380 270 360 240 370 390 200 360 290
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17 11	jun 201 july 201 aug 201 sep 201 oct 201 dec 201 jan 202 jan 202 jan 201 feb 201 feb 201 feb 201 feb 201 mar 201 mar 201	19 19 19 19 19 19 19 19 19 19 19	used used used new used new used used new used new used new used	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2490 2711 3590 2159	230 340 310 270 250 320 290 380 270 360 240 370 390 200 360 290
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	11 15 18 15 17 21 05 07 14 07 11 04 15 17 11 15	jun 201 july 201 aug 201 sep 201 nov 201 dec 201 jan 202 jan 202 jan 201 feb 201 feb 201 feb 201 mar 201 mar 201 mar 201	19 19 19 19 19 19 19 19 19 19 19	used used used new used new used used new used new used new used new used	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599 2122	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2490 2711 3590 2159 2546	230 340 310 270 250 320 290 380 270 360 240 370 390 200 360 290 350 210

```
Extend_Amount Warranty_Period Warranty_periodLeft(in Months)
                                                                                MD/CN \
0
                 23
                 34
                                                                            2
1
                                      1
                                                                                    0
2
                 31
                                      1
                                                                           4
                                                                                    0
                                                                           0
3
                                      1
                 27
                                                                                    0
4
                 25
                                      1
                                                                           6
                                                                                    1
5
                                      1
                                                                           8
                 32
                                                                                    0
                                                                           9
6
                 29
                                      1
                                                                                    1
7
                 38
                                      1
                                                                           4
                                                                                    1
                                                                           7
8
                 27
                                      1
                                                                                    1
9
                 36
                                      1
                                                                            5
                                                                                    0
10
                 24
                                      1
                                                                           8
                                                                                    1
11
                 37
                                      1
                                                                           0
                                                                                    1
                                                                           0
12
                                      1
                 39
                                                                                    1
                                                                           6
13
                 20
                                      1
                                                                                    1
                                                                           5
14
                                      1
                                                                                    1
                 36
                 29
                                                                           4
15
                                      1
                                                                                    1
16
                 35
                                      1
                                                                           2
                                                                                    1
                                                                           2
17
                 21
                                      1
                                                                                    0
18
                 28
                                      1
                                                                           1
                                                                                    0
19
                 22
                                      1
                                                                           1
                                                                                    0
          Previous_claims Refund_Duration(in days)
0
                     1
                                                   3
                    0
1
2
                     0
                                                   3
5
3
                     2
4
                     1
                                                   7
5
                     1
                                                   2
6
                     1
7
                                                   2
                     0
                                                   2
8
                     0
9
                                                   2
                                                   3
10
                     0
                                                   3
3
11
                     0
12
                     0
                                                   3
13
                     0
                                                   4
14
                     1
                                                   4
15
                     2
                     2
                                                   5
5
7
7
16
17
                     3
18
                     1
19
                     1
```

OPERATIONS ON DATASET

There are three operations we can perform on a dataset.

- 1. Row Operation
- 2. Column Operation
- 3. Selection Operation

ROW OPERATIONS-

- row selection
- row addition
- row deletion

ROW SELECTION-

```
#SELECTING ROW
x=b.iloc[0,]
print(x)
```

S.No			1
Name		Vá	agesh
Gender			m
City	ŀ	nydei	rabad
DateOfPurchase	01	jun	2020
Customer_Rating(0-5)			4
ManufactureDate	11	jun	2019
Condition			used
ModelNo			101
Mrp			2380
SoldPrice			2975
Manufacturing_Cost			230
Extend_Amount			23
Warranty_Period			1
<pre>Warranty_periodLeft(in Months)</pre>			0
MD/CN			0
Previous_claims			1
Refund_Duration(in days)			3
Profit			595
Profit_Percentage			25.0
Name: 0, dtype: object			

ROW ADDITION-

We can add the row for the dataset by using LOC() method

But the row data must be the same comparing to the other rows.

b.loc()

	S.No	Name	Gender	City	DateOfPurchase	\
0	1	vagesh	m	hyderabad	01 jun 2020	
1	2	venkat	m	pune	05 july 2020	
2	3	reddy	m	nuzvid	28 aug 2020	
3	4	yashwanth	m	vijayawada	25 sep 2020	
4	5	ram	m	mumbai	27 oct 2020	
5	6	kumar	m	kolkata	31 nov 2020	
6	7	abhi	f	bangalore	15 dec 2020	
7	8	bhavana	f	chennai	17 dec 2020	
8	9	meghana	f	delhi	04 jan 2021	
9	10	swapna	f	vizag	17 jan 2021	
10	11	rani	f	tirupati	21 jan 2021	
11	12	keerthana	f	nellore	14 feb 2021	
12	13	akash	m	kadapa	16 feb 2021	
13	14	siva	m	kurnool	25 feb 2021	
14	15	shankar	m	kakinada	27 feb 2021	
15	16	pradeep	m	srikakulam	01 mar 2021	
16	17	sameena	f	nizamabad	05 mar 2021	
17	18	samba	m	noida	10 mar 2021	
18	19	sai	m	khammam	14 mar 2021	
19	20	vamsi	m	nalgonda	18 mar 2021	
20	1	gut	m	vishakapatnam	01 jan 2023	

C	ustomer_Rating(0-5)	Manufact	ureDa	ate C	ondition	ModelNo	Mrp	SoldPrice \
0		4 11	. jun	2019	use	d 10	1 238	0 2975
1		4 15	july	2019	use	d 10	2 347	2 2536
2		4 18	aug	2019	use	d 10	3 312	7 2199
3		3 15	sep	2019	nei	N 10	4 270	4 3515
4		3 17	oct	2019	nei	N 10	5 257	5 3089
5		3 21	. nov	2019	use	d 10	6 321	0 2288
6		3 05	dec	2019	nei	N 10	7 299	9 2399
7		3 07	' dec	2019	nei	N 10	8 386	6 3966
8		5 14	jan	2020	nei	w 10	9 275	0 3575
9		5 07	'jan	2020	use	d 11	0 366	5 3765
10		4 11	. jan	2019	use	d 11	1 245	5 2946
11		4 04	feb	2019	nei	w 11	2 376	6 1883
12		5 06	feb	2019	use	d 11	3 394	4 3844
13		4 15	feb	2019	nei	w 11	4 208	5 2490
14		3 17	' feb	2019	nei	w 11	5 362	8 2711
15			. mar	2019	nei	w 11	6 299	2 3590
16		1 15	mar	2019	nei	w 11	7 359	9 2159
17		5 11	. mar	2019	use	d 11	8 212	2 2546
18		2 04	mar	2019	use	d 11	9 282	1 3385
19				2019		N 12	0 220	0 2100
20		5 02	: jan	2021	use	d	1	2 3
_	Manufacturing_Cost	Extend_	_		arranty_P			
0	230			23		1		
1	340			34		1		
2	310			31		1		
3	270			27		1		
4	250			25		1		
5	320			32		1		
6 7	290			29		1		
	380			38		1		
8 9	270			27 26		1		
9 10	360 240			36 24		1 1		
11	370			24 37		1		
12	390			39		1		
13	200			29 20		1		
13 14	360			20 36		1		
14 15	290			29		1		
16	350			29 35		1		
16 17	210			21		1		
18	2800			21 28		1		
19	220			20 22		1		
20	4		•	5		6		
20	4			,		J		

<pre>Warranty_periodLeft(in Months)</pre>	MD/CN	Previous_claims	\
0	0	0	1
1	2	0	0
2	4	0	0
3	0	0	2
4	6	1	1
5	8	0	1
6	9	1	1
7	4	1	0
8	7	1	0
9	5	0	0
10	8	1	0
11	0	1	0
12	0	1	0
13	6	1	0
14	5	1	1
15	4	1	2
16	2	1	2
17	2	0	3
18	1	0	1
19	1	0	1
20	7	8	9
Refund_Duration(in days	;)		

Refund_Duration(in days)

0	3
1	3
2	4
3	3
4	5
5	7
6	2
7	2
8	2
9	2
10	3
11	3
12	3
13	3
14	4
15	4
16	5
17	5
18	7
19	7
20	10

ROW DELETION-

By using drop() method we can delete the row.

b.drop(20)

