

PYTHON DEVELOPER

TASK-2

Simple Data Manipulation with Pandas

Description:

The intern will explore basic data analysis using the Pandas library in Python.

Responsibility:

1. Learn how to read CSV files and manipulate data frames using Pandas.
2. Perform simple data cleaning tasks such as handling missing values and removing duplicates.
3. Practice basic data manipulation operations like filtering, sorting, and grouping data.

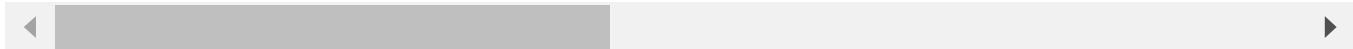
In [1]: `import pandas as pd`

```
# Read a CSV file into a DataFrame
df = pd.read_csv(r"C:\Users\saket\Downloads\01.Data Cleaning and Preprocessing.csv")
df
```

Out[1]:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	upperExt-2	T-lowerExt-2	T-UCZAF
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.545	1.44
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.067	1.54
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.260	1.60
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.142	1.60
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.709	NaN
...
319	10-16:00	23.75	12.667	93.450	1178.252	276.955	347.286	310.970	1.52
320	9-19:00	19.80	12.558	94.352	1184.119	297.071	399.135	319.576	1.45
321	9-20:00	23.01	12.550	90.842	1188.517	289.826	373.633	314.591	1.45
322	9-21:00	24.32	13.083	88.910	1192.879	318.006	364.081	308.559	1.52
323	9-22:00	25.75	13.417	85.451	1186.342	248.312	356.289	310.482	1.47

324 rows × 23 columns



In [2]:

```
# Display the first 5 rows of the DataFrame
print("First 5 rows of the DataFrame:")
print(df.head())

# Display the last 5 rows of the DataFrame
print("\nLast 5 rows of the DataFrame:")
print(df.tail())

# Display the number of rows and columns in the DataFrame
print("\nNumber of rows and columns in the DataFrame:", df.shape)

# Display the data types of the columns in the DataFrame
print("\nData types of the columns in the DataFrame:")
print(df.dtypes)
```

First 5 rows of the DataFrame:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	\
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	
	T-upperExt-2	T-lowerExt-2	UCZAA	WhiteFlow-4	...	SteamFlow-4	\
0	358.282	329.545	1.443	599.253	...	67.122	
1	351.050	329.067	1.549	537.201	...	60.012	
2	350.022	329.260	1.600	549.611	...	61.304	
3	350.938	331.142	1.604	623.362	...	68.496	
4	351.640	332.709	NaN	638.672	...	70.022	
	Lower-HeatT-3	Upper-HeatT-3	ChipMass-4	WeakLiquorF	BlackFlow-2	\	
0	329.432	303.099	175.964	1127.197	1319.039		
1	330.823	304.879	163.202	665.975	1297.317		
2	329.140	303.383	164.013	677.534	1327.072		
3	328.875	302.254	181.487	767.853	1324.461		
4	328.352	300.954	183.929	888.448	1343.424		
	WeakWashF	SteamHeatF-3	T-Top-Chips-4	SulphidityL-4			
0	257.325	54.612	252.077	NaN			
1	241.182	46.603	251.406	29.11			
2	237.272	51.795	251.335	NaN			
3	239.478	54.846	250.312	29.02			
4	215.372	54.186	249.916	29.01			

[5 rows x 23 columns]

