

Start coding or [generate](#) with AI.

By - Devanshu

**Description:** This task involves using the Pandas library to manipulate data.

**Responsibility:** Load a CSV file into a Pandas DataFrame. Perform operations like filtering data based on conditions, handling missing values, and calculating summary statistics.

```
import pandas as pd
```

```
data = pd.read_csv("//01.Data Cleaning and Preprocessing.csv")
```

data



	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	T-upperExt-2	T-lowerExt-2	UCZAA	WhiteFlow-4	...	SteamFlow-4	Lower-HeatT-3
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.545	1.443	599.253	...	67.122	329.432
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.067	1.549	537.201	...	60.012	330.823
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.260	1.600	549.611	...	61.304	329.140
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.142	1.604	623.362	...	68.496	328.875
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.709	NaN	638.672	...	70.022	328.352
...	...	...	...	...	...	...	...	...	...	...	...	...	...
319	10-16:00	23.75	12.667	93.450	1178.252	276.955	347.286	310.970	1.523	513.956	...	61.141	330.117
320	9-19:00	19.80	12.558	94.352	1184.119	297.071	399.135	319.576	1.451	570.058	...	67.667	330.848
321	9-20:00	23.01	12.550	90.842	1188.517	289.826	373.633	314.591	1.457	549.306	...	66.446	330.226
322	9-21:00	24.32	13.083	88.910	1192.879	318.006	364.081	308.559	1.523	504.852	...	61.054	327.346
323	9-22:00	25.75	13.417	85.451	1186.342	248.312	356.289	310.482	1.474	497.375	...	58.247	328.092

324 rows × 23 columns

type(data) #typr of data



```
pandas.core.frame.DataFrame
def __init__(data=None, index: Axes | None=None, columns: Axes | None=None,
dtype: Dtype | None=None, copy: bool | None=None) -> None
```

</usr/local/lib/python3.10/dist-packages/pandas/core/frame.py>  
Two-dimensional, size-mutable, potentially heterogeneous tabular data.

Data structure also contains labeled axes (rows and columns).  
Arithmetic operations align on both row and column labels. Can be

data.info() #prints data information



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 324 entries, 0 to 323
Data columns (total 23 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Observation            324 non-null    object
1   Y-Kappa                324 non-null    float64
2   ChipRate              319 non-null    float64
3   BF-CMratio            307 non-null    float64
4   BlowFlow              308 non-null    float64
5   ChipLevel4            323 non-null    float64
6   T-upperExt-2          322 non-null    float64
7   T-lowerExt-2          322 non-null    float64
8   UCZAA                 299 non-null    float64
9   WhiteFlow-4           323 non-null    float64
10  AAWhiteSt-4           173 non-null    float64
11  AA-Wood-4             323 non-null    float64
12  ChipMoisture-4        323 non-null    float64
13  SteamFlow-4           323 non-null    float64
14  Lower-HeatT-3         322 non-null    float64
15  Upper-HeatT-3         322 non-null    float64
```

```
16 ChipMass-4      323 non-null    float64
17 WeakLiquorF     323 non-null    float64
18 BlackFlow-2     322 non-null    float64
19 WeakWashF       323 non-null    float64
20 SteamHeatF-3    322 non-null    float64
21 T-Top-Chips-4   323 non-null    float64
22 SulphidityL-4   173 non-null    float64
dtypes: float64(22), object(1)
memory usage: 58.3+ KB
```

```
data.describe() #describe statistical
```

	Y-Kappa	ChipRate	BF- CMratio	BlowFlow	ChipLevel14	T- upperExt- 2	lowerExt
count	324.000000	319.000000	307.000000	308.000000	323.000000	322.000000	322.000000
mean	20.635370	14.347937	87.464456	1237.837614	258.164483	356.904295	324.020000
std	3.070036	1.499095	7.995012	100.593735	87.987452	9.209290	7.621400
min	12.170000	9.983000	68.645000	0.000000	0.000000	339.168000	284.633000
25%	18.382500	13.358000	81.823000	1193.215250	213.527000	350.241250	321.420000
50%	20.845000	14.308000	86.739000	1273.138500	271.792000	356.843000	325.669000
75%	23.032500	15.517000	92.372000	1289.196000	321.680000	362.242250	329.175000
max	27.600000	16.958000	121.717000	1351.240000	419.014000	399.135000	337.012000

8 rows × 22 columns

```
data = data.drop_duplicates() #deletes all the duplicates
data
```

	Observation	Y- Kappa	ChipRate	BF- CMratio	BlowFlow	ChipLevel14	T- upperExt- 2	T lowerExt ;
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.540
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.060
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.260
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.140
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.700
...	...	...	...	...	...	...	...	...
298	12-09:00	20.90	15.167	84.640	1283.706	339.440	354.803	311.040
299	12-10:00	24.98	NaN	85.034	1278.345	368.564	357.723	321.380
300	12-11:00	21.00	NaN	88.013	1307.722	278.842	357.438	323.750
301	12-12:00	21.40	NaN	85.490	1255.986	273.484	361.365	322.680
307	31-05:00	20.89	14.308	94.172	1327.832	251.120	351.263	332.480

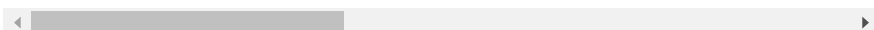
301 rows × 23 columns

```
data.isnull() #return true for null, false for notnull
```



	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel14	T-upperExt-2	T-lowerExt-2
0	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...
298	False	False	False	False	False	False	False	False
299	False	False	True	False	False	False	False	False
300	False	False	True	False	False	False	False	False
301	False	False	True	False	False	False	False	False
307	False	False	False	False	False	False	False	False

301 rows × 23 columns



```
data.isnull().sum() #provides total null values in a row
```



```

Observation      0
Y-Kappa          0
ChipRate         4
BF-CMratio      14
BlowFlow        13
ChipLevel14       1
T-upperExt-2      1
T-lowerExt-2      1
UCZAA           24
WhiteFlow-4       1
AAWhiteSt-4     141
AA-Wood-4         1
ChipMoisture-4    1
SteamFlow-4       1
Lower-HeatT-3     1
Upper-HeatT-3     1
ChipMass-4        1
WeakLiquorF       1
BlackFlow-2       1
WeakWashF         1
SteamHeatF-3      1
T-Top-Chips-4     1
SulphidityL-4    141
dtype: int64

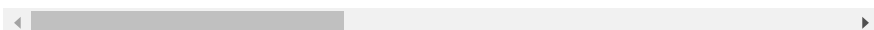
```

```
data.notnull() # true for not null, false for null
```



	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel14	T-upperExt-2	T-lowerExt-2
0	True	True	True	True	True	True	True	True
1	True	True	True	True	True	True	True	True
2	True	True	True	True	True	True	True	True
3	True	True	True	True	True	True	True	True
4	True	True	True	True	True	True	True	True
...	...	...	...	...	...	...	...	...
298	True	True	True	True	True	True	True	True
299	True	True	False	True	True	True	True	True
300	True	True	False	True	True	True	True	True
301	True	True	False	True	True	True	True	True
307	True	True	True	True	True	True	True	True

301 rows × 23 columns



```
data.isnull().sum().sum() # provides total number of null
```

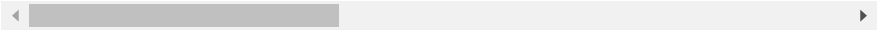
 352

```
data2 = data.fillna(value=0) # fill all the nulls to 0
data2
```



	Observation	Y- Kappa	ChipRate	BF- CMratio	BlowFlow	ChipLevel14	T- upperExt- 2	T lowerExt :
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.544
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.067
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.261
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.144
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.701
...	...	...	...	...	...	...	...	...
298	12-09:00	20.90	15.167	84.640	1283.706	339.440	354.803	311.044
299	12-10:00	24.98	0.000	85.034	1278.345	368.564	357.723	321.387
300	12-11:00	21.00	0.000	88.013	1307.722	278.842	357.438	323.755
301	12-12:00	21.40	0.000	85.490	1255.986	273.484	361.365	322.681
307	31-05:00	20.89	14.308	94.172	1327.832	251.120	351.263	332.481

301 rows × 23 columns

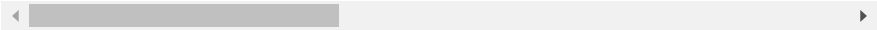


```
data3 = data.fillna(method="pad") #forward filling
data3
```



	Observation	Y- Kappa	ChipRate	BF- CMratio	BlowFlow	ChipLevel14	T- upperExt- 2	T lowerExt :
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.544
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.067
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.261
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.144
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.701
...	...	...	...	...	...	...	...	...
298	12-09:00	20.90	15.167	84.640	1283.706	339.440	354.803	311.044
299	12-10:00	24.98	15.167	85.034	1278.345	368.564	357.723	321.387
300	12-11:00	21.00	15.167	88.013	1307.722	278.842	357.438	323.755
301	12-12:00	21.40	15.167	85.490	1255.986	273.484	361.365	322.681
307	31-05:00	20.89	14.308	94.172	1327.832	251.120	351.263	332.481

301 rows × 23 columns

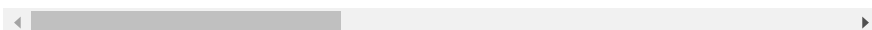


```
data4 = data.fillna(method="bfill") #backward filling
data4
```



	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	T-upperExt-2	T-lowerExt-2
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.541
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.061
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.261
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.141
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.701
...	...	...	...	...	...	...	...	...
298	12-09:00	20.90	15.167	84.640	1283.706	339.440	354.803	311.041
299	12-10:00	24.98	14.308	85.034	1278.345	368.564	357.723	321.381
300	12-11:00	21.00	14.308	88.013	1307.722	278.842	357.438	323.751
301	12-12:00	21.40	14.308	85.490	1255.986	273.484	361.365	322.681
307	31-05:00	20.89	14.308	94.172	1327.832	251.120	351.263	332.481

301 rows × 23 columns



```
import numpy as np
from scipy import stats
```

```
data2.columns #detects the outlier using IQR
```



```
Index(['Observation', 'Y-Kappa', 'ChipRate', 'BF-CMratio', 'BlowFlow',
       'ChipLevel4 ', 'T-upperExt-2 ', 'T-lowerExt-2 ', 'UCZAA',
       'WhiteFlow-4 ', 'AAWhiteSt-4 ', 'AA-Wood-4 ', 'ChipMoisture-4 ',
       'SteamFlow-4 ', 'Lower-HeatT-3', 'Upper-HeatT-3 ', 'ChipMass-4 ',
       'WeakLiquorF ', 'BlackFlow-2 ', 'WeakWashF ', 'SteamHeatF-3 ',
       'T-Top-Chips-4 ', 'SulphidityL-4 '],
      dtype='object')
```

```
data2.drop(["Observation"],axis=1,inplace=True) #dropping unwanted column
data2.columns
```



```
Index(['Y-Kappa', 'ChipRate', 'BF-CMratio', 'BlowFlow', 'ChipLevel4 ',
       'T-upperExt-2 ', 'T-lowerExt-2 ', 'UCZAA', 'WhiteFlow-4 ',
       'AAWhiteSt-4 ', 'AA-Wood-4 ', 'ChipMoisture-4 ', 'SteamFlow-4 ',
       'Lower-HeatT-3', 'Upper-HeatT-3 ', 'ChipMass-4 ', 'WeakLiquorF ',
       'BlackFlow-2 ', 'WeakWashF ', 'SteamHeatF-3 ', 'T-Top-Chips-4 ',
       'SulphidityL-4 '],
      dtype='object')
```

```
Q1=data2.quantile(0.25)
Q3=data2.quantile(0.75)
IQR = Q3-Q1 #assigning value to IQR
print(IQR)
```



```
Y-Kappa          4.550
ChipRate         2.233
BF-CMratio       10.912
BlowFlow        96.766
ChipLevel4      105.868
T-upperExt-2     11.994
T-lowerExt-2     7.609
UCZAA            0.152
WhiteFlow-4     100.098
AAWhiteSt-4      6.143
AA-Wood-4        1.486
ChipMoisture-4   2.186
SteamFlow-4      8.840
Lower-HeatT-3    8.585
Upper-HeatT-3    7.852
ChipMass-4       19.347
WeakLiquorF     180.613
BlackFlow-2     280.829
WeakWashF       267.219
SteamHeatF-3     6.903
T-Top-Chips-4    2.044
SulphidityL-4    30.420
dtype: float64
```

```
data2 = data2[~((data2 < (Q1 - 1.5*IQR)) |(data2 > (Q3 + 1.5*IQR))).any(axis=1)] #formula for IQL
data2
```



	Y- Kappa	ChipRate	BF- CMratio	BlowFlow	ChipLevel4	T- upperExt- 2	T- lowerExt- 2	UCZAA	Whit
1	27.60	16.810	79.022	1328.360	341.327	351.050	329.067	1.549	