	S.No	Name	Gender	City [DateOfI	Purcl	nase	<pre>Customer_Rating(0-5) \</pre>
0	1	vagesh	n m	hyderabad	01	jun	2020	4
1	2	venkat	: m	pune	0 5 <u>5</u>	july	2020	4
2	3	reddy	, m	nuzvid	28	aug	2020	4
3	4	yashwanth	n m	vijayawada	25	sep	2020	3 3
4	5	ram	n m	mumbai	27	oct	2020	3
5	6	kumar		kolkata	31	nov	2020	3
6	7	abhi		bangalore	15	dec	2020	3
7	8	bhavana		chennai			2020	3
8	9	meghana		delhi	04	jan	2021	5 5
9	10	swapna		vizag	17	jan	2021	5
10	11	rani		tirupati	21	jan	2021	4
11	12	keerthana	ı f	nellore	14	feb	2021	4
12	13	akash		kadapa			2021	5
13	14	siva		kurnool			2021	4
14	15	shankar	· m	kakinada	27	feb	2021	3
15	16	pradeep		srikakulam	01	mar	2021	2
16	17	sameena	ı f	nizamabad	05	mar	2021	1
17	18	samba	n m	noida	10	mar	2021	5
18	19	sai	. m	khammam			2021	2
19	20	vamsi	. m	nalgonda	18	mar	2021	1
	Manufa	ctureDate	Condition	ModelNo	Mrp	Sol	dPrice	<pre>Manufacturing_Cost \</pre>
0		ctureDate jun 2019	Condition used		Mrp 2380	Solo	dPrice 2975	<u>~_</u>
0 1	11			101	-	Sol		230
	11 15	jun 2019	used	101 102	2380	Solo	297	230 340
1	11 15 18	jun 2019 july 2019	used used	101 102 103	2380 3472	Solo	2975 2536	230 5 340 9 310
1 2	11 15 18 15	jun 2019 july 2019 aug 2019	used used used	101 102 103 104	2380 3472 3127	Solo	2975 2536 2199	230 340 310 3270
1 2 3	11 15 18 15 17	jun 2019 july 2019 aug 2019 sep 2019	used used used new	101 102 103 104 105	2380 3472 3127 2704	Solo	2975 2536 2199 3515	230 340 310 5 270 250
1 2 3 4	11 15 18 15 17 21	jun 2019 july 2019 aug 2019 sep 2019 oct 2019	used used used new new	101 102 103 104 105 106	2380 3472 3127 2704 2575	Solo	2975 2536 2199 3515 3089	230 340 310 310 320
1 2 3 4 5	11 15 18 15 17 21 05	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019	used used used new new used	101 102 103 104 105 106 107	2380 3472 3127 2704 2575 3210	Solo	2975 2536 2199 3515 3089 2288	230 340 310 310 3270 250 320 290
1 2 3 4 5 6	11 15 18 15 17 21 05	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019	used used used new new used new	101 102 103 104 105 106 107	2380 3472 3127 2704 2575 3210 2999	Solo	2975 2536 2199 3515 3089 2288 2399	230 340 310 270 250 3 320 290 380
1 2 3 4 5 6 7	11 15 18 15 17 21 05 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019	used used used new new used new	101 102 103 104 105 106 107 108 109	2380 3472 3127 2704 2575 3210 2999 3866	Solo	2975 2536 2199 3515 3089 2288 2399 3966	230 340 310 310 3270 320 320 320 320 380 380 370
1 2 3 4 5 6 7 8	11 15 18 15 17 21 05 07 14 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019	used used used new new used new new	101 102 103 104 105 106 107 108 109 110	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946	230 340 310 310 3270 250 320 290 380 270 360 360 240
1 2 3 4 5 6 7 8 9	11 15 18 15 17 21 05 07 14 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019 jan 2020	used used used new new used new new used	101 102 103 104 105 106 107 108 109 110	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765	230 340 310 310 3270 250 320 290 380 270 360 360 240
1 2 3 4 5 6 7 8 9 10	11 15 18 15 17 21 05 07 14 07 11	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019	used used new new used new new used used	101 102 103 104 105 106 107 108 109 110 111	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946	230 340 310 310 3270 320 320 320 320 320 320 320 320 320 32
1 2 3 4 5 6 7 8 9 10	11 15 18 15 17 21 05 07 14 07 11 04 06	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019	used used used new used new new used new used used used	101 102 103 104 105 106 107 108 109 110 111 112 113	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883	230 340 310 310 270 250 3 320 290 380 270 360 240 370 390
1 2 3 4 5 6 7 8 9 10 11 12 13 14	11 15 18 15 17 21 05 07 14 07 11 04 06 15	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019	used used used new used new new used new used used used new	101 102 103 104 105 106 107 108 109 110 111 112 113 114	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711	230 340 310 310 3270 250 380 290 380 270 360 360 4 390 4 390 4 390 5 200 360
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019	used used used new used new new used used used new used new	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596	230 340 310 310 270 250 320 290 380 270 360 240 370 4 390 200 4 360 290
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019 mar 2019	used used used new used new used used new used new used new used new	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596 2159	230 340 310 310 270 250 320 290 380 270 360 240 370 4 390 200 4 360 290 350
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17 11	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019 mar 2019 mar 2019	used used used new used new used used used new used new used	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599 2122	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596 2159 2159 2546	230 340 310 310 3270 250 320 290 380 270 360 240 370 4 390 200 4 360 290 350 290 350 210
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17 11 15	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019 mar 2019	used used used new used new used used new used new used new used new	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596 2159	230 340 310 310 3270 250 320 290 380 270 360 270 360 240 370 4 390 200 4 360 290 350 210 2800

```
Extend_Amount Warranty_Period Warranty_periodLeft(in Months)
                                                                                MD/CN \
0
                 23
                                                                                   0
                 34
                                                                           2
1
                                      1
                                                                                   0
2
                 31
                                      1
                                                                           4
                                                                                   0
                                                                           0
3
                 27
                                      1
                                                                                   0
4
                 25
                                      1
                                                                           6
                                                                                   1
5
                                      1
                                                                           8
                 32
                                                                                   0
                                                                           9
6
                 29
                                      1
                                                                                   1
7
                 38
                                      1
                                                                           4
                                                                                   1
                                                                           7
8
                 27
                                      1
                                                                                   1
9
                 36
                                      1
                                                                           5
                                                                                   0
10
                 24
                                      1
                                                                           8
                                                                                   1
11
                 37
                                      1
                                                                           0
                                                                                   1
                                                                           0
12
                                      1
                 39
                                                                                   1
                                                                           6
13
                 20
                                      1
                                                                                   1
                                                                           5
14
                                      1
                 36
                                                                                   1
                 29
                                                                           4
15
                                      1
                                                                                   1
16
                 35
                                      1
                                                                           2
                                                                                   1
17
                 21
                                      1
                                                                           2
                                                                                   0
18
                 28
                                      1
                                                                           1
                                                                                   0
19
                 22
                                      1
                                                                           1
                                                                                   0
      Previous_claims
                                 Refund_Duration(in days)
0
                     1
                                                   3
1
                    0
2
                    0
3
                                                   3
                     2
                                                   5
4
                     1
                                                   7
5
                     1
                                                   2
6
                    1
                                                   2
7
                    0
                                                   2
8
                    0
9
                                                   2
                                                   3
10
                    0
                                                   3
3
11
                    0
12
                     0
                                                   3
13
                    0
                                                   4
14
                    1
                                                   4
15
                    2
                    2
                                                   5
5
7
7
16
17
                     3
18
                    1
19
                    1
```

COLUMN OPERATIONS

- Column Selection
- Column Addition
- Column deletion

COLUMN SELECTION-

We can select the column by using the column name and the data frame.

DataFrameObj.['column name']

```
print(b['Mrp'])
0
      2380
1
      3472
2
      3127
3
      2704
      2575
4
5
      3210
6
      2999
7
      3866
8
      2750
9
      3665
10
      2455
11
      3766
12
      3944
13
      2085
14
      3628
15
      2992
16
      3599
17
      2122
18
      2821
19
      2200
Name: Mrp, dtype: int64
```

COLUMN ADDITION

We can add columns by dataframe[' ']

I want to calculate profit of the data frame and add it to the existing columns.

PROFIT= SELLING PRICE - COST PRICE

Here in our DataFrame SoldPrice-Mrp=Profit

```
b['Profit']=b['SoldPrice']-b['Mrp']
print(b)
```

	S.No	Name	Gender	City [DateOfI	Purch	nase	<pre>Customer_Rating(0-5)</pre>	\
0	1	vagesh	n m	hyderabad			2020	_	
1	2	venkat		pune	05 ·	july	2020	4	
2	3	reddy	, m	nuzvid	-	-	2020	4	
3	4	yashwanth		vijayawada	25	sep	2020	3	
4	5	ram	n m	mumbai		-	2020	3	
5	6	kumar	· m	kolkata	31	nov	2020	3	
6	7	abhi	. f	bangalore	15	dec	2020	3	
7	8	bhavana	ı f	chennai	17	dec	2020	3	
8	9	meghana	ı f	delhi	04	jan	2021	5	
9	10	swapna	ı f	vizag	17	jan	2021	5	
10	11	rani	. f	tirupati	21	jan	2021	4	
11	12	keerthana	ı f	nellore	14	feb	2021	4	
12	13	akash	n m	kadapa	16	feb	2021	5	
13	14	siva	n m	kurnool	25	feb	2021	4	
14	15	shankar	· m	kakinada	27	feb	2021	3	
15	16	pradeep	m :	srikakulam	01	mar	2021	2	
16	17	sameena	ı f	nizamabad	05	mar	2021	1	
17	18	samba	n m	noida	10	mar	2021	5	
18	19	sai	. m	khammam	14	mar	2021	2	
19	20	vamsi	. m	nalgonda	18	mar	2021	1	
a		ctureDate			Mrp	Solo	dPrice	<u>0=</u>	\
0	11	jun 2019	used	101	2380	Solo	2975	230	\
1	11 15	jun 2019 july 2019	used used	101 102	2380 3472	Solo	2975 2536	230 340	\
1 2	11 15 18	jun 2019 july 2019 aug 2019	used used used	101 102 103	2380 3472 3127	Solo	2975 2536 2199	230 5 340 310	\
1 2 3	11 15 18 15	jun 2019 july 2019 aug 2019 sep 2019	used used used new	101 102 103 104	2380 3472 3127 2704	Solo	2975 2536 2199 3515	230 340 310 3270	\
1 2 3 4	11 15 18 15 17	jun 2019 july 2019 aug 2019 sep 2019 oct 2019	used used used new new	101 102 103 104 105	2380 3472 3127 2704 2575	Solo	2975 2536 2199 3515 3089	230 340 310 310 3270 250	\
1 2 3 4 5	11 15 18 15 17 21	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019	used used used new new used	101 102 103 104 105 106	2380 3472 3127 2704 2575 3210	Solo	2975 2536 2199 3515 3089 2288	230 340 310 310 3270 250 33	\
1 2 3 4 5 6	11 15 18 15 17 21 05	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019	used used used new new used new	101 102 103 104 105 106 107	2380 3472 3127 2704 2575 3210 2999	Solo	2975 2536 2199 3515 3089 2288 2399	230 340 310 5 270 250 3 320 290	\
1 2 3 4 5 6 7	11 15 18 15 17 21 05	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019	used used used new new used new	101 102 103 104 105 106 107 108	2380 3472 3127 2704 2575 3210 2999 3866	Solo	2975 2536 2199 3515 3089 2288 2399 3966	230 340 310 310 270 250 320 290 380	\
1 2 3 4 5 6 7 8	11 15 18 15 17 21 05 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019 jan 2020	used used used new new used new new	101 102 103 104 105 106 107 108 109	2380 3472 3127 2704 2575 3210 2999 3866 2750	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575	230 340 310 310 3270 320 320 320 320 380 380 370	\
1 2 3 4 5 6 7 8	11 15 18 15 17 21 05 07 14	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019 jan 2020 jan 2020	used used used new new used new new used	101 102 103 104 105 106 107 108 109 110	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765	230 340 310 310 270 250 320 290 380 320 380 380 360	\
1 2 3 4 5 6 7 8 9 10	11 15 18 15 17 21 05 07 14 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019	used used used new new used new new used used used used	101 102 103 104 105 106 107 108 109 110	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946	230 340 310 5 270 250 3 320 290 5 380 270 5 360 240	\
1 2 3 4 5 6 7 8 9 10 11	11 15 18 15 17 21 05 07 14 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019	used used used new new used new new used new used used new	101 102 103 104 105 106 107 108 109 110 111	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883	230 340 310 5 270 250 8 320 290 5 380 270 5 360 240 370	\
1 2 3 4 5 6 7 8 9 10 11 12	11 15 18 15 17 21 05 07 14 07 11 04	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019	used used used new used new new used new used used used new	101 102 103 104 105 106 107 108 109 110 111 112 113	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844	230 340 310 310 270 250 3320 290 380 270 360 270 360 240 380 370	\
1 2 3 4 5 6 7 8 9 10 11 12 13	11 15 18 15 17 21 05 07 14 07 11 04 06	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019	used used used new used new new used used used new used new	101 102 103 104 105 106 107 108 109 110 111 112 113	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496	230 340 310 270 250 320 290 380 290 380 270 360 360 370 390 200	\
1 2 3 4 5 6 7 8 9 10 11 12 13 14	11 15 18 15 17 21 05 07 14 07 11 04 06 15	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019	used used used new used new new used used used new used new	101 102 103 104 105 106 107 108 109 110 111 112 113 114	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711	230 340 310 5 270 250 3 320 290 5 380 270 5 360 240 3 390 4 390 4 360	\
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019	used used used new new used new used used new used new used new used	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496	230 340 310 5 270 250 8 320 290 5 380 270 5 360 240 8 370 4 390 200 4 360 290	\
1 2 3 4 5 6 7 8 9 10 11 12 13 14	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019	used used used new used new new used used used new used new	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596	230 340 310 270 250 3 320 290 380 270 360 240 370 4 390 4 390 4 390 5 290 5 360 6 240 7 360 8 370 8 390 9 360 9 360 9 360 9 360 9 370 9 3	\
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17 11	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019 mar 2019	used used used new new used new used new used new used new used new new	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596 2159	230 340 310 270 250 320 290 380 290 380 270 360 370 390 200 360 290 350 290	\
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17 11 15	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019 mar 2019 mar 2019	used used used new new used new used used new used new used new used new used	101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599 2122	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596 2159 2546	230 340 310 310 3270 250 320 250 320 290 380 270 380 270 380 270 380 270 360 240 370 4 390 200 4 360 290 350 210 2800	\