Last 5 rows of the DataFrame:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	\
319	10-16:00	23.75	12.667	93.450	1178.252	276.955	
320	9-19:00	19.80	12.558	94.352	1184.119	297.071	
321	9-20:00	23.01	12.550	90.842	1188.517	289.826	
322	9-21:00	24.32	13.083	88.910	1192.879	318.006	
323	9-22:00	25.75	13.417	85.451	1186.342	248.312	
	T-upperExt-2	T-lowerExt-2	UCZAA	WhiteFlow-4	...	SteamFlow-4	\
319	347.286	310.970	1.523	513.956	...	61.141	
320	399.135	319.576	1.451	570.058	...	67.667	
321	373.633	314.591	1.457	549.306	...	66.446	
322	364.081	308.559	1.523	504.852	...	61.054	
323	356.289	310.482	1.474	497.375	...	58.247	
	Lower-HeatT-3	Upper-HeatT-3	ChipMass-4	WeakLiquorF	BlackFlow-2	\	
319	330.117	304.006	148.174	1027.201	1357.271		
320	330.848	304.616	165.178	906.962	1311.177		
321	330.226	304.686	160.841	887.125	1319.226		
322	327.346	304.363	147.589	804.423	1320.225		
323	328.092	304.093	144.218	828.328	1320.848		
	WeakWashF	SteamHeatF-3	T-Top-Chips-4	SulphidityL-4			
319	381.643	45.264	252.947	30.86			
320	25.494	50.528	252.092	30.70			
321	0.638	45.549	252.438	NaN			
322	0.000	43.725	253.176	31.13			
323	1.276	43.840	253.216	NaN			

[5 rows x 23 columns]

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js taFrame: (324, 23)

```
Data types of the columns in the DataFrame:  
Observation          object  
Y-Kappa              float64  
ChipRate             float64  
BF-CMratio           float64  
BlowFlow              float64  
ChipLevel4            float64  
T-upperExt-2          float64  
T-lowerExt-2          float64  
UCZAA                float64  
WhiteFlow-4            float64  
AAWhiteSt-4            float64  
AA-Wood-4              float64  
ChipMoisture-4         float64  
SteamFlow-4             float64  
Lower-HeatT-3           float64  
Upper-HeatT-3           float64  
ChipMass-4              float64  
WeakLiquorF             float64  
BlackFlow-2              float64  
WeakWashF               float64  
SteamHeatF-3              float64  
T-Top-Chips-4            float64  
SulphidityL-4             float64  
dtype: object
```

```
In [3]: print("\nStatistical summary of the DataFrame:")  
print(df.describe())
```

Statistical summary of the DataFrame:

	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	\
count	324.000000	319.000000	307.000000	308.000000	323.000000	
mean	20.635370	14.347937	87.464456	1237.837614	258.164483	
std	3.070036	1.499095	7.995012	100.593735	87.987452	
min	12.170000	9.983000	68.645000	0.000000	0.000000	
25%	18.382500	13.358000	81.823000	1193.215250	213.527000	
50%	20.845000	14.308000	86.739000	1273.138500	271.792000	
75%	23.032500	15.517000	92.372000	1289.196000	321.680000	
max	27.600000	16.958000	121.717000	1351.240000	419.014000	
	T-upperExt-2	T-lowerExt-2	UCZAA	WhiteFlow-4	AAWhiteSt-4	\
count	322.000000	322.000000	299.000000	323.000000	173.000000	
mean	356.904295	324.020180	1.492010	591.732260	6.140410	
std	9.209290	7.621402	0.105923	67.016351	0.081609	
min	339.168000	284.633000	1.182000	405.111000	5.890000	
25%	350.241250	321.420000	1.431500	540.989500	6.089000	
50%	356.843000	325.669000	1.498000	592.895000	6.135000	
75%	362.242250	329.175000	1.560500	639.480500	6.199000	
max	399.135000	337.012000	1.747000	731.394000	6.340000	
	...	SteamFlow-4	Lower-HeatT-3	Upper-HeatT-3	ChipMass-4	\
count	...	323.000000	322.000000	322.000000	323.000000	
mean	...	66.668285	325.567820	300.525699	162.222322	
std	...	5.708587	4.609862	4.568484	14.160688	
min	...	48.568000	318.051000	293.312000	113.922000	
25%	...	62.518000	321.385500	296.513250	153.032500	
50%	...	67.429000	324.741000	299.126000	163.690000	
75%	...	71.522000	329.845250	304.244750	172.555000	
max	...	76.147000	333.854000	311.146000	189.268000	
	WeakLiquorF	BlackFlow-2	WeakWashF	SteamHeatF-3	T-Top-Chips-4	\
count	323.000000	322.000000	323.000000	322.000000	323.000000	
mean	873.828941	1175.917016	263.543068	49.696907	251.240087	
std	122.073521	149.334010	163.666942	4.551909	1.283432	
min	486.938000	838.948000	0.000000	35.510000	248.359000	
25%	792.019500	1044.817500	134.649000	46.389750	250.312000	
50%	865.254000	1150.221500	269.193000	50.277000	251.380000	
75%	965.286500	1319.021250	405.563000	53.294250	252.323500	
max	1226.277000	1395.767000	715.715000	63.332000	254.122000	
	SulphidityL-4					
count	173.000000					
mean	30.411671					
std	0.701317					
min	29.010000					
25%	29.970000					
50%	30.370000					
75%	30.820000					
max	32.840000					

