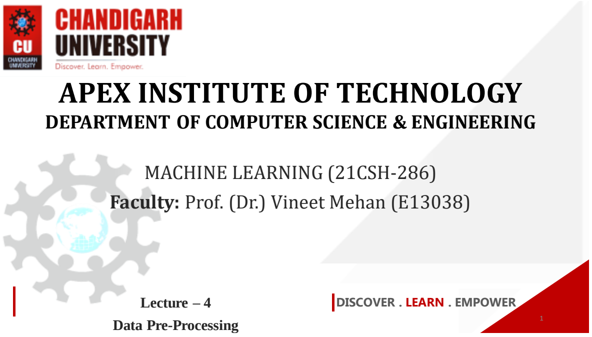


CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

APEX INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

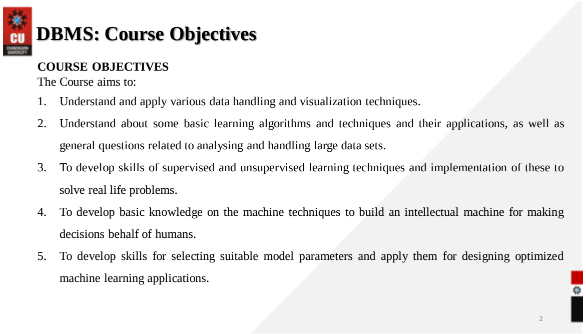
MACHINE LEARNING (21CSH-286)

Faculty: Prof. (Dr.) Vineet Mehan (E13038)

Lecture – 4

Data Pre-Processing

DISCOVER . LEARN . EMPOWER

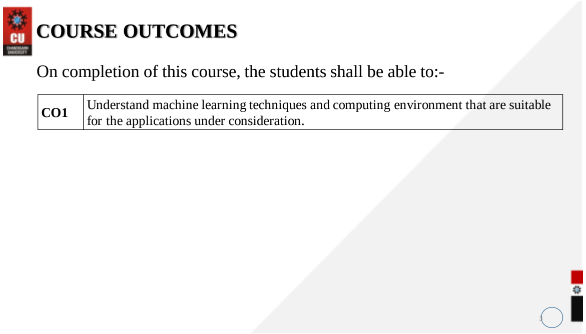


DBMS: Course Objectives

COURSE OBJECTIVES

The Course aims to:

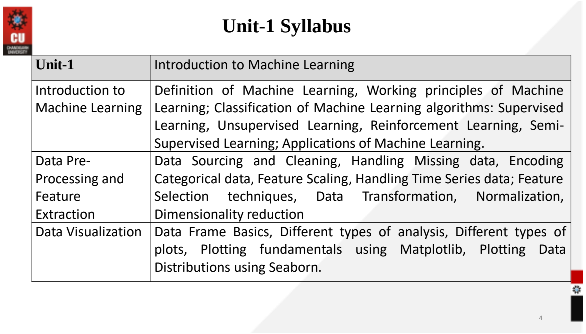
1. Understand and apply various data handling and visualization techniques.
2. Understand about some basic learning algorithms and techniques and their applications, as well as general questions related to analysing and handling large data sets.
3. To develop skills of supervised and unsupervised learning techniques and implementation of these to solve real life problems.
4. To develop basic knowledge on the machine techniques to build an intellectual machine for making decisions behalf of humans.
5. To develop skills for selecting suitable model parameters and apply them for designing optimized machine learning applications.



COURSE OUTCOMES

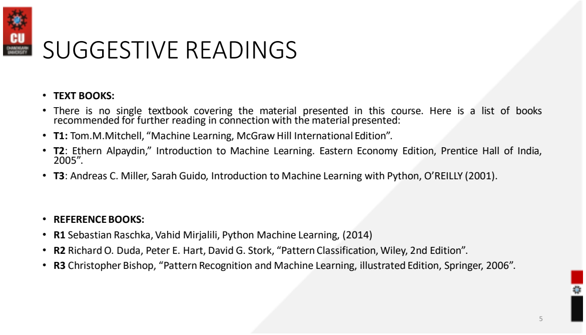
On completion of this course, the students shall be able to:-

CO1	Understand machine learning techniques and computing environment that are suitable for the applications under consideration.



Unit-1 Syllabus

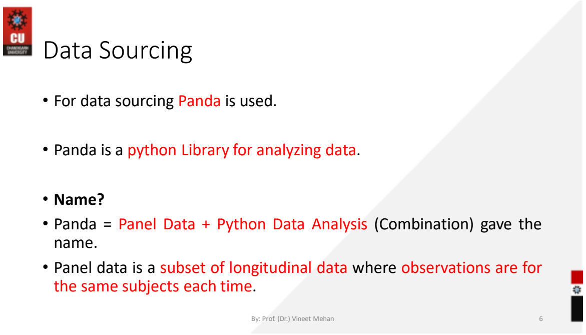
Unit-1	Introduction to Machine Learning
Introduction to Machine Learning	Definition of Machine Learning, Working principles of Machine Learning; Classification of Machine Learning algorithms: Supervised Learning, Unsupervised Learning, Reinforcement Learning, Semi-Supervised Learning; Applications of Machine Learning.
Data Pre-Processing and Feature Extraction	Data Sourcing and Cleaning, Handling Missing data, Encoding Categorical data, Feature Scaling, Handling Time Series data; Feature Selection techniques, Data Transformation, Normalization, Dimensionality reduction
Data Visualization	Data Frame Basics, Different types of analysis, Different types of plots, Plotting fundamentals using Matplotlib, Plotting Data Distributions using Seaborn.