	Extend_Amount	Warranty_Period	<pre>Warranty_periodLeft(in Months)</pre>	MD/CN \
0	23	1	0	0
1	34	1	2	0
2	31	1	4	0
3	27	1	0	0
4	25	1	6	1
5	32	1	8	0
6	29	1	9	1
7	38	1	4	1
8	27	1	7	1
9	36	1	5	0
10	24	1	8	1
11	37	1	0	1
12	39	1	0	1
13	20	1	6	1
14	36	1	5	1
15	29	1	4	1
16	35	_ 1	2	1
17	21	_ 1	2	9
18	28	1	1	Ø
19	22	1	1	0
17	22	_	1	J

	Previous_claims	Refund_Duration(in	days)	Profit
0	1		3	595
1	0		3	-936
2	0		4	-928
3	2		3	811
4	1		5	514
5	1		7	-922
6	1		2	-600
7	0		2	100
8	0		2	825
9	0		2	100
10	0		3	491
11	0		3	-1883
12	0		3	-100
13	0		3	405
14	1		4	-917
1 5	2		4	598
16	2		5	-1440
17	3		5	424
18	1		7	564
19	1		7	-100

Similarly, I want to calculate profit percentage and add it to the columns list.

Profit %= profit*100 / Cost Price

According to our code

```
b['Profit_Percentage']=b['Profit']*100/b['Mrp']
print(b)
```

	S.No	Name Ge	nder	City D	ateOfPurchase	<pre>Customer_Rating(0-5) \</pre>
0	1	vagesh	m	hyderabad	01 jun 2020	4
1	2	venkat	m	pune	05 july 2020	4
2	3	reddy	m	nuzvid	28 aug 2020	4
3	4	yashwanth	m	vijayawada	25 sep 2020	3
4	5	ram	m	mumbai	27 oct 2020	3
5	6	kumar	m	kolkata	31 nov 2020	3
6	7	abhi	f	bangalore	15 dec 2020	3
7	8	bhavana	f	chennai	17 dec 2020	3
8	9	meghana	f	delhi	04 jan 2021	5
9	10	swapna	f	vizag	17 jan 2021	5
10	11	rani	f	tirupati	21 jan 2021	4
11	12	keerthana	f	nellore	14 feb 2021	4
12	13	akash	m	kadapa	16 feb 2021	5
13	14	siva	m	kurnool	25 feb 2021	4
14	15	shankar	m	kakinada	27 feb 2021	3
15	16	pradeep	m	srikakulam	01 mar 2021	2
16	17	sameena	f	nizamabad	05 mar 2021	1
17	18	samba	m	noida	10 mar 2021	5
18	19	sai	m	khammam	14 mar 2021	2
19	20	vamsi	m	nalgonda	18 mar 2021	1

	ManufactureDate					Manufacturing_Co	
0	11 jun 2019		101		2975		230
1	15 july 2019		102		2536		840
2	18 aug 2019		103		2199		310
3	15 sep 2019		104		3515		270
4	17 oct 2019		105		3089		250
5	21 nov 2019		106		2288		320
6	05 dec 2019		107		2399		190
7	07 dec 2019		108		3966		880
8	14 jan 2020		109		3575		270
9	07 jan 2020	used used	116	3665	3765	3	860
10	11 jan 2019		111	. 2455	2946	2	240
11	04 feb 2019	new	112	3766	1883	3	370
12	06 feb 2019	used	113	3944	3844	3	390
13	15 feb 2019	new	114	2085	2490	2	200
14	17 feb 2019	new	115	3628	2711	3	860
15	11 mar 2019	new	116	2992	3590	2	290
16	15 mar 2019	new	117	3599	2159	3	350
17	11 mar 2019	used	118	3 2122	2546	2	210
18	04 mar 2019	used	119	2821	3385	28	800
19	28 mar 2019	new new	126	2200	2100	2	220
	Extend Amount	Warrantv	Period	Warrant	v periodLef	t(in Months) MD/	′CN \
0	Extend_Amount	Warranty_	=	Warrant	y_periodLef	t(in Months) MD/ 0	
0 1	23	: Warranty_	1	Warrant	y_periodLef	0	0
1	23 34	: Warranty_	1 1	Warrant	y_periodLef	0 2	0 0
1 2	23 34 31	: Warranty_	1 1 1	Warrant	y_periodLef	0 2 4	0 0 0
1 2 3	23 34 31 27	: Warranty_	1 1	Warrant	y_periodLef	0 2 4 0	0 0 0 0
1 2 3 4	23 34 31 27 25	: Warranty_	1 1 1 1	Warrant	y_periodLef	0 2 4 0 6	0 0 0 0 0
1 2 3 4 5	23 34 31 27 25 32	: Warranty_	1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8	0 0 0 0 1
1 2 3 4 5 6	23 34 31 27 25 32 29	: Warranty_	1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9	0 0 0 0 1 0
1 2 3 4 5 6 7	23 34 31 27 25 32 29 38	: Warranty_	1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4	0 0 0 0 1 0 1
1 2 3 4 5 6 7 8	23 34 31 27 25 32 29 38 27	: Warranty_	1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7	0 0 0 1 0 1 1
1 2 3 4 5 6 7 8	23 34 31 27 25 32 29 38 27 36	: Warranty_	1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5	0 0 0 1 0 1 1 1
1 2 3 4 5 6 7 8 9	23 34 31 27 25 32 29 38 27 36 24	: Warranty_	1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5	0 0 0 1 0 1 1 1 0 1
1 2 3 4 5 6 7 8 9 10	23 34 31 27 25 32 29 38 27 36 24	: Warranty_	1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5 8 0	0 0 0 1 0 1 1 1 1 1
1 2 3 4 5 6 7 8 9 10 11 12	23 34 31 27 25 32 29 38 27 36 24 37 39	: Warranty_	1 1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5 8 0 0	0 0 0 1 0 1 1 1 0 1
1 2 3 4 5 6 7 8 9 10 11 12 13	23 34 31 27 25 32 29 38 27 36 24 37 39 20	: Warranty_	1 1 1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5 8 0 0 6	0 0 0 1 0 1 1 1 0 1 1 1 1
1 2 3 4 5 6 7 8 9 10 11 12 13 14	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36	: Warranty_	1 1 1 1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5 8 0 0 6 5	0 0 0 1 0 1 1 1 1 1 1 1
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36 29	: Warranty_	1 1 1 1 1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5 8 0 0 6 5 4	0 0 0 1 0 1 1 1 1 1 1 1 1
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36 29 35	: Warranty_	1 1 1 1 1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5 8 0 0 6 5 4 2	0 0 0 1 0 1 1 1 1 1 1 1 1 1
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36 29 35 21	: Warranty_	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5 8 0 0 6 5 4 2 2	0 0 0 1 0 1 1 1 1 1 1 1 1
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36 29 35	: Warranty_	1 1 1 1 1 1 1 1 1 1 1 1 1	Warrant	y_periodLef	0 2 4 0 6 8 9 4 7 5 8 0 0 6 5 4 2	0 0 0 1 0 1 1 1 1 1 1 1 1 1

	Previous_claims	Refund_Duration(in	days)	Profit	Profit_Percentage
0	1		3	595	25.000000
1	0		3	-936	-26.958525
2	0		4	-928	-29.677007
3	2		3	811	29.992604
4	1		5	514	19.961165
5	1		7	-922	-28.722741
6	1		2	-600	-20.006669
7	0		2	100	2.586653
8	0		2	825	30.000000
9	0		2	100	2.728513
10	0		3	491	20.000000
11	0		3	-1883	-50.000000
12	0		3	-100	-2.535497
13	0		3	405	19.424460
14	1		4	-917	-25.275634
15	2		4	598	19.986631
16	2		5	-1440	-40.011114
17	3		5	424	19.981150
18	1		7	564	19.992910
19	1		7	-100	-4.545455

COLUMN DELETION –

To delete an entire column from a Pandas Series in Python, you can use the 'drop' method or simply select the columns you want to keep.

```
del b['Mrp']
print(b)
```

The taken dataset 'Mrp' column is removed.

The output of the code is given below.

	S.No	Name Ge	ender	City D	ateOfPurchase	Customer_Rating(0-5)	\
0	1	vagesh	m	hyderabad	01 jun 2020	4	
1	2	venkat	m	pune	05 july 2020	4	
2	3	reddy	m	nuzvid	28 aug 2020	4	
3	4	yashwanth	m	vijayawada	25 sep 2020	3	
4	5	ram	m	mumbai	27 oct 2020	3	
5	6	kumar	m	kolkata	31 nov 2020	3	
6	7	abhi	f	bangalore	15 dec 2020	3	
7	8	bhavana	f	chennai	17 dec 2020	3	
8	9	meghana	f	delhi	04 jan 2021	5	
9	10	swapna	f	vizag	17 jan 2021	5	
10	11	rani	f	tirupati	21 jan 2021	4	
11	12	keerthana	f	nellore	14 feb 2021	4	
12	13	akash	m	kadapa	16 feb 2021	5	
13	14	siva	m	kurnool	25 feb 2021	4	
14	15	shankar	m	kakinada	27 feb 2021	3	
15	16	pradeep	m	srikakulam	01 mar 2021	2	
16	17	sameena	f	nizamabad	05 mar 2021	1	
17	18	samba	m	noida	10 mar 2021	5	
18	19	sai	m	khammam	14 mar 2021	2	
19	20	vamsi	m	nalgonda	18 mar 2021	1	

	Manufactu	reDate	Condition	ModelNo	SoldPrice	<pre>Manufacturing_Cost \</pre>
0	11 jun	2019	used	101	2975	230
1	15 july	2019	used	102	2536	340
2	18 aug	2019	used	103	2199	310
3	15 sep	2019	new	104	3515	270
4	17 oct	2019	new	105	3089	250
5	21 nov	2019	used	106	2288	320
6	05 dec	2019	new	107	2399	290
7	07 dec	2019	new	108	3966	380
8	14 jan	2020	new	109	3575	270
9	07 jan	2020	used	110	3765	360
10	11 jan	2019	used	111	2946	240
11	04 feb	2019	new	112	1883	370
12	06 feb	2019	used	113	3844	390
13	15 feb	2019	new	114	2490	200
14	17 feb	2019	new	115	2711	360
15	11 mar	2019	new	116	3590	290
16	15 mar	2019	new	117	2159	350
17	11 mar	2019	used	118	2546	210
18	04 mar	2019	used	119	3385	2800
19	28 mar	2019	new	120	2100	220