[8 rows x 22 columns]

```
In [4]: df = df.drop_duplicates()
df
```

Out[4]:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	upperExt-2	T-lowerExt-2	T-UCZAF
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.545	1.44:
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.067	1.549
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.260	1.600
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.142	1.604
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.709	NaN
...
298	12-09:00	20.90	15.167	84.640	1283.706	339.440	354.803	311.041	1.63!
299	12-10:00	24.98	NaN	85.034	1278.345	368.564	357.723	321.387	NaN
300	12-11:00	21.00	NaN	88.013	1307.722	278.842	357.438	323.757	NaN
301	12-12:00	21.40	NaN	85.490	1255.986	273.484	361.365	322.689	NaN
307	31-05:00	20.89	14.308	94.172	1327.832	251.120	351.263	332.485	1.52:

301 rows × 23 columns

In [5]: df.isnull()

Out[5]:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	upperExt-2	T-lowerExt-2	T-UCZAF
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	True
...
298	False	False	False	False	False	False	False	False	False
299	False	False	True	False	False	False	False	False	True
300	False	False	True	False	False	False	False	False	True
301	False	False	True	False	False	False	False	False	True
307	False	False	False	False	False	False	False	False	False

301 rows × 23 columns

In [6]: df.isnull().sum()

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

```
Out[6]: Observation      0
Y-Kappa          0
ChipRate         4
BF-CMratio      14
BlowFlow        13
ChipLevel4      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
```

In [7]: `df.notnull()`

Out[7]:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	T-upperExt-2	T-lowerExt-2	UCZAA
0	True	True	True	True	True	True	True	True	True
1	True	True	True	True	True	True	True	True	True
2	True	True	True	True	True	True	True	True	True
3	True	True	True	True	True	True	True	True	True
4	True	True	True	True	True	True	True	True	False
...
298	True	True	True	True	True	True	True	True	True
299	True	True	False	True	True	True	True	True	False
300	True	True	False	True	True	True	True	True	False
301	True	True	False	True	True	True	True	True	False
307	True	True	True	True	True	True	True	True	True

301 rows × 23 columns

In [8]: `df.isnull().sum().sum()`

Out[8]: 352

In [10]: `# Fill missing values with a specific value
df1=df.fillna(0) # fill missing values with 0`

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

Out[10]:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	T-upperExt-2	T-lowerExt-2	UCZAI
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.545	1.44:
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.067	1.549
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.260	1.600
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.142	1.604
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.709	0.000
...
298	12-09:00	20.90	15.167	84.640	1283.706	339.440	354.803	311.041	1.63!
299	12-10:00	24.98	0.000	85.034	1278.345	368.564	357.723	321.387	0.000
300	12-11:00	21.00	0.000	88.013	1307.722	278.842	357.438	323.757	0.000
301	12-12:00	21.40	0.000	85.490	1255.986	273.484	361.365	322.689	0.000
307	31-05:00	20.89	14.308	94.172	1327.832	251.120	351.263	332.485	1.52:

301 rows × 23 columns

In [11]:

```
#Fill missing values with forward fill
df2=df.fillna(method='ffill') # fill missing values with forward fill
df2
```

Out[11]:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	T-upperExt-2	T-lowerExt-2	UCZAI
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.545	1.44:
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.067	1.549
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.260	1.600
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.142	1.604
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.709	1.604
...
298	12-09:00	20.90	15.167	84.640	1283.706	339.440	354.803	311.041	1.63!
299	12-10:00	24.98	15.167	85.034	1278.345	368.564	357.723	321.387	1.63!
300	12-11:00	21.00	15.167	88.013	1307.722	278.842	357.438	323.757	1.63!
301	12-12:00	21.40	15.167	85.490	1255.986	273.484	361.365	322.689	1.63!
307	31-05:00	20.89	14.308	94.172	1327.832	251.120	351.263	332.485	1.52:

301 rows × 23 columns

In [12]:

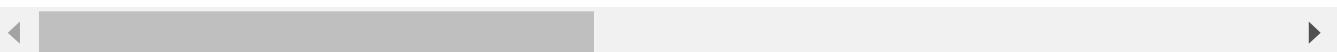
```
#Fill missing values with backward fill
df3=df.fillna(method='bfill') # fill missing values with backward fill
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

Out[12]:

	Observation	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	upperExt-2	T-lowerExt-2	T-UCZAA
0	31-00:00	23.10	16.520	121.717	1177.607	169.805	358.282	329.545	1.44
1	31-01:00	27.60	16.810	79.022	1328.360	341.327	351.050	329.067	1.54
2	31-02:00	23.19	16.709	79.562	1329.407	239.161	350.022	329.260	1.60
3	31-03:00	23.60	16.478	81.011	1334.877	213.527	350.938	331.142	1.60
4	31-04:00	22.90	15.618	93.244	1334.168	243.131	351.640	332.709	1.43
...
298	12-09:00	20.90	15.167	84.640	1283.706	339.440	354.803	311.041	1.63
299	12-10:00	24.98	14.308	85.034	1278.345	368.564	357.723	321.387	1.52
300	12-11:00	21.00	14.308	88.013	1307.722	278.842	357.438	323.757	1.52
301	12-12:00	21.40	14.308	85.490	1255.986	273.484	361.365	322.689	1.52
307	31-05:00	20.89	14.308	94.172	1327.832	251.120	351.263	332.485	1.52

301 rows × 23 columns



In [13]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats
```

In [14]:

df2.columns

Out[14]:

```
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')
```

In [15]:

```
df2.drop(['Observation'], axis=1, inplace=True)
df2.columns
```

Out[15]:

```
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')
```

In [16]:

```
Q1 = df2.quantile(0.25)
Q3 = df2.quantile(0.75)
IQR=Q3-Q1
print(IQR)
```

```

Y-Kappa           4.55000
ChipRate         2.12500
BF-CMratio      10.13800
BlowFlow          96.76600
ChipLevel4       105.86800
T-upperExt-2     12.05500
T-lowerExt-2     7.58900
UCZAA             0.13900
WhiteFlow-4      100.09800
AAWhiteSt-4      0.10550
AA-Wood-4        1.47800
ChipMoisture-4   2.18600
SteamFlow-4       8.84000
Lower-HeatT-3     8.57600
Upper-HeatT-3     7.78500
ChipMass-4        19.34700
WeakLiquorF      180.61300
BlackFlow-2       280.07500
WeakWashF         263.33500
SteamHeatF-3      6.90100
T-Top-Chips-4    2.02900
SulphidityL-4     0.69525
dtype: float64