SUGGESTIVE READINGS

- **TEXT BOOKS:**
- There is no single textbook covering the material presented in this course. Here is a list of books recommended for further reading in connection with the material presented:
- **T1:** Tom.M.Mitchell, “Machine Learning, McGraw Hill International Edition”.
- **T2**: Ethern Alpaydin,” Introduction to Machine Learning. Eastern Economy Edition, Prentice Hall of India, 2005”.
- **T3**: Andreas C. Miller, Sarah Guido, Introduction to Machine Learning with Python, O’REILLY (2001).

- **REFERENCE BOOKS:**
- **R1** Sebastian Raschka, Vahid Mirjalili, Python Machine Learning, (2014)
- **R2** Richard O. Duda, Peter E. Hart, David G. Stork, “Pattern Classification, Wiley, 2nd Edition”.
- **R3** Christopher Bishop, “Pattern Recognition and Machine Learning, illustrated Edition, Springer, 2006”.



Data Sourcing

- For data sourcing Panda is used.
- Panda is a python Library for analyzing data.
- **Name?**
- Panda = Panel Data + Python Data Analysis (Combination) gave the name.
- Panel data is a subset of longitudinal data where observations are for the same subjects each time.

Data Sourcing

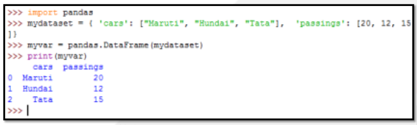
- Use of Panda ?
- Pandas allow us to analyze big data and make conclusions based on statistical theories.
- Pandas can clean messy data sets, and make them readable and relevant.
- Pandas are used in Data Science.

Data Sourcing

- Data Science: is a branch of computer science where we study how to store, use and analyze data for deriving information from it.
- How to install Pandas?
- 1. Open cmd prompt
- 2. Type
- >>> python –m pip install pandas

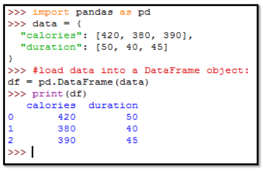
Make a data Frame that tells the type of vehicles that passed a toll plaza.

- import pandas
- mydataset = { 'cars': ["Maruti", "Hundai", "Tata"], 'passings': [20, 12, 15]}
- myvar = pandas.DataFrame(mydataset)
- print(myvar)



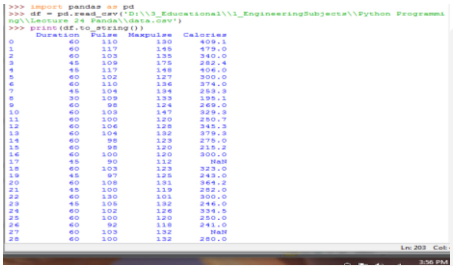
```
>>> import pandas
>>> mydataset = { 'cars': ["Maruti", "Hundai", "Tata"],  'passings': [20, 12, 15
]}
>>> myvar = pandas.DataFrame(mydataset)
>>> print(myvar)
     cars  passings
0  Maruti        20
1  Hundai        12
2    Tata        15
>>> |
```

Import pandas as pd and use pd



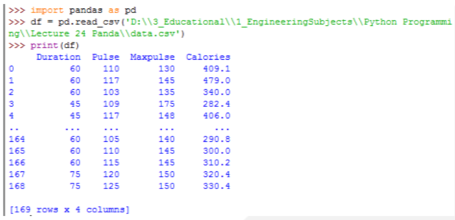
```
>>> import pandas as pd
>>> data = {
  "calories": [420, 380, 390],
  "duration": [50, 40, 45]
}
>>> #load data into a DataFrame object:
df = pd.DataFrame(data)
>>> print(df)
   calories  duration
0       420        50
1       380        40
2       390        45
>>> |
```

Read data from a CSV File



```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python Programmi
ng\\Lecture 24 Panda\\data.csv')
>>> print(df.to_string())
     Duration  Pulse  Maxpulse  Calories
0          60    110       130     409.1
1          60    117       145     479.0
2          60    103       135     340.0
3          45    109       175     282.4
4          45    117       148     406.0
5          60    102       127     300.0
6          60    110       136     374.0
7          45    104       134     253.3
8          30    109       133     195.1
9          60     98       124     269.0
10         60    103       147     329.3
11         60    100       120     250.7
12         60    106       128     345.3
13         60    104       132     379.3
14         60     98       123     275.0
15         60     98       120     215.2
16         60    100       120     300.0
17         45     90       112       NaN
18         60    103       123     323.0
19         45     97       125     243.0
20         60    108       131     364.2
21         45    100       119     282.0
22         60    130       101     300.0
23         45    105       132     246.0
24         60    102       126     334.5
25         60    100       120     250.0
26         60     92       118     241.0
27         60    103       132       NaN
28         60    100       132     280.0
```