	Extend_Amount	Warranty_Period	Warranty_periodLeft(in	Months)	MD/CN	\
0	23	1		0	0	
1	34	1		2	0	
2	31	1		4	0	
3	27	1		0	0	
4	25	1		6	1	
5	32	1		8	0	
6	29	1		9	1	
7	38	1		4	1	
8	27	1		7	1	
9	36	1		5	0	
10	24	1		8	1	
11	37	1		0	1	
12	39	1		0	1	
13	20	1		6	1	
14	36	1		5	1	
15	29	1		4	1	
16	35	1		2	1	
17	21	1		2	0	
18	28	1		1	0	
19	22	1		1	0	

	Previous_claims	Refund_Duration(in days)
0	1	3
1	0	3
2	0	4
3	2	3
4	1	5
5	1	7
6	1	2
7	0	2
8	0	2
9	0	2
10	0	3
11	0	3
12	0	3
13	0	3
14	1	4
15	2	4
16	2	5
17	3	5
18	1	7
19	1	7

INFO()

The info method provides a summary of the data including the data types of each column the number of non-null values.

print(b.info)

	S.No	Name Ge	nder	City D	ateOfPurchase	<pre>Customer_Rating(0-5)</pre>	\
0	1	vagesh	m	hyderabad	01 jun 2020	4	
1	2	venkat	m	pune	05 july 2020	4	
2	3	reddy	m	nuzvid	28 aug 2020	4	
3	4	yashwanth	m	vijayawada	25 sep 2020	3	
4	5	ram	m	mumbai	27 oct 2020	3	
5	6	kumar	m	kolkata	31 nov 2020	3	
6	7	abhi	f	bangalore	15 dec 2020	3	
7	8	bhavana	f	chennai	17 dec 2020	3	
8	9	meghana	f	delhi	04 jan 2021	5	
9	10	swapna	f	vizag	17 jan 2021	5	
10	11	rani	f	tirupati	21 jan 2021	4	
11	12	keerthana	f	nellore	14 feb 2021	4	
12	13	akash	m	kadapa	16 feb 2021	5	
13	14	siva	m	kurnool	25 feb 2021	4	
14	15	shankar	m	kakinada	27 feb 2021	3	
15	16	pradeep	m	srikakulam	01 mar 2021	2	
16	17	sameena	f	nizamabad	05 mar 2021	1	
17	18	samba	m	noida	10 mar 2021	5	
18	19	sai	m	khammam	14 mar 2021	2	
19	20	vamsi	m	nalgonda	18 mar 2021	1	

	Manufacture	eDate	Condition	ModelNo	Mrp	SoldPrice	Manufacturing_Cost	\
0	11 jun	2019	used	101	2380	2975	230	
1	15 july	2019	used	102	3472	2536	340	
2	18 aug	2019	used	103	3127	2199	310	
3	15 sep	2019	new	104	2704	3515	270	
4	17 oct	2019	new	105	2575	3089	250	
5	21 nov	2019	used	106	3210	2288	320	
6	05 dec	2019	new	107	2999	2399	290	
7	07 dec	2019	new	108	3866	3966	380	
8	14 jan	2020	new	109	2750	3575	270	
9	07 jan	2020	used	110	3665	3765	360	
10	11 jan	2019	used	111	2455	2946	240	
11	04 feb	2019	new	112	3766	1883	370	
12	06 feb	2019	used	113	3944	3844	390	
13	15 feb	2019	new	114	2085	2490	200	
14	17 feb	2019	new	115	3628	2711	360	
15	11 mar	2019	new	116	2992	3590	290	
16	15 mar	2019	new	117	3599	2159	350	
17	11 mar	2019	used	118	2122	2546	210	
18	04 mar	2019	used	119	2821	3385	2800	
19	28 mar	2019	new	120	2200	2100	220	

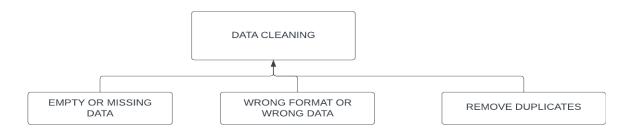
	Extend_Amount	Warranty_Period	Warranty_periodLeft(in	Months)	MD/CN	\
0	23	1		0	0	
1	34	1		2	0	
2	31	1		4	0	
3	27	1		0	0	
4	25	1		6	1	
5	32	1		8	0	
6	29	1		9	1	
7	38	1		4	1	
8	27	1		7	1	
9	36	1		5	0	
10	24	1		8	1	
11	37	1		0	1	
12	39	1		0	1	
13	20	1		6	1	
14	36	1		5	1	
15	29	1		4	1	
16	35	1		2	1	
17	21	1		2	0	
18	28	1		1	0	
19	22	1		1	0	

	Previous_claims	Refund_Duration(in o	days)	Profit	Profit_Percentage
0	1		3	595	25.000000
1	0		3	-936	-26.958525
2	0		4	-928	-29.677007
3	2		3	811	29.992604
4	1		5	514	19.961165
5	1		7	-922	-28.722741
6	1		2	-600	-20.006669
7	0		2	100	2.586653
8	0		2	825	30.000000
9	0		2	100	2.728513
10	0		3	491	20.000000
11	0		3	-1883	-50.000000
12	0		3	-100	-2.535497
13	0		3	405	19.424460
14	1		4	-917	-25.275634
15	2		4	598	19.986631
16	2		5	-1440	-40.011114
17	3		5	424	19.981150
18	1		7	564	19.992910
19	1		7	-100	-4.545455

3) CLEANING THE DATA

It is the process of remaining or replacing the NAN values.

NAN – Not A Null (or) Not A None



EMPTY OR MISSING DATA

We can solve empty cells by using two methods.

- isnull()
- notnull()

```
import pandas as pd
import numpy as np
a={'S.No':pd.Series([1,2,3,4,np.nan,6,7,8,9,10,11,12,np.nan,14,15,16,17,18,19,20]),
   'Name':pd.Series(['vagesh',np.nan,'reddy','yashwanth','ram','kumar','abhi',
   'bhavana', 'meghana', 'swapna', 'ram', 'keerthana', np.nan, 'siva', 'shankar', 'pradeep',
   'sameena','samba','sai','vamsi']),
   'Gender':pd.Series(['m','m',np.nan,'m','m','m','f','f','f',np.nan,'f','f',np.nan,
                        'm', 'm', 'm', 'f', np.nan, 'm', np.nan]),
   'City':pd.Series(['hyderabad','pune','mumbai','vijayawada','mumbai','kolkata',
       'bangalore', np.nan, 'delhi', 'vizag', 'tirupati', 'nellore', 'kadapa', 'kurnool',
        np.nan,'srikakulam','nizamabad',np.nan,'khammam','nalgonda',]),
   'DateOfPurchase':pd.Series(["01 jun 2020","05 july 2020","28 aug 2020",
              "25 sep 2020", "27 oct 2020", "31 nov 2020", "15 dec 2020", "17 dec 2020",
              "04 jan 2021","17 jan 2021","21 jan 2021","14 feb 2021","16 feb 2021",
              "25 feb 2021", "27 feb 2021", "01 mar 2021", "05 mar 2021", "10 mar 2021",
              "14 mar 2021","18 mar 2021"]),
   'Customer Rating(0-5)':([4,4,4,3,np.nan,3,3,3,5,5,4,np.nan,5,4,3,2,1,np.nan,2,1]),
   'ManufactureDate':(["11 jun 2019","15 july 2019","18 aug 2019","15 sep 2019",
          "17 oct 2019", "21 nov 2019", "05 dec 2019", "07 dec 2019", "14 jan 2020",
          "07 jan 2020","11 jan 2019","04 feb 2019","06 feb 2019","15 feb 2019",
          "17 feb 2019","11 mar 2019","15 mar 2019","11 mar 2019","04 mar 2019",
          "28 mar 2019"]),
   'Condition':(['used', 'used', np.nan, 'new', 'new', 'used', np.nan, 'new', 'new', 'used',
                 'used', 'new', np.nan, 'new', 'new', np.nan, 'new', 'used', 'used', 'new']),
   'ModelNo': ([101,102,103,103,105,106,107,108,109,110,111,111,np.nan,114,115,116,
               117,118,np.nan,120]),
   'Mrp':([2380, 3472, 3127, 2704, 2575, 3210, 2999, 3866, 2750, 3665, 2455, 3766,
            3944, 2085, 3628, 2992, 3599, 2122, 2821, 2200]),
   'SoldPrice':([2975, 2536, 2199, 3515, 3089, 2288, 2399, 3966, 3575, 3765, 2946,
                  1883, 3844, 2490, 2711, 3590, 2159, 2546, 3385, 2100]),
   'Manufacturing Cost': ([230,340,310,270,250,320,290,380,270,360,240,370,390,200,
                           360,290,350,210,2800,220]),
   'Extend Amount':([23,34,31,27,25,32,29,38,27,36,24,37,39,20,36,29,35,21,28,22]),
   'Warranty periodLeft(in Months)':([0,2,4,0,6,8,9,4,7,5,8,0,0,6,5,4,2,2,1,1]),
   'MD/CN': ([0,0,0,0,1,0,1,1,1,0,1,1,1,1,1,1,1,1,0,0,0]),
   'Previous claims':([1,0,0,2,1,1,1,0,0,0,0,0,0,0,1,2,2,3,1,1]),
   'Refund Duration(in days)':([3,3,np.nan,3,5,7,2,2,2,2,3,3,np.nan,3,4,4,5,5,7,7])
b=pd.DataFrame(a)
b['Profit']=b['SoldPrice']-b['Mrp']
b['Profit Percentage']=b['Profit']*100/b['Mrp']
print(b)
```