```

```
In [17]: df2=df2[~((df2<(Q1-1.5*IQR))|(df2>(Q3+1.5*IQR))).any(axis=1)]
df2
```

Out[17]:

	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	T-upperExt-2	T-lowerExt-2	UCZAA	WhiteFlow-4
1	27.60	16.810	79.022	1328.360	341.327	351.050	329.067	1.549	537.201
2	23.19	16.709	79.562	1329.407	239.161	350.022	329.260	1.600	549.611
5	14.23	15.350	85.518	1171.604	198.538	344.014	325.195	1.436	628.245
6	13.49	13.700	98.186	1243.688	116.275	346.208	326.982	1.434	696.766
7	22.65	14.100	91.887	1307.852	288.989	352.321	331.162	1.468	625.549
...
298	20.90	15.167	84.640	1283.706	339.440	354.803	311.041	1.635	532.419
299	24.98	15.167	85.034	1278.345	368.564	357.723	321.387	1.635	520.365
300	21.00	15.167	88.013	1307.722	278.842	357.438	323.757	1.635	553.070
301	21.40	15.167	85.490	1255.986	273.484	361.365	322.689	1.635	590.199
307	20.89	14.308	94.172	1327.832	251.120	351.263	332.485	1.522	631.514

235 rows × 22 columns

```
In [18]: print("\nStatistical summary of the DataFrame:")
print(df2.describe())
```

Statistical summary of the DataFrame:

	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	\
count	235.000000	235.000000	235.000000	235.000000	235.000000	
mean	20.675489	14.602409	86.364787	1253.884400	265.499540	
std	2.944999	1.296108	6.833130	47.605312	69.787154	
min	12.480000	10.833000	68.645000	1084.083000	61.783000	
25%	18.465000	13.687500	81.314500	1213.838000	224.714000	
50%	20.800000	14.692000	85.534000	1278.006000	271.899000	
75%	23.005000	15.604000	91.729500	1288.466000	315.738500	
max	27.600000	16.958000	105.911000	1351.240000	394.234000	

	T-upperExt-2	T-lowerExt-2	UCZAA	WhiteFlow-4	AAWhiteSt-4	\
count	235.000000	235.000000	235.000000	235.000000	235.000000	
mean	356.919094	324.992796	1.488749	601.162217	6.142443	
std	7.548497	5.565984	0.103720	60.329201	0.067374	
min	340.222000	310.421000	1.224000	468.841000	5.940000	
25%	350.851500	321.829000	1.431000	549.484000	6.097000	
50%	357.438000	325.863000	1.488000	599.914000	6.140000	
75%	361.605000	329.161000	1.558500	652.770000	6.194000	
max	375.047000	337.012000	1.710000	731.394000	6.340000	

	...	SteamFlow-4	Lower-HeatT-3	Upper-HeatT-3	ChipMass-4	\
count	...	235.000000	235.000000	235.000000	235.000000	
mean	...	67.402953	325.031523	299.996872	164.141877	
std	...	4.982040	4.680400	4.547736	11.473545	
min	...	52.962000	318.051000	293.312000	133.878000	
25%	...	63.819000	321.215000	296.345000	156.110000	
50%	...	67.901000	322.412000	297.726000	164.250000	
75%	...	71.707500	329.787500	303.981000	172.555000	
max	...	75.974000	333.854000	311.146000	189.268000	

	WeakLiquorF	BlackFlow-2	WeakWashF	SteamHeatF-3	T-Top-Chips-4	\
count	235.000000	235.000000	235.000000	235.000000	235.000000	
mean	879.282800	1159.761268	268.771315	49.889251	251.196153	
std	116.360066	151.553026	164.891435	4.065122	1.218812	
min	596.446000	838.948000	0.000000	38.283000	248.359000	
25%	795.064000	1017.102500	142.538000	46.778000	250.325000	
50%	887.125000	1130.605000	284.877000	50.253000	251.253000	
75%	970.082000	1318.663500	413.789000	53.051500	252.232000	
max	1122.110000	1395.767000	715.715000	59.564000	254.122000	