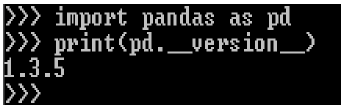
Reading CSV but print without converting to string



```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python Programmi
ng\\Lecture 24 Panda\\data.csv')
>>> print(df)
     Duration  Pulse  Maxpulse  Calories
0          60    110       130     409.1
1          60    117       145     479.0
2          60    103       135     340.0
3          45    109       175     282.4
4          45    117       148     406.0
..        ...    ...       ...       ...
164        60    105       140     290.8
165        60    110       145     300.0
166        60    115       145     310.2
167        75    120       150     320.4
168        75    125       150     330.4

[169 rows x 4 columns]
```

Checking the pandas version

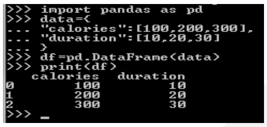


```
>>> import pandas as pd
>>> print(pd.__version__)
1.3.5
>>>
```

Pandas Data Frames

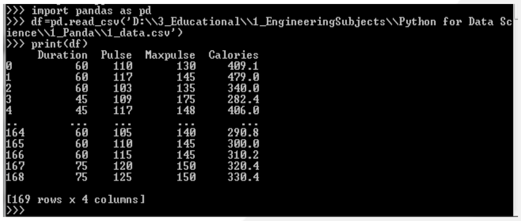
- A Pandas DataFrame is a 2 dimensional data structure, like a 2 dimensional array, or a table with rows and columns.

- Create a simple Panda Data Frame



```
>>> import pandas as pd
>>> data={
... "calories":[100,200,300],
... "duration":[10,20,30]
... }
>>> df=pd.DataFrame(data)
>>> print(df)
   calories  duration
0       100        10
1       200        20
2       300        30
>>> _
```

Load the CSV file into data Frame



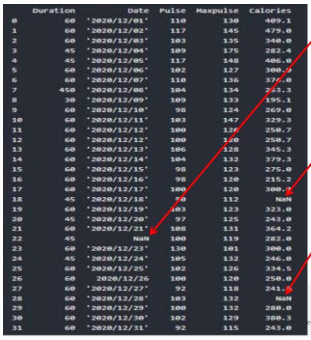
```
>>> import pandas as pd
>>> df=pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data Sc
ience\\1_Panda\\1_data.csv')
>>> print(df)
     Duration  Pulse  Maxpulse  Calories
0          60    110       130     409.1
1          60    117       145     479.0
2          60    103       135     340.0
3          45    109       175     282.4
4          45    117       148     406.0
..        ...    ...       ...       ...
164        60    105       140     290.8
165        60    110       145     300.0
166        60    115       145     310.2
167        75    120       150     320.4
168        75    125       150     330.4

[169 rows x 4 columns]
>>>
```

Data Cleaning

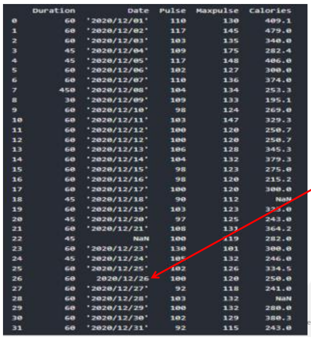
- Data cleaning means fixing bad data in your data set.

- Bad data could be:
  - Empty cells
  - Data in wrong format
  - Wrong data
  - Duplicates



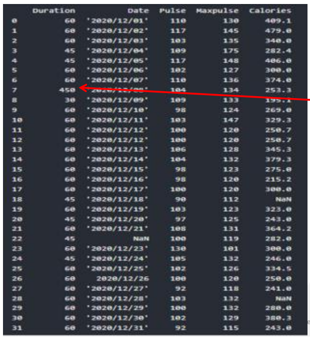
```
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
7        450  '2020/12/08'    104       134     253.3
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
12        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
25        60  '2020/12/25'    102       126     334.5
26        60    2020/12/26    100       120     250.0
27        60  '2020/12/27'     92       118     241.0
28        60  '2020/12/28'    103       132       NaN
29        60  '2020/12/29'    100       132     280.0
30        60  '2020/12/30'    102       129     380.3
31        60  '2020/12/31'     92       115     243.0
```

The data set contains some empty cells ("Date" in row 22, and "Calories" in row 18 and 28).