	S.No	Name	Gender	City [DateOf	Purcl	nase	<pre>Customer_Rating(0-5) \</pre>
0	1.0	vagesl	n m	hyderabad	01	jun	2020	4.0
1	2.0	Nal	N m	pune	0 5 <u></u>	july	2020	4.0
2	3.0	reddy	y NaN	mumbai	28	aug	2020	4.0
3	4.0	yashwanth	n m	vijayawada	25	sep	2020	3.0
4	NaN	rar	n m	mumbai	27	oct	2020	NaN
5	6.0	kuma	∩ m	kolkata	31	nov	2020	3.0
6	7.0	abh		bangalore	15	dec	2020	3.0
7	8.0	bhavana		NaN	17	dec	2020	3.0
8	9.0	meghana	a f	delhi	04	jan	2021	5.0
9	10.0	swapna	a NaN	vizag	17	jan	2021	5.0
10	11.0	rar		tirupati	21	jan	2021	4.0
11	12.0	keerthana	a f	nellore	14	feb	2021	NaN
12	NaN	Nal	N NaN	kadapa	16	feb	2021	5.0
13	14.0	siva	a m	kurnool	25	feb	2021	4.0
14	15.0	shankaı	r m	NaN	27	feb	2021	3.0
15	16.0	pradee	o m	srikakulam	01	mar	2021	2.0
16	17.0	sameena	a f	nizamabad	05	mar	2021	1.0
17	18.0	samba	a NaN	NaN	10	mar	2021	NaN
18	19.0	sa	i m	khammam	14	mar	2021	2.0
19	20.0	vams	i NaN	nalgonda	18	mar	2021	1.0
	Manufa	cturaData	Condition	n ModelNo	Mnn	Sol.	dDnica	Manufacturing Cost \
а		ctureDate			Mrp	Solo	dPrice	<u>5</u>
0	11	jun 2019	used	101.0	2380	Sol	2975	5 230
1	11 15	jun 2019 july 2019	used used	d 101.0 d 102.0	2380 3472	Solo	297 <u>9</u> 2536	5 230 5 340
1 2	11 15 18	jun 2019 july 2019 aug 2019	used used NaM	101.0 102.0 103.0	2380 3472 3127	Solo	2975 2536 2199	230 5 340 9 310
1 2 3	11 15 18 15	jun 2019 july 2019 aug 2019 sep 2019	used used NaN new	101.0 102.0 103.0 103.0	2380 3472 3127 2704	Solo	2975 2536 2199 3515	230 340 310 5 270
1 2 3 4	11 15 18 15 17	jun 2019 july 2019 aug 2019 sep 2019 oct 2019	used used NaN new new	101.0 102.0 103.0 103.0 105.0	2380 3472 3127 2704 2575	Solo	2975 2536 2199 3515 3089	230 340 310 5 270 250
1 2 3 4 5	11 15 18 15 17 21	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019	used used NaN new new used	101.0 102.0 103.0 103.0 105.0 106.0	2380 3472 3127 2704 2575 3210	Solo	2975 2536 2199 3515 3089 2288	230 340 310 310 270 250 3320
1 2 3 4 5 6	11 15 18 15 17 21 05	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019	used used NaN new new used NaN	101.0 102.0 103.0 103.0 105.0 106.0 107.0	2380 3472 3127 2704 2575 3210 2999	Solo	2975 2536 2199 3515 3089 2288 2399	230 340 310 5 270 250 320 290
1 2 3 4 5 6 7	11 15 18 15 17 21 05 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019	used used NaM new new used NaM new	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0	2380 3472 3127 2704 2575 3210 2999 3866	Solo	2975 2536 2199 3515 3089 2288 2399 3966	230 340 310 5 270 250 8 320 9 290 5
1 2 3 4 5 6 7 8	11 15 18 15 17 21 05 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019 jan 2020	used used NaM new used NaM new	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0	2380 3472 3127 2704 2575 3210 2999 3866 2750	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575	230 340 310 310 5 270 250 8 320 290 380 5 270
1 2 3 4 5 6 7 8	11 15 18 15 17 21 05 07 14	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 nov 2019 dec 2019 dec 2019 jan 2020 jan 2020	used used NaM new used NaM new used	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 110.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3769	230 340 310 5 270 250 3 320 290 5 380 270 360
1 2 3 4 5 6 7 8 9 10	11 15 18 15 17 21 05 07 14 07	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020	used used NaM new used NaM new used used	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 110.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946	230 340 310 5 270 250 8 320 9 290 5 380 5 270 360 240
1 2 3 4 5 6 7 8 9 10	11 15 18 15 17 21 05 07 14 07 11	jun 2019 july 2019 aug 2019 sep 2019 nov 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019	used used NaM new used NaM new used used new	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 110.0 111.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883	230 340 310 310 5 270 250 3 320 290 5 380 5 270 360 240 370
1 2 3 4 5 6 7 8 9 10 11 12	11 15 18 15 17 21 05 07 14 07 11 04	jun 2019 july 2019 aug 2019 sep 2019 nov 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019	used used NaM new used NaM new used used new NaM	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 110.0 111.0 NaN	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844	230 340 310 5 270 250 8 320 290 5 380 270 5 360 5 240 8 370 4
1 2 3 4 5 6 7 8 9 10 11 12 13	11 15 18 15 17 21 05 07 14 07 11 04 06 15	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019	used used NaM new used new used used new NaM new	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 111.0 111.0 NaN 114.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496	230 340 310 310 270 250 3 320 290 380 5 270 360 360 370 4 390 200
1 2 3 4 5 6 7 8 9 10 11 12 13 14	11 15 18 15 17 21 05 07 14 07 11 04 06 15	jun 2019 july 2019 aug 2019 sep 2019 nov 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019	used used NaM new used used used new NaM new new	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 110.0 111.0 NaN 114.0 115.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628	Solo	2975 2536 2199 3515 3085 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711	230 340 310 310 320 250 320 290 380 270 360 360 4 390 200 1 360
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019	used used NaM new used NaM new used new NaM new NaM	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 111.0 111.0 NaN 114.0 115.0 116.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596	230 340 310 310 5 270 250 3 320 290 5 380 5 270 5 360 6 340 370 4 390 4 390 10 11 12 13 14 16 16 16 16 16 16 16 16 16 16
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019 mar 2019	used used NaM new used new used new NaM new NaM new NaM	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 111.0 111.0 NaN 114.0 115.0 116.0 117.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2712 3596 2159	230 340 310 310 270 250 3 320 290 380 5 270 360 240 3 370 4 390 4 390 6 200 1 360 290 350
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17 11	jun 2019 july 2019 aug 2019 sep 2019 nov 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019 mar 2019 mar 2019	used used NaM new used used used new NaM new NaM new used	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 110.0 111.0 NaN 114.0 115.0 116.0 117.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599 2122	Solo	2975 2536 2199 3515 3085 2288 2399 3966 3575 3765 2946 1883 3844 2496 2711 3596 2159 2159 22546	230 340 310 310 320 250 320 290 380 270 380 270 360 370 4 390 200 1 360 290 350 290 350 210
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	11 15 18 15 17 21 05 07 14 07 11 04 06 15 17 11 15	jun 2019 july 2019 aug 2019 sep 2019 oct 2019 dec 2019 dec 2019 jan 2020 jan 2020 jan 2019 feb 2019 feb 2019 feb 2019 feb 2019 feb 2019 mar 2019 mar 2019	used used NaM new used new used new NaM new NaM new NaM	101.0 102.0 103.0 103.0 105.0 106.0 107.0 108.0 109.0 111.0 111.0 111.0 111.0 111.0 111.0 111.0 111.0 111.0 111.0 111.0 111.0 111.0	2380 3472 3127 2704 2575 3210 2999 3866 2750 3665 2455 3766 3944 2085 3628 2992 3599	Solo	2975 2536 2199 3515 3089 2288 2399 3966 3575 3765 2946 1883 3844 2496 2712 3596 2159	230 340 310 310 320 250 380 290 380 270 360 360 370 4 390 4 390 1 360 290 1 360 290 1 290 1 29

	Extend_Amount	Warranty_Period Wa	arranty_	periodLe	ft(in Months) MD/CN	\
0	23	1			0	0	
1	34	1			2	0	
2	31	1			4	0	
3	27	1			0	0	
4	25	1			6	1	
5	32	1			8	0	
6	29	1			9	1	
7	38	1			4	1	
8	27	1			7	1	
9	36	1			5	0	
10	24	1			8	1	
11	37	1			0	1	
12	39	1			0	1	
13	20	1			6	1	
14	36	1			5	1	
15	29	1			4	1	
16	35	1			2	1	
17	21	1			2	0	
18	28	1			1	0	
19	22	1			1	0	
	Previous_claims	Refund_Duration(ir	, days)	Profit	Profit_Perc	ontago	
0	1	Kerunu_Duracton(II	3.0	595		000000	
1	0		3.0	-936		958525	
2	0		NaN	-928		677007	
3	2		3.0	811		992604	
4	1		5.0	514		961165	
5	1		7.0	-922		722741	
6	1		2.0	-600		006669	
7	0		2.0	100		586653	
8	0		2.0	825		000000	
9	0		2.0	100		728513	
10	0		3.0	491		000000	
11	0		3.0	-1883		000000	
12	0		NaN	-100		535497	
13	0		3.0	405		424460	
14	1		4.0	-917		275634	
15	2		4.0	598		986631	
16	2		5.0	-1440		011114	
17	3		5.0	424		981150	
18	1		7.0	564		992910	
19	1		7.0	-100		545455	

print(b.isnull())

	S.No	Name	Gender	City	DateOfP	urchase	Customer_Ratin	g(0-5)	\
0	False	False	False	-		False	_	False	
1	False	True	False	False		False		False	
2	False	False	True	False		False		False	
3	False	False	False	False		False		False	
4	True	False	False	False		False		True	
5	False	False	False	False		False		False	
6	False	False	False	False		False		False	
7	False	False	False	True		False		False	
8	False	False	False	False		False		False	
9	False	False	True	False		False		False	
10	False	False	False	False		False		False	
11	False	False	False	False		False		True	
12	True	True	True	False		False		False	
13	False	False	False	False		False		False	
14	False	False	False	True		False		False	
15	False	False	False	False		False		False	
16	False	False	False	False		False		False	
17	False	False	True	True		False		True	
18	False	False	False	False		False		False	
19	False	False	True	False		False		False	
Ma	nufactu	noDato	Conditi	ion Mod	lo I No	Mnn Sol	dDnico Manufac	tunina	Cost \
	nufactu		Conditi			-	dPrice Manufac	turing_	
0	nufactu	Fal	.se	False	False	False	False	turing_	False
0 1	nufactu	Fal Fal	.se .se	False False	False False	False False	False False	turing_	False False
0 1 2	nufactu	Fal Fal Fal	.se .se .se	False False True	False False False	False False False	False False False	turing_	False False False
0 1 2 3	nufactu	Fal Fal Fal Fal	.se .se .se	False False True False	False False False False	False False False False	False False False False	turing_	False False False False
0 1 2 3 4	nufactu	Fal Fal Fal Fal	se se se se	False False True False False	False False False False False	False False False False False	False False False False False	turing_	False False False False False
0 1 2 3 4 5	nufactu	Fal Fal Fal Fal Fal	se se se se	False False True False False False	False False False False False	False False False False False	False False False False False False	turing_	False False False False False
0123456	nufactu	Fal Fal Fal Fal Fal	se se se se se se	False True False False False True	False False False False False False False	False False False False False False False	False False False False False False	turing_	False False False False False False False
01234567	nufactu	Fal Fal Fal Fal Fal Fal	se se se se se se	False False False False False False False True False	False False False False False False False False	False False False False False False False False	False False False False False False False False False	turing_	False False False False False False False False
 0 1 2 3 4 5 6 7 8 	nufactu	Fal Fal Fal Fal Fal Fal Fal	se se se se se se se	False True False False False True False False False	False False False False False False False False False	False	False	turing_	False False False False False False False False False
0 1 2 3 4 5 6 7 8	nufactu	Fal Fal Fal Fal Fal Fal Fal	se se se se se se se	False True False False True False True False False False False	False	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se	False True False False True False False False False False False False	False	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9 10	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se	False True False False True False False False False False False False	False	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9 10 11 12	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se	False False False False True False False False False False False True	False True	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9 10 11 12 13	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se se	False False False False True False False False False False False False False False	False	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se se	False False False False True False False False False False False True	False	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se se se	False True False True False	False	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se se se	False True False False False False False False	False	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se se se se	False True False	False	False	False	turing_	False
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	nufactu	Fal Fal Fal Fal Fal Fal Fal Fal Fal Fal	se se se se se se se se se se se	False True False False False False False False	False	False	False	turing_	False

	Extend_Amount	Warranty_Period	Warranty_	periodLe	ft(in Months)	MD/CN	\
0	False	False			False	False	
1	False	False			False	False	
2	False	False			False	False	
3	False	False			False	False	
4	False	False			False	False	
5	False	False			False	False	
6	False	False			False	False	
7	False	False			False	False	
8	False	False			False	False	
9	False	False			False	False	
10	False	False			False	False	
11	False	False			False	False	
12	False	False			False	False	
13	False	False			False	False	
14	False	False			False	False	
15	False	False			False	False	
16	False	False			False	False	
17	False	False			False	False	
18	False	False			False	False	
19	False	False			False	False	
	Previous_claims	Refund_Duration				_	
0	False	Refund_Duration	False	False	_	False	
1	_	Refund_Duration		False False	_	_	
1 2	False False False	Refund_Duration	False False True	False False False	-	False False False	
1 2 3	False False False False	Refund_Duration	False False True False	False False False False	_	False False False False	
1 2	False False False	Refund_Duration	False False True	False False False	_	False False False	
1 2 3 4 5	False False False False	Refund_Duration	False False True False False False	False False False False	-	False False False False	
1 2 3 4	False False False False False	Refund_Duration	False False False False False False	False False False False False	_	False False False False False	
1 2 3 4 5	False False False False False False	Refund_Duration	False False True False False False	False False False False False False	_	False False False False False False	
1 2 3 4 5 6	False False False False False False	Refund_Duration	False False False False False False False False False	False False False False False False False	-	False False False False False False False	
1 2 3 4 5 6 7	False False False False False False False False False	Refund_Duration	False False False False False False False False	False False False False False False False False	-	False False False False False False False False	
1 2 3 4 5 6 7 8	False	Refund_Duration	False	False		False	
1 2 3 4 5 6 7 8	False	Refund_Duration	False	False		False	
1 2 3 4 5 6 7 8 9	False	Refund_Duration	False	False		False	
1 2 3 4 5 6 7 8 9 10 11 12 13	False	Refund_Duration	False	False		False	
1 2 3 4 5 6 7 8 9 10 11 12 13 14	False	Refund_Duration	False True	False		False	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	False	Refund_Duration	False	False		False	
1 2 3 4 5 6 7 8 9 10 11 12 13 14	False	Refund_Duration	False	False		False	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	False	Refund_Duration	False	False		False	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	False	Refund_Duration	False	False		False	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	False	Refund_Duration	False	False		False	

print(b.notnull())

	S.No	Name	Gender	City	DateOfPurchase	Customer_Rating(0-5)	\
0	True	True	True	True	True	True	
1	True	False	True	True	True	True	
2	True	True	False	True	True	True	
3	True	True	True	True	True	True	
4	False	True	True	True	True	False	
5	True	True	True	True	True	True	
6	True	True	True	True	True	True	
7	True	True	True	False	True	True	
8	True	True	True	True	True	True	
9	True	True	False	True	True	True	
10	True	True	True	True	True	True	
11	True	True	True	True	True	False	
12	False	False	False	True	True	True	
13	True	True	True	True	True	True	
14	True	True	True	False	True	True	
15	True	True	True	True	True	True	
16	True	True	True	True	True	True	
17	True	True	False	False	True	False	
18	True	True	True	True	True	True	
19	True	True	False	True	True	True	

	ManufactureDate	Condition	ModelNo	Mrp	SoldPrice	Manufacturing_0	Cost \
0	Tru	e True	e True	True	. True	<u> </u>	True
1	Tru	e True	e True	True	True	<u> </u>	True
2	Tru	e False	e True	True	. True	<u> </u>	True
3	Tru	e True	e True	True	. True	<u> </u>	True
4	Tru	e True	e True	True	e True	<u> </u>	True
5	Tru	e True	e True	True	. True	<u>.</u>	True
6	Tru	e False	e True	True	. True	<u>.</u>	True
7	Tru	e True	e True	True	e True	<u>.</u>	True
8	Tru	e True	e True	True	e True	<u>.</u>	True
9	Tru	e True	e True	True	e True	<u>.</u>	True
1	0 Tru	e True	e True	True	e True	<u> </u>	True
1	1 Tru	e True	e True	True	e True	<u>.</u>	True
1	2 Tru	e False	e False	True	e True	<u>.</u>	True
1	3 Tru	e True	e True	True	e True	<u>.</u>	True
1	4 Tru	e True	e True	True	. True	<u>.</u>	True
1	5 Tru	e False	e True	True	e True	<u>.</u>	True
1	6 Tru	e True	e True	True	e True	<u>.</u>	True
1	7 Tru	e True	e True	True	e True	<u>.</u>	True
1	8 Tru	e True	e False	True	. True	<u>.</u>	True
1	9 Tru	e True	e True	True	. True	<u>)</u>	True

Extend_Amount Warranty_Period Warranty_periodLeft(in Months) MD/CN True True True True True True True True True True True True True	`
1TrueTrueTrue2TrueTrueTrue	
2 True True True True	
3 True True True True	
4 True True True True	
5 True True True True	
6 True True True True	
7 True True True True	
8 True True True True	
9 True True True True	
10 True True True True	
11 True True True True	
True True True True True	
True True True True	
14 True True True True	
True True True True True	
16 True True True True	
17 True True True True	
18 True True True True	
19 True True True True	

	Previous_claims	Refund_Duration(in days)	Profit	Profit_Percentage
0	True	True	True	True
1	True	True	True	True
2	True	False	True	True
3	True	True	True	True
4	True	True	True	True
5	True	True	True	True
6	True	True	True	True
7	True	True	True	True
8	True	True	True	True
9	True	True	True	True
10	True	True	True	True
11	True	True	True	True
12	True	False	True	True
13	True	True	True	True
14	True	True	True	True
15	True	True	True	True
16	True	True	True	True
17	True	True	True	True
18	True	True	True	True
19	True	True	True	True

WRONG FORMAT OR WRONG DATA

It means an individual column contains same datatype values, if not it doesn't perform any statistical operations.

There are two methods to handle the data .

- dropna() it drops all the NAN values
- fillna() it is used to fill in the NAN values

Fillna() is again divided into two types

16

35

Fillna(method='pad') – replaces NAN values with the previous values.

Fillna(method='bfill') – replaces NAN values with the next values.

pı	rint(b.	drop	ona())									
	S.No		Name	Gender	 City	DateOf	Purcl	nase	Customer_Rat	ing(0-5	5)	\
0	1.0	١	vagesh	n m	hyderabad	01	jun	2020		4	1.0	
3	4.0	yasł	hwanth	n m v	vijayawada	25	sep	2020		3	3.0	
5	6.0		kumar	n m	kolkata	31	nov	2020		3	3.0	
8	9.0	me	eghana	a f	delhi	04	jan	2021		5	.0	
10	11.0		ram	n f	tirupati	21	jan	2021		4	1.0	
13	14.0		siva	a m	kurnool	25	feb	2021		4	1.0	
16	17.0	Sã	ameena	a f	nizamabad	05	mar	2021		1	0	
				Condition		Mrp	Sol	dPrice		~-		\
0	11	jun	2019	used	101.0	2380		2975		23		
3		•	2019	new	103.0	2704		3515		27		
5	21	nov	2019	used	106.0	3210		2288		32	20	
8	14	jan	2020	new	109.0	2750		3575		27		
10		_	2019	used	111.0	2455		2946		24	10	
13			2019	new	114.0	2085		2490	1	20		
16	15	mar	2019	new	117.0	3599		2159		35	0	
	Exten	d_Amo	ount	Warranty_F	Period Wa	rranty	_per:	iodLef	t(in Months)) MD/CN	ı \	
0			23		1				6) 6)	
3			27		1				6) 6)	
5			32		1				8	3 6)	
8			27		1				7	7 1	_	
10			24		1				8	3 1	_	
13			20		1				6	5 1	L	
					_				_			

1

2

1

	Previous claims	Refund Duration(in days)	Profit	Profit Percentage
0	1	3.0	595	25.000000
U	т	5.0	293	23.000000
3	2	3.0	811	29.992604
5	1	7.0	-922	-28.722741
8	0	2.0	825	30.000000
10	0	3.0	491	20.000000
13	0	3.0	405	19.424460
16	2	5.0	-1440	-40.011114

```
print(b.fillna(method='pad'))
```

	S.No Name Gender		City Da	ateOfPurchase	<pre>Customer_Rating(0-5) \</pre>	
0	1.0	vagesh	m	hyderabad	01 jun 2020	4.0
1	2.0	vagesh	m	pune	05 july 2020	4.0
2	3.0	reddy	m	mumbai	28 aug 2020	4.0
3	4.0	yashwanth	m	vijayawada	25 sep 2020	3.0
4	4.0	ram	m	mumbai	27 oct 2020	3.0
5	6.0	kumar	m	kolkata	31 nov 2020	3.0
6	7.0	abhi	f	bangalore	15 dec 2020	3.0
7	8.0	bhavana	f	bangalore	17 dec 2020	3.0
8	9.0	meghana	f	delhi	04 jan 2021	5.0
9	10.0	swapna	f	vizag	17 jan 2021	5.0
10	11.0	ram	f	tirupati	21 jan 2021	4.0
11	12.0	keerthana	f	nellore	14 feb 2021	4.0
12	12.0	keerthana	f	kadapa	16 feb 2021	5.0
13	14.0	siva	m	kurnool	25 feb 2021	4.0
14	15.0	shankar	m	kurnool	27 feb 2021	3.0
15	16.0	pradeep	m	srikakulam	01 mar 2021	2.0
16	17.0	sameena	f	nizamabad	05 mar 2021	1.0
17	18.0	samba	f	nizamabad	10 mar 2021	1.0
18	19.0	sai	m	khammam	14 mar 2021	2.0
19	20.0	vamsi	m	nalgonda	18 mar 2021	1.0

	Manufacture	eDate	Condition	ModelNo	Mrp	SoldPrice	Manufacturing_Cost	\
0	11 jun	2019	used	101.0	2380	2975	230	
1	15 july	2019	used	102.0	3472	2536	340	
2	18 aug	2019	used	103.0	3127	2199	310	
3	15 sep	2019	new	103.0	2704	3515	270	
4	17 oct	2019	new	105.0	2575	3089	250	
5	21 nov	2019	used	106.0	3210	2288	320	
6	05 dec	2019	used	107.0	2999	2399	290	
7	07 dec	2019	new	108.0	3866	3966	380	
8	14 jan	2020	new	109.0	2750	3575	270	
9	07 jan	2020	used	110.0	3665	3765	360	
10	11 jan	2019	used	111.0	2455	2946	240	
11	04 feb	2019	new	111.0	3766	1883	370	
12	06 feb	2019	new	111.0	3944	3844	390	
13	15 feb	2019	new	114.0	2085	2490	200	
14	17 feb	2019	new	115.0	3628	2711	360	
15	11 mar	2019	new	116.0	2992	3590	290	
16	15 mar	2019	new	117.0	3599	2159	350	
17	11 mar	2019	used	118.0	2122	2546	210	
18	04 mar	2019	used	118.0	2821	3385	2800	
19	28 mar	2019	new	120.0	2200	2100	220	

	Extend_Amount	Warranty_Period	Warranty_periodLeft(in Months)	MD/CN	\
0	23	1	0	0	
1	34	1	2	0	
2	31	1	4	0	
3	27	1	0	0	
4	25	1	6	1	
5	32	1	8	0	
6	29	1	9	1	
7	38	1	4	1	
8	27	1	7	1	
9	36	1	5	0	
10	24	1	8	1	
11	37	1	0	1	
12	39	1	0	1	
13	20	1	6	1	
14	36	1	5	1	
15	29	1	4	1	
16	35	1	2	1	
17	21	1	2	0	
18	28	1	1	0	
19	22	1	1	0	

	Previous_claims	Refund_Duration(in	days)	Profit	Profit_Percentage
0	1		3.0	595	25.000000
1	0		3.0	-936	-26.958525
2	0		3.0	-928	-29.677007
3	2		3.0	811	29.992604
4	1		5.0	514	19.961165
5	1		7.0	-922	-28.722741
6	1		2.0	-600	-20.006669
7	0		2.0	100	2.586653
8	0		2.0	825	30.000000
9	0		2.0	100	2.728513
10	0		3.0	491	20.000000
11	0		3.0	-1883	-50.000000
12	0		3.0	-100	-2.535497
13	0		3.0	405	19.424460
14	1		4.0	-917	-25.275634
15	2		4.0	598	19.986631
16	2		5.0	-1440	-40.011114
17	3		5.0	424	19.981150
18	1		7.0	564	19.992910
19	1		7.0	-100	-4.545455

print(b.fillna(method='bfill'))

	S.No	Name Ge	nder	City D	ateOfPurchase	<pre>Customer_Rating(0-5) \</pre>
0	1.0	vagesh	m	hyderabad	01 jun 2020	4.0
1	2.0	reddy	m	pune	05 july 2020	4.0
2	3.0	reddy	m	mumbai	28 aug 2020	4.0
3	4.0	yashwanth	m	vijayawada	25 sep 2020	3.0
4	6.0	ram	m	mumbai	27 oct 2020	3.0
5	6.0	kumar	m	kolkata	31 nov 2020	3.0
6	7.0	abhi	f	bangalore	15 dec 2020	3.0
7	8.0	bhavana	f	delhi	17 dec 2020	3.0
8	9.0	meghana	f	delhi	04 jan 2021	5.0
9	10.0	swapna	f	vizag	17 jan 2021	5.0
10	11.0	ram	f	tirupati	21 jan 2021	4.0
11	12.0	keerthana	f	nellore	14 feb 2021	5.0
12	14.0	siva	m	kadapa	16 feb 2021	5.0
13	14.0	siva	m	kurnool	25 feb 2021	4.0
14	15.0	shankar	m	srikakulam	27 feb 2021	3.0
15	16.0	pradeep	m	srikakulam	01 mar 2021	2.0
16	17.0	sameena	f	nizamabad	05 mar 2021	1.0
17	18.0	samba	m	khammam	10 mar 2021	2.0
18	19.0	sai	m	khammam	14 mar 2021	2.0
19	20.0	vamsi	NaN	nalgonda	18 mar 2021	1.0

	Manufacture	Data	Condition	ModolNo	Mnn	ColdDaico	Manufacturi	na Cost	`
0	11 jun		used	101.0	Mrp 2380	2975	Manufacturi	230	\
1	15 july		used	102.0	3472	2536		340	
2	18 aug			102.0	3127	2199		310	
3	15 aug 15 sep		new	103.0	2704	3515		270	
	•		new						
4	17 oct		new	105.0	2575	3089		250	
5	21 nov		used	106.0	3210	2288		320	
6	05 dec		new	107.0	2999	2399		290	
7	07 dec		new	108.0	3866	3966		380	
8	14 jan		new	109.0	2750	3575		270	
9	07 jan		used	110.0	3665	3765		360	
10	11 jan		used	111.0	2455	2946		240	
11	04 feb		new	111.0	3766	1883		370	
12	06 feb		new	114.0	3944	3844		390	
13	15 feb		new	114.0	2085	2490		200	
14	17 feb		new	115.0	3628	2711		360	
15	11 mar		new	116.0	2992	3590		290	
16	15 mar		new	117.0	3599	2159		350	
17	11 mar		used	118.0	2122	2546		210	
18	04 mar		used	120.0	2821	3385		2800	
19	28 mar	2019	new	120.0	2200	2100		220	
	Extend_Amo	unt	Warranty_P	eriod Wa	rranty	_periodLeft	(in Months)	MD/CN	\
0	Extend_Amo	unt 23	Warranty_P	eriod Wa 1	rranty	_periodLeft	(in Months) 0	MD/CN 0	\
0 1	Extend_Amo		Warranty_P		rranty	_periodLeft			\
	Extend_Amo	23	Warranty_P	1	rranty	_periodLeft	0	0	\
1	Extend_Amo	23 34	Warranty_P	1 1	rranty	_periodLeft	0 2	0 0	\
1 2	Extend_Amo	23 34 31	Warranty_P	1 1 1	rranty	_periodLeft	0 2 4	0 0 0	\
1 2 3	Extend_Amo	23 34 31 27	Warranty_P	1 1 1	rranty	_periodLeft	0 2 4 0	0 0 0	\
1 2 3 4	Extend_Amo	23 34 31 27 25	Warranty_P	1 1 1 1	rranty	_periodLeft	0 2 4 0 6	0 0 0 0 1	\
1 2 3 4 5	Extend_Amo	23 34 31 27 25 32	Warranty_P	1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8	0 0 0 0 1 0	\
1 2 3 4 5 6	Extend_Amo	23 34 31 27 25 32 29	Warranty_P	1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9	0 0 0 0 1 0	\
1 2 3 4 5 6 7	Extend_Amo	23 34 31 27 25 32 29 38	Warranty_P	1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9	0 0 0 0 1 0 1	\
1 2 3 4 5 6 7 8	Extend_Amo	23 34 31 27 25 32 29 38 27	Warranty_P	1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7	0 0 0 1 0 1 1	\
1 2 3 4 5 6 7 8	Extend_Amo	23 34 31 27 25 32 29 38 27 36	Warranty_P	1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5	0 0 0 1 0 1 1 1	
1 2 3 4 5 6 7 8 9 10	Extend_Amo	23 34 31 27 25 32 29 38 27 36 24	Warranty_P	1 1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5	0 0 0 1 0 1 1 1 0	
1 2 3 4 5 6 7 8 9 10	Extend_Amo	23 34 31 27 25 32 29 38 27 36 24 37	Warranty_P	1 1 1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5 8	0 0 0 1 0 1 1 1 0	
1 2 3 4 5 6 7 8 9 10 11 12	Extend_Amo	23 34 31 27 25 32 29 38 27 36 24 37 39	Warranty_P	1 1 1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5 8 0 0	0 0 0 1 0 1 1 1 1 1	
1 2 3 4 5 6 7 8 9 10 11 12 13	Extend_Amo	23 34 31 27 25 32 29 38 27 36 24 37 39 20	Warranty_P	1 1 1 1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5 8 0 6	0 0 0 1 0 1 1 1 1 1	
1 2 3 4 5 6 7 8 9 10 11 12 13 14	Extend_Amo	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36	Warranty_P	1 1 1 1 1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5 8 0 6 6 5	0 0 0 1 0 1 1 1 1 1 1	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	Extend_Amo	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36 29	Warranty_P	1 1 1 1 1 1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5 8 0 6 6 9	0 0 0 1 0 1 1 1 1 1 1	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	Extend_Amo	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36 29 35	Warranty_P	1 1 1 1 1 1 1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5 8 0 6 5 4 2	0 0 0 1 0 1 1 1 1 1 1 1	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17	Extend_Amo	23 34 31 27 25 32 29 38 27 36 24 37 39 20 36 29 35 21	Warranty_P	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	rranty	_periodLeft	0 2 4 0 6 8 9 4 7 5 8 0 6 5 4 2 2	0 0 0 1 0 1 1 1 1 1 1 1	

	Previous_claims	Refund_Duration(in days)	Profit	Profit_Percentage
0	1	3.0	595	25.000000
1	0	3.0	-936	-26.958525
2	0	3.0	-928	-29.677007
3	2	3.0	811	29.992604
4	1	5.0	514	19.961165
5	1	7.0	-922	-28.722741
6	1	2.0	-600	-20.006669
7	0	2.0	100	2.586653
8	0	2.0	825	30.000000
9	0	2.0	100	2.728513
10	0	3.0	491	20.000000
11	0	3.0	-1883	-50.000000
12	0	3.0	-100	-2.535497
13	0	3.0	405	19.424460
14	1	4.0	-917	-25.275634
15	2	4.0	598	19.986631
16	2	5.0	-1440	-40.011114
17	3	5.0	424	19.981150
18	1	7.0	564	19.992910
19	1	7.0	-100	-4.545455

REMOVE DUPLICATES

print(b.drop_duplicates)

	S.No	Name Ge	ender	City Da	ateOfPurchase Custo	<pre>mer_Rating(0-5) \</pre>
0	1.0	vagesh	m	hyderabad	01 jun 2020	4.0
1	2.0	NaN	m	pune	05 july 2020	4.0
2	3.0	reddy	NaN	mumbai	28 aug 2020	4.0
3	4.0	yashwanth	m	vijayawada	25 sep 2020	3.0
4	NaN	ram	m	mumbai	27 oct 2020	NaN
5	6.0	kumar	m	kolkata	31 nov 2020	3.0
6	7.0	abhi	f	bangalore	15 dec 2020	3.0
7	8.0	bhavana	f	NaN	17 dec 2020	3.0
8	9.0	meghana	f	delhi	04 jan 2021	5.0
9	10.0	swapna	NaN	vizag	17 jan 2021	5.0
10	11.0	ram	f	tirupati	21 jan 2021	4.0
11	12.0	keerthana	f	nellore	14 feb 2021	NaN
12	NaN	NaN	NaN	kadapa	16 feb 2021	5.0
13	14.0	siva	m	kurnool	25 feb 2021	4.0
14	15.0	shankar	m	NaN	27 feb 2021	3.0
15	16.0	pradeep	m	srikakulam	01 mar 2021	2.0
16	17.0	sameena	f	nizamabad	05 mar 2021	1.0
17	18.0	samba	NaN	NaN	10 mar 2021	NaN
18	19.0	sai	m	khammam	14 mar 2021	2.0
19	20.0	vamsi	NaN	nalgonda	18 mar 2021	1.0