	SulphidityL-4
count	235.000000
mean	30.488136
std	0.620493
min	29.110000
25%	30.120000
50%	30.450000
75%	30.910000
max	31.796000

[8 rows x 22 columns]

```
In [20]: # Group the DataFrame by a column and calculate the mean
print("\nGroup the DataFrame by a column and calculate the mean:")
print(df2.groupby("ChipRate").mean())
```

Group the DataFrame by a column and calculate the mean:

	Y-Kappa	BF-CMratio	BlowFlow	ChipLevel4	T-upperExt-2	\
ChipRate						
10.833	14.53	93.121	1202.818	339.119	342.535	
11.642	14.39	104.759	1244.014	198.128	361.416	
11.817	18.40	87.893	1268.595	384.941	359.381	
11.875	18.04	102.457	1242.285	223.763	356.604	
12.058	20.30	89.523	1180.205	300.596	348.649	
...	
16.817	21.50	78.969	1237.844	285.892	353.644	
16.825	18.13	84.836	1246.387	259.527	355.515	
16.833	18.30	91.341	1236.911	287.397	356.787	
16.867	23.50	76.937	1284.849	334.170	364.132	
16.958	25.40	74.405	1230.776	295.522	350.216	

	T-lowerExt-2	UCZAA	WhiteFlow-4	AAwhiteSt-4	AA-Wood-4	...	\
ChipRate							
10.833	324.194	1.258	601.694	6.049	18.447	...	
11.642	333.614	1.466	717.305	6.130	19.186	...	
11.817	328.182	1.253	620.819	6.207	19.194	...	
11.875	329.952	1.251	650.726	6.130	19.192	...	
12.058	318.905	1.548	577.128	6.090	16.822	...	
...	
16.817	324.500	1.437	621.887	6.215	17.757	...	
16.825	325.985	1.437	662.577	6.160	17.759	...	
16.833	327.856	1.540	661.532	6.160	17.759	...	
16.867	329.250	1.551	620.208	6.236	18.201	...	
16.958	322.740	1.471	540.151	6.115	17.754	...	

	SteamFlow-4	Lower-HeatT-3	Upper-HeatT-3	ChipMass-4	\
ChipRate					
10.833	63.969	322.040	299.894	146.584	
11.642	74.930	323.022	294.487	185.172	
11.817	68.653	322.267	298.114	159.595	
11.875	70.963	321.385	297.691	156.046	
12.058	66.904	329.492	306.086	173.064	
...	
16.817	68.750	321.235	296.616	176.460	
16.825	72.407	321.904	296.185	186.165	
16.833	71.891	323.285	297.755	187.777	
16.867	72.448	322.259	297.509	171.093	
16.958	52.962	318.913	295.843	146.914	

	WeakLiquorF	BlackFlow-2	WeakWashF	SteamHeatF-3	\
ChipRate					
10.833	602.040	838.948	48.632	41.508	
11.642	846.029	970.157	84.701	56.733	
11.817	1013.265	985.213	0.000	46.750	
11.875	923.950	886.286	43.873	50.948	
12.058	731.689	1291.897	291.689	47.304	
...	
16.817	852.132	1141.946	344.089	52.043	
16.825	810.272	1132.813	0.000	54.044	
16.833	841.552	1144.876	0.000	53.897	
16.867	888.854	1019.068	603.462	50.394	
16.958	727.366	1052.879	476.150	43.701	

	T-Top-Chips-4	SulphidityL-4
ChipRate		
10.833	251.630	30.65
11.642	249.982	30.54
11.817	250.958	31.60

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js 0.54
12.058 252.458 29.46

```
...          ...          ...
16.817      250.678      29.70
16.825      250.266      29.63
16.833      250.212      29.63
16.867      248.435      31.03
16.958      252.097      29.35
```