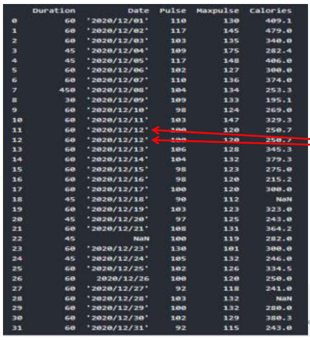
```
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
7        450  '2020/12/08'    104       134     253.3
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
12        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
25        60  '2020/12/25'    102       126     334.5
26        60    2020/12/26    100       120     250.0
27        60  '2020/12/27'     92       118     241.0
28        60  '2020/12/28'    103       132       NaN
29        60  '2020/12/29'    100       132     280.0
30        60  '2020/12/30'    102       129     380.3
31        60  '2020/12/31'     92       115     243.0
```

The data set contains wrong format ("Date" in row 26).



```
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
7        450  '2020/12/08'    104       134     253.3
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
12        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
25        60  '2020/12/25'    102       126     334.5
26        60    2020/12/26    100       120     250.0
27        60  '2020/12/27'     92       118     241.0
28        60  '2020/12/28'    103       132       NaN
29        60  '2020/12/29'    100       132     280.0
30        60  '2020/12/30'    102       129     380.3
31        60  '2020/12/31'     92       115     243.0
```

The data set contains wrong data ("Duration" in row 7).

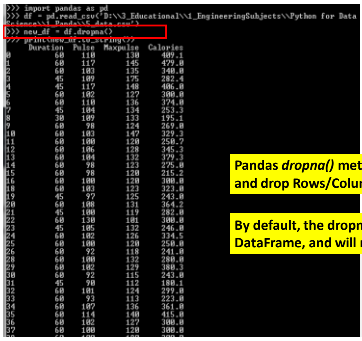


```
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
7        450  '2020/12/08'    104       134     253.3
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
12        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
25        60  '2020/12/25'    102       126     334.5
26        60    2020/12/26    100       120     250.0
27        60  '2020/12/27'     92       118     241.0
28        60  '2020/12/28'    103       132       NaN
29        60  '2020/12/29'    100       132     280.0
30        60  '2020/12/30'    102       129     380.3
31        60  '2020/12/31'     92       115     243.0
```

The data set contains duplicates (row 11 and 12).

1. Remove Rows

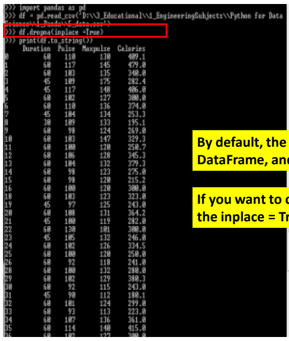
- One way to deal with empty cells is to remove rows that contain empty cells.
- This is usually OK, since data sets can be very big, and removing a few rows will not have a big impact on the result.
- See Row 17 and 27 (removed)



```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\5_data.csv')
>>> new_df = df.dropna()
>>> print(new_df.to_string())
    Duration  Pulse  Maxpulse  Calories
0         60    110       130     409.1
1         60    117       145     479.0
2         60    103       135     340.0
3         45    109       175     282.4
4         45    117       148     406.0
5         60    102       127     300.0
6         60    110       136     374.0
7         45    104       134     253.3
8         30    109       133     195.1
9         60     98       124     269.0
10        60    103       147     329.3
11        60    100       120     250.7
12        60    106       128     345.3
13        60    104       132     379.3
14        60     98       123     275.0
15        60     98       120     215.2
16        60    100       120     300.0
18        60    103       123     323.0
19        45     97       125     243.0
20        60    108       131     364.2
21        45    100       119     282.0
22        60    130       101     300.0
23        45    105       132     246.0
24        60    102       126     334.5
25        60    100       120     250.0
26        60     92       118     241.0
28        60    100       132     280.0
29        60    102       129     380.3
30        60     92       115     243.0
31        45     90       112     180.1
32        60    101       124     299.0
33        60     93       113     223.0
34        60    107       136     361.0
35        60    114       140     415.0
36        60    102       127     300.0
37        60    100       120     300.0
```

Pandas *dropna()* method allows the user to analyze and drop Rows/Columns with Null values

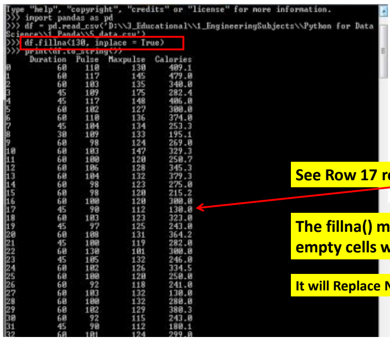
By default, the dropna() method returns a new DataFrame, and will not change the original.



```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\5_data.csv')
>>> df.dropna(inplace =True)
>>> print(df.to_string())
    Duration  Pulse  Maxpulse  Calories
0         60    110       130     409.1
1         60    117       145     479.0
2         60    103       135     340.0
3         45    109       175     282.4
4         45    117       148     406.0
5         60    102       127     300.0
6         60    110       136     374.0
7         45    104       134     253.3
8         30    109       133     195.1
9         60     98       124     269.0
10        60    103       147     329.3
11        60    100       120     250.7
12        60    106       128     345.3
13        60    104       132     379.3
14        60     98       123     275.0
15        60     98       120     215.2
16        60    100       120     300.0
18        60    103       123     323.0
19        45     97       125     243.0
20        60    108       131     364.2
21        45    100       119     282.0
22        60    130       101     300.0
23        45    105       132     246.0
24        60    102       126     334.5
25        60    100       120     250.0
26        60     92       118     241.0
28        60    100       132     280.0
29        60    102       129     380.3
30        60     92       115     243.0
31        45     90       112     180.1
32        60    101       124     299.0
33        60     93       113     223.0
34        60    107       136     361.0
35        60    114       140     415.0
```

By default, the dropna() method returns a new DataFrame, and will not change the original.

If you want to change the original DataFrame, use the inplace = True argument.



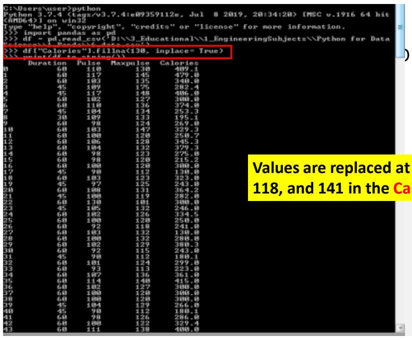
```
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\5_data.csv')
>>> df.fillna(130, inplace = True)
>>> print(df.to_string())
    Duration  Pulse  Maxpulse  Calories
0         60    110       130     409.1
1         60    117       145     479.0
2         60    103       135     340.0
3         45    109       175     282.4
4         45    117       148     406.0
5         60    102       127     300.0
6         60    110       136     374.0
7         45    104       134     253.3
8         30    109       133     195.1
9         60     98       124     269.0
10        60    103       147     329.3
11        60    100       120     250.7
12        60    106       128     345.3
13        60    104       132     379.3
14        60     98       123     275.0
15        60     98       120     215.2
16        60    100       120     300.0
17        45     90       112     130.0
18        60    103       123     323.0
19        45     97       125     243.0
20        60    108       131     364.2
21        45    100       119     282.0
22        60    130       101     300.0
23        45    105       132     246.0
24        60    102       126     334.5
25        60    100       120     250.0
26        60     92       118     241.0
27        60    103       132     130.0
28        60    100       132     280.0
29        60    102       129     380.3
30        60     92       115     243.0
31        45     90       112     180.1
32        60    101       124     299.0
```

See Row 17 replaced with 130

The fillna() method allows us to replace empty cells with a value.

It will Replace NULL values with the number 130.

olumn



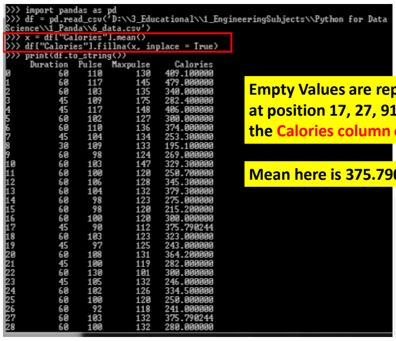
```
C:\Users\user>python
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 20:34:20) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\6_data.csv')
>>> df["Calories"].fillna(130, inplace= True)
>>> print(df.to_string())
    Duration  Pulse  Maxpulse  Calories
0         60    110       130     409.1
1         60    117       145     479.0
2         60    103       135     340.0
3         45    109       175     282.4
4         45    117       148     406.0
5         60    102       127     300.0
6         60    110       136     374.0
7         45    104       134     253.3
8         30    109       133     195.1
9         60     98       124     269.0
10        60    103       147     329.3
11        60    100       120     250.7
12        60    106       128     345.3
13        60    104       132     379.3
14        60     98       123     275.0
15        60     98       120     215.2
16        60    100       120     300.0
17        45     90       112     130.0
18        60    103       123     323.0
19        45     97       125     243.0
20        60    108       131     364.2
21        45    100       119     282.0
22        60    130       101     300.0
23        45    105       132     246.0
24        60    102       126     334.5
25        60    100       120     250.0
26        60     92       118     241.0
27        60    103       132     130.0
28        60    100       132     280.0
29        60    102       129     380.3
30        60     92       115     243.0
31        45     90       112     180.1
32        60    101       124     299.0
33        60     93       113     223.0
34        60    107       136     361.0
35        60    114       140     415.0
36        60    102       127     300.0
37        60    100       120     300.0
38        60    100       120     300.0
39        45    104       129     266.0
40        45     90       112     180.1
41        60     98       126     286.0
42        60    100       122     329.4
43        60    111       138     400.0
```

Values are replaced at position 17, 27, 91, 118, and 141 in the Calories column only.

5. Replace Using Mean, Median, or Mode

- A common way to replace empty cells, is to calculate the mean, median or mode value of the column.

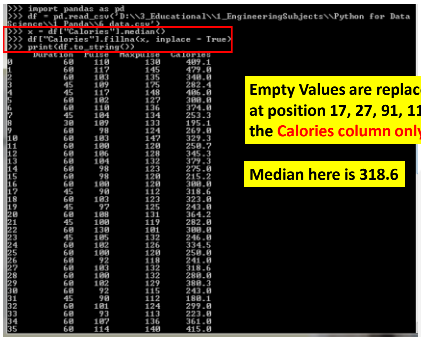
- Mean → Average
- Median → Center value
- Mode → Most common occurring value



```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\6_data.csv')
>>> x = df["Calories"].mean()
>>> df["Calories"].fillna(x, inplace = True)
>>> print(df.to_string())
    Duration  Pulse  Maxpulse    Calories
0         60    110       130  409.100000
1         60    117       145  479.000000
2         60    103       135  340.000000
3         45    109       175  282.400000
4         45    117       148  406.000000
5         60    102       127  300.000000
6         60    110       136  374.000000
7         45    104       134  253.300000
8         30    109       133  195.100000
9         60     98       124  269.000000
10        60    103       147  329.300000
11        60    100       120  250.700000
12        60    106       128  345.300000
13        60    104       132  379.300000
14        60     98       123  275.000000
15        60     98       120  215.200000
16        60    100       120  300.000000
17        45     90       112  375.790244
18        60    103       123  323.000000
19        45     97       125  243.000000
20        60    108       131  364.200000
21        45    100       119  282.000000
22        60    130       101  300.000000
23        45    105       132  246.000000
24        60    102       126  334.500000
25        60    100       120  250.000000
26        60     92       118  241.000000
27        60    103       132  375.790244
28        60    100       132  280.000000
```

Empty Values are replaced with mean at position 17, 27, 91, 118, and 141 in the Calories column only.

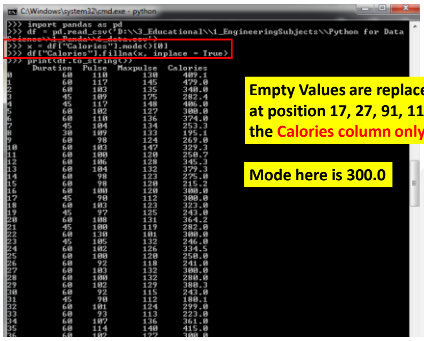
Mean here is 375.790244



```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\6_data.csv')
>>> x = df["Calories"].median()
>>> df["Calories"].fillna(x, inplace = True)
>>> print(df.to_string())
    Duration  Pulse  Maxpulse  Calories
0         60    110       130     409.1
1         60    117       145     479.0
2         60    103       135     340.0
3         45    109       175     282.4
4         45    117       148     406.0
5         60    102       127     300.0
6         60    110       136     374.0
7         45    104       134     253.3
8         30    109       133     195.1
9         60     98       124     269.0
10        60    103       147     329.3
11        60    100       120     250.7
12        60    106       128     345.3
13        60    104       132     379.3
14        60     98       123     275.0
15        60     98       120     215.2
16        60    100       120     300.0
17        45     90       112     318.6
18        60    103       123     323.0
19        45     97       125     243.0
20        60    108       131     364.2
21        45    100       119     282.0
22        60    130       101     300.0
23        45    105       132     246.0
24        60    102       126     334.5
25        60    100       120     250.0
26        60     92       118     241.0
27        60    103       132     318.6
28        60    100       132     280.0
29        60    102       129     380.3
30        60     92       115     243.0
31        45     90       112     180.1
32        60    101       124     299.0
33        60     93       113     223.0
34        60    107       136     361.0
35        60    114       140     415.0
```

Empty Values are replaced with median at position 17, 27, 91, 118, and 141 in the Calories column only.

Median here is 318.6



C:\Windows\system32\cmd.exe - python

```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\6_data.csv')
>>> x = df["Calories"].mode()[0]
>>> df["Calories"].fillna(x, inplace = True)
>>> print(df.to_string())
    Duration  Pulse  Maxpulse  Calories
0         60    110       130     409.1
1         60    117       145     479.0
2         60    103       135     340.0
3         45    109       175     282.4
4         45    117       148     406.0
5         60    102       127     300.0
6         60    110       136     374.0
7         45    104       134     253.3
8         30    109       133     195.1
9         60     98       124     269.0
10        60    103       147     329.3
11        60    100       120     250.7
12        60    106       128     345.3
13        60    104       132     379.3
14        60     98       123     275.0
15        60     98       120     215.2
16        60    100       120     300.0
17        45     90       112     300.0
18        60    103       123     323.0
19        45     97       125     243.0
20        60    108       131     364.2
21        45    100       119     282.0
22        60    130       101     300.0
23        45    105       132     246.0
24        60    102       126     334.5
25        60    100       120     250.0
26        60     92       118     241.0
27        60    103       132     300.0
28        60    100       132     280.0
29        60    102       129     380.3
30        60     92       115     243.0
31        45     90       112     180.1
32        60    101       124     299.0
33        60     93       113     223.0
34        60    107       136     361.0
35        60    114       140     415.0
36        60    102       127     300.0
```

Empty Values are replaced with mode at position 17, 27, 91, 118, and 141 in the Calories column only.

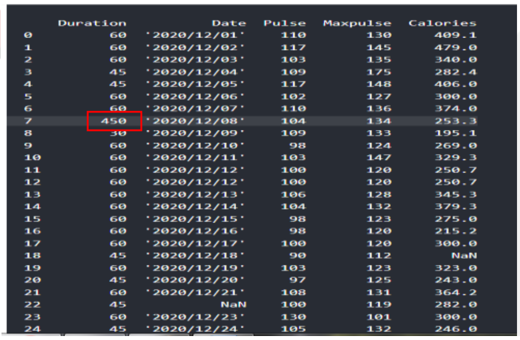
Mode here is 300.0

Wrong Data

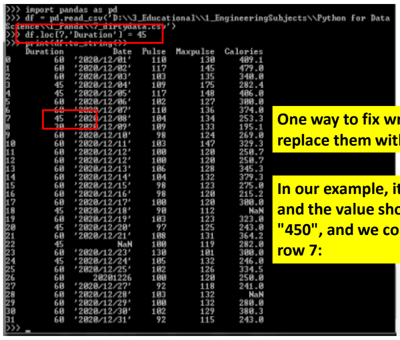
- "Wrong data" does not have to be "empty cells" or "wrong format", it can just be wrong, like if someone registered "199" instead of "1.99".

- Sometimes you can spot wrong data by looking at the data set, because you have an expectation of what it should be.

- If you take a look at our data set, you can see that in row 7, the duration is 450, but for all the other rows the duration is between 30 and 60.



```
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
7        450  '2020/12/08'    104       134     253.3
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
12        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
```



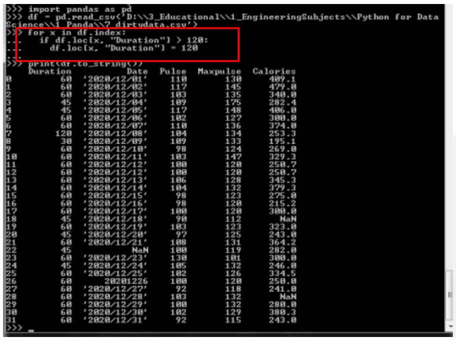
```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\7_dirtydata.csv')
>>> df.loc[7,'Duration'] = 45
>>> print(df.to_string())
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
7         45  '2020/12/08'    104       134     253.3
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
12        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
25        60  '2020/12/25'    102       126     334.5
26        60      20201226    100       120     250.0
27        60  '2020/12/27'     92       118     241.0
28        60  '2020/12/28'    103       132       NaN
29        60  '2020/12/29'    100       132     280.0
30        60  '2020/12/30'    102       129     380.3
31        60  '2020/12/31'     92       115     243.0
>>> _
```

One way to fix wrong values is to replace them with something else.

In our example, it is most likely a typo, and the value should be "45" instead of "450", and we could just insert "45" in row 7:

For Larger Data

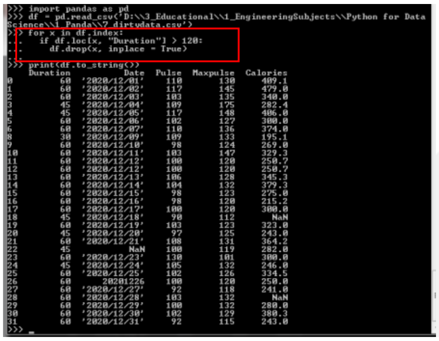
- For small data sets you might be able to replace the wrong data one by one, but not for big data sets.
- To replace wrong data for larger data sets you can create some rules, e.g. set some boundaries for legal values, and replace any values that are outside of the boundaries.



```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\7_dirtydata.csv')
>>> for x in df.index:
...   if df.loc[x, "Duration"] > 120:
...     df.loc[x, "Duration"] = 120
...
>>> print(df.to_string())
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
7        120  '2020/12/08'    104       134     253.3
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
12        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
25        60  '2020/12/25'    102       126     334.5
26        60      20201226    100       120     250.0
27        60  '2020/12/27'     92       118     241.0
28        60  '2020/12/28'    103       132       NaN
29        60  '2020/12/29'    100       132     280.0
30        60  '2020/12/30'    102       129     380.3
31        60  '2020/12/31'     92       115     243.0
>>> _
```

Removing Rows

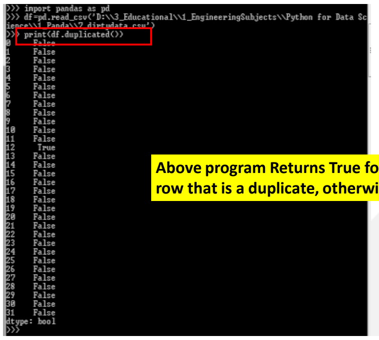
- Another way of handling wrong data is to remove the rows that contains wrong data.
- This way you do not have to find out what to replace them with, and there is a good chance you do not need them to do your analyses.
- Value at position no 7 is removed



```
>>> import pandas as pd
>>> df = pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data
Science\\1_Panda\\7_dirtydata.csv')
>>> for x in df.index:
...   if df.loc[x, "Duration"] > 120:
...     df.drop(x, inplace = True)
...
>>> print(df.to_string())
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
12        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
25        60  '2020/12/25'    102       126     334.5
26        60      20201226    100       120     250.0
27        60  '2020/12/27'     92       118     241.0
28        60  '2020/12/28'    103       132       NaN
29        60  '2020/12/29'    100       132     280.0
30        60  '2020/12/30'    102       129     380.3
31        60  '2020/12/31'     92       115     243.0
>>> _
```

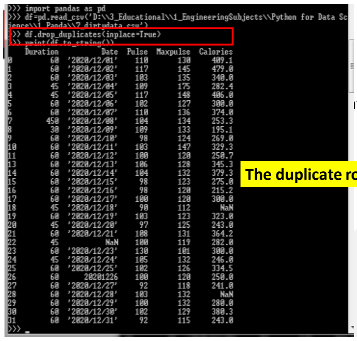
Duplicate Data

- Duplicate rows are rows that have been registered more than one time.
- By taking a look at our test data set, we can assume that row 11 and 12 are duplicates.
- To discover duplicates, we can use the duplicated() method.
- The duplicated() method returns a Boolean values for each row.



```
>>> import pandas as pd
>>> df=pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data Sc
ience\\1_Panda\\7_dirtydata.csv')
>>> print(df.duplicated())
0      False
1      False
2      False
3      False
4      False
5      False
6      False
7      False
8      False
9      False
10     False
11     False
12      True
13     False
14     False
15     False
16     False
17     False
18     False
19     False
20     False
21     False
22     False
23     False
24     False
25     False
26     False
27     False
28     False
29     False
30     False
31     False
dtype: bool
>>>
```

Above program Returns True for every row that is a duplicate, otherwise False



```
>>> import pandas as pd
>>> df=pd.read_csv('D:\\3_Educational\\1_EngineeringSubjects\\Python for Data Sc
ience\\1_Panda\\7_dirtydata.csv')
>>> df.drop_duplicates(inplace=True)
>>> print(df.to_string())
    Duration          Date  Pulse  Maxpulse  Calories
0         60  '2020/12/01'    110       130     409.1
1         60  '2020/12/02'    117       145     479.0
2         60  '2020/12/03'    103       135     340.0
3         45  '2020/12/04'    109       175     282.4
4         45  '2020/12/05'    117       148     406.0
5         60  '2020/12/06'    102       127     300.0
6         60  '2020/12/07'    110       136     374.0
7        450  '2020/12/08'    104       134     253.3
8         30  '2020/12/09'    109       133     195.1
9         60  '2020/12/10'     98       124     269.0
10        60  '2020/12/11'    103       147     329.3
11        60  '2020/12/12'    100       120     250.7
13        60  '2020/12/13'    106       128     345.3
14        60  '2020/12/14'    104       132     379.3
15        60  '2020/12/15'     98       123     275.0
16        60  '2020/12/16'     98       120     215.2
17        60  '2020/12/17'    100       120     300.0
18        45  '2020/12/18'     90       112       NaN
19        60  '2020/12/19'    103       123     323.0
20        45  '2020/12/20'     97       125     243.0
21        60  '2020/12/21'    108       131     364.2
22        45           NaN    100       119     282.0
23        60  '2020/12/23'    130       101     300.0
24        45  '2020/12/24'    105       132     246.0
25        60  '2020/12/25'    102       126     334.5
26        60      20201226    100       120     250.0
27        60  '2020/12/27'     92       118     241.0
28        60  '2020/12/28'    103       132       NaN
29        60  '2020/12/29'    100       132     280.0
30        60  '2020/12/30'    102       129     380.3
31        60  '2020/12/31'     92       115     243.0
>>> _
```

tes() method.

The duplicate row (row no 12) is now removed

Summary

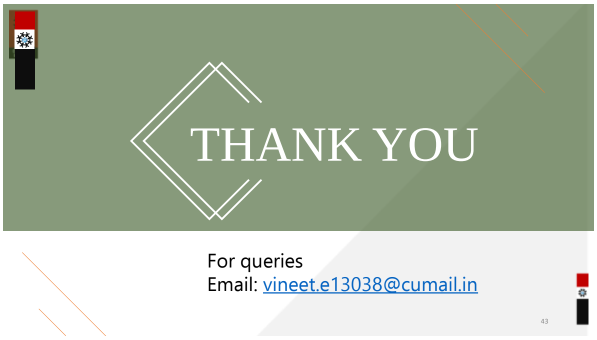
- Methods of Sourcing Data
- Methods of Cleaning Data

Task

- Applying various methods that are used for sourcing the data by taking a suitable arrays\datasets etc. (BT-Level3)
- Design a model that is used to clean Empty cells, Data in wrong format, Wrong data, and Duplicates. (BT-Level6)

REFERENCES

- https://www.javatpoint.com/machine-learning
- https://www.tutorialspoint.com/machine_learning/index.htm
- https://www.w3schools.com/python/



THANK YOU

For queries
Email: vineet.e13038@cumail.in

43