	ManufacturaData	Condition	ModolNo	Mnn	ColdDoico Ma	nufacturina	Cost	`
Ω	ManufactureDate		ModelNo	Mrp	SoldPrice Ma	nutactur.ing_	_	۱ (
0	11 jun 2019	used	101.0		2975		23	
1	15 july 2019	used	102.0		2536		34	
2	18 aug 2019	NaN	103.0		2199		31	
3	15 sep 2019	new	103.0		3515		27	
4	17 oct 2019	new	105.0		3089		25	
5	21 nov 2019	used	106.0		2288			20
6	05 dec 2019	NaN	107.0		2399		29	
7	07 dec 2019	new	108.0		3966		38	
8	14 jan 2020	new	109.0		3575		27	
9	07 jan 2020	used	110.0		3765		36	
10	11 jan 2019	used	111.0		2946		24	
11	04 feb 2019	new	111.0		1883		37	
12	06 feb 2019	NaN	NaN		3844			90
13	15 feb 2019	new	114.0		2490		26	
14	17 feb 2019	new	115.0	3628	2711		36	50
15	11 mar 2019	NaN	116.0	2992	3590		29	90
16	15 mar 2019	new	117.0	3599	2159		35	50
17	11 mar 2019	used	118.0	2122	2546		21	LØ
18	04 mar 2019	used	NaN	2821	3385		286	90
19	28 mar 2019	new	120.0	2200	2100		220	
	Extend_Amount	Warranty	Dariod W	lannanti	/_periodLeft(i	n Months) 1	MD/CN	\
0	23	warrancy_	1	iai i aiic	_periodlerc(i	0		o 0
1	34		1			2		0
2	31		1			4		0
3	27		1			9		0
4	25		1			6		1
5	32							
6	29		1 1			8		0
						9		1
7	38		1			4		1
8	27		1			7		1
9	36		1			5		0
10	24		1			8		1
11	37		1			0		1
12	39		1			0		1
13	20		1			6		1
14	36		1			5		1
15	29		1			4		1
16								
	35		1			2		1
17	35 21		1 1			2		0
	35		1					

	Previous_claims	Refund_Duration(in	days)	Profit	Profit_Percentage
0	1		3.0	595	25.000000
1	0		3.0	-936	-26.958525
2	0		NaN	-928	-29.677007
3	2		3.0	811	29.992604
4	1		5.0	514	19.961165
5	1		7.0	-922	-28.722741
6	1		2.0	-600	-20.006669
7	0		2.0	100	2.586653
8	0		2.0	825	30.000000
9	0		2.0	100	2.728513
10	0		3.0	491	20.000000
11	0		3.0	-1883	-50.000000
12	0		NaN	-100	-2.535497
13	0		3.0	405	19.424460
14	1		4.0	-917	-25.275634
15	2		4.0	598	19.986631
16	2		5.0	-1440	-40.011114
17	3		5.0	424	19.981150
18	1		7.0	564	19.992910
19	1		7.0	-100	-4.545455

4) ANALYZING THE DATA

In this section we need to perform or manipulate statistical operations. This is the major step in the entire data analysis part in which we can perform different statistical tool operations. These are known as Computational Tools.

Min()

Used to find the minimum value in the entire column.

```
print(b['Mrp'].min())
```

2085

Max()

Used to find the maximum value in the entire column.

```
print(b['Mrp'].max())
```

3944

```
print(b['SoldPrice'].max())
```

3966

Rank()

Gives a rank to the columns or entire data frame according to the ascending order or descending order for Numerical values, for alphabets it follows alphabetical order.

```
print(b['Mrp'].rank())
0
       4.0
1
      14.0
2
      12.0
3
       7.0
4
       6.0
5
      13.0
6
      11.0
7
      19.0
8
       8.0
9
      17.0
10
       5.0
11
      18.0
12
      20.0
13
       1.0
14
      16.0
15
      10.0
16
      15.0
17
       2.0
18
       9.0
19
       3.0
Name: Mrp, dtype: float64
```

CORRELATION-

- It is a relation between two data column data members.
- We use method called corr()
- It is scaled form of a covariance.
- Correlation values lies between (-1 to +1)

Attribute1.corr(attribute2)

Types of correlations:

we have three types of correlation.

- Positive (0 to 1)
- Negative (0 to -1)
- No correlation (0)

```
print(b['Extend_Amount'].corr(b['MD/CN']))
```

0.22010645891035724

```
print(b['Mrp'].corr(b['Extend_Amount']))
```

0.9986407965884486

```
print(b['SoldPrice'].corr(b['Extend_Amount']))
```

0.1836663464733947

```
print(b['Previous_claims'].corr(b['MD/CN']))
```

-0.20751433915982234

```
print(b['SoldPrice'].corr(b['MD/CN']))
```

0.12165190710678243

COVARIANCE-

The covariance is the relation between two data members of two different columns.

- It is the measure of a correlation.
- It lies between $(-\infty \text{ to } \infty)$.

```
print(b['Mrp'].cov(b['SoldPrice']))
```

73196.52631578947

```
print(b['Mrp'].cov(b['Extend_Amount']))
```

3646.9473684210525

```
print(b['SoldPrice'].cov(b['Extend_Amount']))
```

724.8605263157896

```
print(b['Previous_claims'].cov(b['MD/CN']))
```

-0.09473684210526313

```
print(b['SoldPrice'].cov(b['MD/CN']))
```

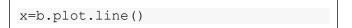
40.7078947368421

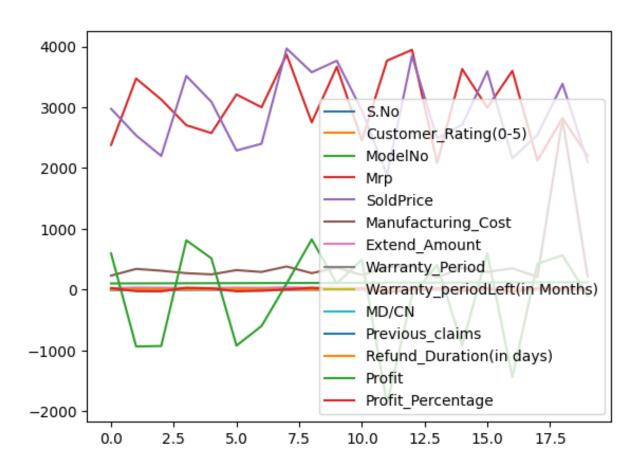
5) VISUALIZING THE DATA & SHARING THE RESULT

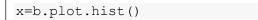
Data visualization is a process of representing data in a graphical way. Here we can represent the below graph formats.

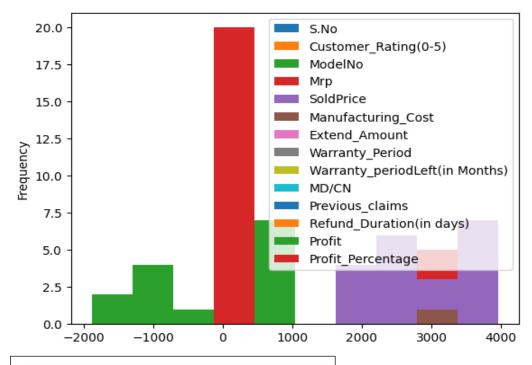
- 1. Line Graph
- 2. Bar Graph
- 3. Box Graph
- 4. KDE Graph
- 5. Area Graph
- 6. Histogram Graph

Here we need to use matplotlib module to represent graphs using the code.

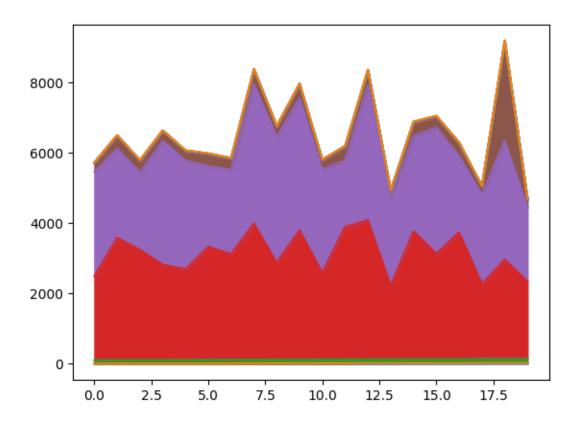








x=b.plot.area()



OUTPUT

CORELATION AND CO-VARIANCE ARE MANUALLY CALCULATED

Find correlation Blw Extend Amount and Molant
$$S = \frac{n \pm \alpha y}{\sqrt{(n \pm x^2 - (\pm x)^2)}} = \frac{20(339)}{\sqrt{(n \pm x^2 - (\pm x)^2)}} = \frac{20(339)}{\sqrt{(20(13271) - (593)^2)}} = \frac{20(339)}{\sqrt{(20(13271) - (593)^2)}} = \frac{6740 - 6543}{\sqrt{(265420 - 351649)}} = \frac{257}{\sqrt{(13770)}} = \frac{257}{\sqrt{(13770)}}$$

```
Find
      correlation
                 the SoldPrice and Extend Amand.
        8= n(2xy) -(2x)(2y)
           (nex2-(Exx) (ney - (2y))
          = 20(1732316) - (57961) (593)
          (20(176140103)-(57961)2) (20(1827)-(5935)
          34646320-34370873
          1 (3522802060 - 3359477521)
                          (365420 - (351649)
           1(16332 4539)(13771)
          275447
         J 163310768
          275447 = 0.183666359
                                        HIGHLY
                                        POSITIVE
         122779.31
```

Find correlation Hw Premium claim and MO (CN)
$$8 = \frac{n(2xy) - (2x)(2y)}{(2xy)(2y)}$$

$$= 20(7) - 16(1)$$

$$- (20(27) - (16)^2)(20(1) - (11)^2)$$

$$= 40 - 176$$

$$\sqrt{(560 - 256)(220 - 127)}$$

$$= -36$$

$$\sqrt{(304)(99)}$$

$$= -36$$

$$\sqrt{(3096)}$$

$$=$$

Covariance blue Mrp and Extend Amount

$$\text{QV}(x, y) = \underbrace{x(x_i - \overline{x})(y_i - \overline{y})}$$
 N-I
 $= \underbrace{(57342)(563.35)}$
 $= \underbrace{1305468.7}$
 $= 3646.947$

SUMMARY

In this project, we aimed to develop a warranty claim fraud detection system using data analytics, with a focus on time series analysis techniques implemented in Python.

Warranty claim fraud is a significant issue for many businesses, and detecting fraudulent claims is essential to minimize financial losses and maintain trust with customers.

Benefits:

- Improved fraud detection capabilities, leading to cost savings for the organization.
- Enhanced customer trust and satisfaction by reducing fraudulent claims.
- Data-driven decision-making for warranty claim management.