[193 rows x 21 columns]

```
In [21]: # Sort the DataFrame by a column
print("\nSort the DataFrame by a column:")
print(df2.sort_values("ChipRate"))
```

Sort the DataFrame by a column:

	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	T-upperExt-2	\
37	14.53	10.833	93.121	1202.818	339.119	342.535	
175	14.39	11.642	104.759	1244.014	198.128	361.416	
148	18.40	11.817	87.893	1268.595	384.941	359.381	
176	18.04	11.875	102.457	1242.285	223.763	356.604	
20	20.30	12.058	89.523	1180.205	300.596	348.649	
..	\
71	21.50	16.817	78.969	1237.844	285.892	353.644	
72	18.13	16.825	84.836	1246.387	259.527	355.515	
73	18.30	16.833	91.341	1236.911	287.397	356.787	
99	23.50	16.867	76.937	1284.849	334.170	364.132	
64	25.40	16.958	74.405	1230.776	295.522	350.216	
	T-lowerExt-2	UCZAA	WhiteFlow-4	AAWhiteSt-4	...	SteamFlow-4	\
37	324.194	1.258	601.694	6.049	...	63.969	
175	333.614	1.466	717.305	6.130	...	74.930	
148	328.182	1.253	620.819	6.207	...	68.653	
176	329.952	1.251	650.726	6.130	...	70.963	
20	318.905	1.548	577.128	6.090	...	66.904	
..	\
71	324.500	1.437	621.887	6.215	...	68.750	
72	325.985	1.437	662.577	6.160	...	72.407	
73	327.856	1.540	661.532	6.160	...	71.891	
99	329.250	1.551	620.208	6.236	...	72.448	
64	322.740	1.471	540.151	6.115	...	52.962	
	Lower-HeatT-3	Upper-HeatT-3	ChipMass-4	WeakLiquorF	BlackFlow-2	\	
37	322.040	299.894	146.584	602.040	838.948		
175	323.022	294.487	185.172	846.029	970.157		
148	322.267	298.114	159.595	1013.265	985.213		
176	321.385	297.691	156.046	923.950	886.286		
20	329.492	306.086	173.064	731.689	1291.897		
..	\
71	321.235	296.616	176.460	852.132	1141.946		
72	321.904	296.185	186.165	810.272	1132.813		
73	323.285	297.755	187.777	841.552	1144.876		
99	322.259	297.509	171.093	888.854	1019.068		
64	318.913	295.843	146.914	727.366	1052.879		
	WeakWashF	SteamHeatF-3	T-Top-Chips-4	SulphidityL-4			
37	48.632	41.508	251.630	30.65			
175	84.701	56.733	249.982	30.54			
148	0.000	46.750	250.958	31.60			
176	43.873	50.948	250.597	30.54			
20	291.689	47.304	252.438	29.46			
..	\
71	344.089	52.043	250.678	29.70			
72	0.000	54.044	250.266	29.63			
73	0.000	53.897	250.212	29.63			
99	603.462	50.394	248.435	31.03			
64	476.150	43.701	252.097	29.35			

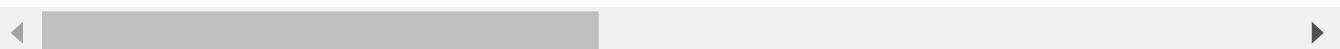
[235 rows x 22 columns]

```
In [22]: df2[df2['ChipRate'] == 11.817] # filter rows where column_name is equal to value
```

Out[22]:

	Y-Kappa	ChipRate	BF-CMratio	BlowFlow	ChipLevel4	upperExt-2	T-2	T-2	UCZAA	WhiteFlow-4
148	18.4	11.817	87.893	1268.595	384.941	359.381	328.182	1.253	620.819	

1 rows × 22 columns



In []: