pandas

In [1]:
```
import pandas as pd
```

In [2]:
```
# series in pandas
s=pd.Series([1,2,3,4])
s
```

Out[2]:
```
0    1
1    2
2    3
3    4
dtype: int64
```

In [3]:
```
pd.Series(["a","b"],index=[1,2])
```

Out[3]:
```
1    a
2    b
dtype: object
```

In [4]:
```
pd.Series({"a":1,"b":2})
```

Out[4]:
```
a    1
b    2
dtype: int64
```

In [5]:
```
# DataFrame in pandas
pd.DataFrame([1,2,3,4])
```

Out[5]:

	0
0	1
1	2
2	3
3	4

In [6]:
```
pd.DataFrame(["a","b","c"],index=[1,2,3])
```

Out[6]:

	0
1	a
2	b
3	c

In [7]:
```
pd.DataFrame(["a","b","c","d"],index=[0,1,2,3],columns=["A"])
```

Out[7]:

	A
0	a
1	b
2	c
3	d

In [8]:
```
pd.DataFrame({"A":[1,2,3,4],"B":[5,6,7,8]})
```

Out[8]:

	A	B
0	1	5
1	2	6
2	3	7
3	4	8

In [9]:
```
pd.DataFrame({
    "a":pd.Series([1,2,3,4]),
    "b":[5,6,7,8]
})
```

Out[9]:

	a	b
0	1	5
1	2	6
2	3	7
3	4	8

In [10]:
```
# how to read csv file
df=pd.read_csv("Text.csv")
df
```

Out[10]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [11]:
```
# show all columns
print(df.columns)
print(len(df.columns))
```

```
Index(['total_bill', 'tip', 'sex', 'smoker', 'day', 'time', 'size'], dtype='object')
7
```

In [12]:
```
# show no of row
pd.read_csv("Text.csv",nrows=3)
```

Out[12]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	NaN	No	NaN	Dinner	3
2	21.01	3.50	Male	No	Sun	NaN	3

In [13]:
```
# show no. of columns
print(pd.read_csv("Text.csv",usecols=[2]))
print(pd.read_csv("Text.csv",usecols=[1,3]))
```

```
       sex
0    Female
1       NaN
2      Male
3       NaN
4    Female
..      ...
239    Male
240  Female
241    Male
242    Male
243  Female

[244 rows x 1 columns]
      tip smoker
0    1.01     No
1    1.66     No
2    3.50     No
3     NaN    NaN
4    3.61     No
..    ...    ...
239  5.92     No
240  2.00    Yes
241  2.00    Yes
242  1.75     No
243  3.00     No

[244 rows x 2 columns]
```

In [14]:
```
# no. of skiprows
pd.read_csv("Text.csv",skiprows=2)
```

Out[14]:

	10.34	1.66	Unnamed: 2	No	Unnamed: 4	Dinner	3
0	21.01	3.50	Male	No	Sun	NaN	3.0
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	24.59	3.61	Female	No	Sun	Dinner	4.0
3	25.29	4.71	Male	No	NaN	Dinner	4.0
4	8.77	2.00	Male	No	Sun	Dinner	2.0
...	...	...	...	...	...	...	...
237	29.03	5.92	Male	No	Sat	Dinner	3.0
238	27.18	2.00	Female	Yes	Sat	Dinner	2.0
239	22.67	2.00	Male	Yes	Sat	Dinner	2.0
240	17.82	1.75	Male	No	Sat	Dinner	2.0
241	18.78	3.00	Female	No	Thur	Dinner	2.0

242 rows × 7 columns

In [15]:
```
pd.read_csv("Text.csv",skiprows=[2])
```

Out[15]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	21.01	3.50	Male	No	Sun	NaN	3.0
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	24.59	3.61	Female	No	Sun	Dinner	4.0
4	25.29	4.71	Male	No	NaN	Dinner	4.0
...	...	...	...	...	...	...	...
238	29.03	5.92	Male	No	Sat	Dinner	3.0
239	27.18	2.00	Female	Yes	Sat	Dinner	2.0
240	22.67	2.00	Male	Yes	Sat	Dinner	2.0
241	17.82	1.75	Male	No	Sat	Dinner	2.0
242	18.78	3.00	Female	No	Thur	Dinner	2.0

243 rows × 7 columns

In [16]:
```
# how to set index to particular columns
pd.read_csv("Text.csv",index_col="sex")
```

Out[16]:

sex	total_bill	tip	smoker	day	time	size
Female	16.99	1.01	No	Sun	Dinner	2.0
NaN	10.34	1.66	No	NaN	Dinner	3.0
Male	21.01	3.50	No	Sun	NaN	3.0
NaN	NaN	NaN	NaN	NaN	NaN	NaN
Female	24.59	3.61	No	Sun	Dinner	4.0
...	...	...	...	...	...	...
Male	29.03	5.92	No	Sat	Dinner	3.0
Female	27.18	2.00	Yes	Sat	Dinner	2.0
Male	22.67	2.00	Yes	Sat	Dinner	2.0
Male	17.82	1.75	No	Sat	Dinner	2.0
Female	18.78	3.00	No	Thur	Dinner	2.0

244 rows × 6 columns

In [17]:
```
# set a header to particular row in df
pd.read_csv("Text.csv",header=2)
```

Out[17]:

	10.34	1.66	Unnamed: 2	No	Unnamed: 4	Dinner	3
0	21.01	3.50	Male	No	Sun	NaN	3.0
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	24.59	3.61	Female	No	Sun	Dinner	4.0
3	25.29	4.71	Male	No	NaN	Dinner	4.0
4	8.77	2.00	Male	No	Sun	Dinner	2.0
...	...	...	...	...	...	...	...
237	29.03	5.92	Male	No	Sat	Dinner	3.0
238	27.18	2.00	Female	Yes	Sat	Dinner	2.0
239	22.67	2.00	Male	Yes	Sat	Dinner	2.0
240	17.82	1.75	Male	No	Sat	Dinner	2.0
241	18.78	3.00	Female	No	Thur	Dinner	2.0

242 rows × 7 columns

In [18]:
```
# how to remove header
pd.read_csv("Text.csv",header=None)
```

Out[18]:

	0	1	2	3	4	5	6
0	total_bill	tip	sex	smoker	day	time	size
1	16.99	1.01	Female	No	Sun	Dinner	2
2	10.34	1.66	NaN	No	NaN	Dinner	3
3	21.01	3.5	Male	No	Sun	NaN	3
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...
240	29.03	5.92	Male	No	Sat	Dinner	3
241	27.18	2	Female	Yes	Sat	Dinner	2
242	22.67	2	Male	Yes	Sat	Dinner	2
243	17.82	1.75	Male	No	Sat	Dinner	2
244	18.78	3	Female	No	Thur	Dinner	2

245 rows × 7 columns

In [19]:
```
# set a prefix on columns
pd.read_csv("Text.csv",header=None,prefix="column")
```

Out[19]:

	column0	column1	column2	column3	column4	column5	column6
0	total_bill	tip	sex	smoker	day	time	size
1	16.99	1.01	Female	No	Sun	Dinner	2
2	10.34	1.66	NaN	No	NaN	Dinner	3
3	21.01	3.5	Male	No	Sun	NaN	3
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...
240	29.03	5.92	Male	No	Sat	Dinner	3
241	27.18	2	Female	Yes	Sat	Dinner	2
242	22.67	2	Male	Yes	Sat	Dinner	2
243	17.82	1.75	Male	No	Sat	Dinner	2
244	18.78	3	Female	No	Thur	Dinner	2

245 rows × 7 columns

In [20]:
```
# how to a fix columns name
pd.read_csv("Text.csv",names=["a","b","c","d","e","f"])
```

Out[20]:

	a	b	c	d	e	f
total_bill	tip	sex	smoker	day	time	size
16.99	1.01	Female	No	Sun	Dinner	2
10.34	1.66	NaN	No	NaN	Dinner	3
21.01	3.5	Male	No	Sun	NaN	3
NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...
29.03	5.92	Male	No	Sat	Dinner	3
27.18	2	Female	Yes	Sat	Dinner	2
22.67	2	Male	Yes	Sat	Dinner	2
17.82	1.75	Male	No	Sat	Dinner	2
18.78	3	Female	No	Thur	Dinner	2

245 rows × 6 columns

In [21]:
```
# head method
df.head()
```

Out[21]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0

In [22]:
```
df.head(10)
```

Out[22]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
5	25.29	4.71	Male	No	NaN	Dinner	4.0
6	8.77	2.00	Male	No	Sun	Dinner	2.0
7	26.88	3.12	Male	No	Sun	Dinner	4.0
8	15.04	1.96	Male	No	Sun	Dinner	2.0
9	14.78	3.23	Male	No	Sun	Dinner	2.0

In [23]:
```
# tail method
df.tail()
```

Out[23]:

	total_bill	tip	sex	smoker	day	time	size
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

In [24]:
```
df.tail(10)
```

Out[24]:

	total_bill	tip	sex	smoker	day	time	size
234	15.53	3.00	Male	Yes	Sat	Dinner	2.0
235	10.07	1.25	Male	No	Sat	Dinner	2.0
236	12.60	1.00	Male	Yes	Sat	Dinner	2.0
237	32.83	1.17	Male	Yes	Sat	Dinner	2.0
238	35.83	4.67	Female	No	Sat	Dinner	3.0
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

In [25]:
```
# change datatype
pd.read_csv("Text.csv",dtype={"size":"float64"})
```

Out[25]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [26]:
```
df=pd.read_csv("Text.csv")
df
```

Out[26]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [27]:
```
# write true ,place of yes
pd.read_csv("Text.csv",true_values=["yes"])
```

Out[27]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [28]:
```
# write false ,place of no
pd.read_csv("Text.csv",false_values=["no"])
```

Out[28]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [29]:
```
# convert a string into Nan(not a no.)
pd.read_csv("Text.csv",na_values="No")
```

Out[29]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	NaN	Sun	Dinner	2.0
1	10.34	1.66	NaN	NaN	NaN	Dinner	3.0
2	21.01	3.50	Male	NaN	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	NaN	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	NaN	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	NaN	Sat	Dinner	2.0
243	18.78	3.00	Female	NaN	Thur	Dinner	2.0

244 rows × 7 columns

In [30]:
```
pd.read_csv("Text.csv",na_values={"time":"Dinner"})
```

Out[30]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	NaN	2.0
1	10.34	1.66	NaN	No	NaN	NaN	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	NaN	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	NaN	3.0
240	27.18	2.00	Female	Yes	Sat	NaN	2.0
241	22.67	2.00	Male	Yes	Sat	NaN	2.0
242	17.82	1.75	Male	No	Sat	NaN	2.0
243	18.78	3.00	Female	No	Thur	NaN	2.0

244 rows × 7 columns

In [31]:
```
pd.read_csv("Text.csv",keep_default_na=False)
```

Out[31]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66		No		Dinner	3
2	21.01	3.5	Male	No	Sun		3
3							
4	24.59	3.61	Female	No	Sun	Dinner	4
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2	Female	Yes	Sat	Dinner	2
241	22.67	2	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3	Female	No	Thur	Dinner	2

244 rows × 7 columns

In [32]:
```
# here we show none of value is NaN
pd.read_csv("Text.csv",na_filter=False)
```

Out[32]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66		No		Dinner	3
2	21.01	3.5	Male	No	Sun		3
3							
4	24.59	3.61	Female	No	Sun	Dinner	4
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2	Female	Yes	Sat	Dinner	2
241	22.67	2	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3	Female	No	Thur	Dinner	2

244 rows × 7 columns

In [33]:
```
# show true or false, where value is NaN
df.isnull()
```

Out[33]:

	total_bill	tip	sex	smoker	day	time	size
0	False	False	False	False	False	False	False
1	False	False	True	False	True	False	False
2	False	False	False	False	False	True	False
3	True	True	True	True	True	True	True
4	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...
239	False	False	False	False	False	False	False
240	False	False	False	False	False	False	False
241	False	False	False	False	False	False	False
242	False	False	False	False	False	False	False
243	False	False	False	False	False	False	False

244 rows × 7 columns

In [34]:
```
df.isnull().sum()
```

Out[34]:
```
total_bill    1
tip           1
sex           2
smoker        1
day           3
time          2
size          2
dtype: int64
```

In [35]:
```
df.isnull().sum().sum()
```

Out[35]:
```
12
```

In [36]:
```
df.notnull()
```

Out[36]:

	total_bill	tip	sex	smoker	day	time	size
0	True	True	True	True	True	True	True
1	True	True	False	True	False	True	True
2	True	True	True	True	True	False	True
3	False	False	False	False	False	False	False
4	True	True	True	True	True	True	True
...	...	...	...	...	...	...	...
239	True	True	True	True	True	True	True
240	True	True	True	True	True	True	True
241	True	True	True	True	True	True	True
242	True	True	True	True	True	True	True
243	True	True	True	True	True	True	True

244 rows × 7 columns

In [37]:
```
df.notnull().sum()
```

Out[37]:
```
total_bill    243
tip           243
sex           242
smoker        243
day           241
time          242
size          242
dtype: int64
```

In [38]:
```
df.notnull().sum().sum()
```

Out[38]:
```
1696
```

In [39]:
```
# how to drop NaN's values
df.dropna()
```

Out[39]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
6	8.77	2.00	Male	No	Sun	Dinner	2.0
7	26.88	3.12	Male	No	Sun	Dinner	4.0
8	15.04	1.96	Male	No	Sun	Dinner	2.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

239 rows × 7 columns

In [40]:
```
# how to drop NaN according to row
df.dropna(axis=0)
```

Out[40]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
6	8.77	2.00	Male	No	Sun	Dinner	2.0
7	26.88	3.12	Male	No	Sun	Dinner	4.0
8	15.04	1.96	Male	No	Sun	Dinner	2.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

239 rows × 7 columns

In [41]:
```
# how to drop NaN according to columns
df.dropna(axis=1)
```

Out[41]:

0
1
2
3
4
...
239
240
241
242
243

244 rows × 0 columns

In [42]:
```
# apply "how" condition
df.dropna(how="any")
```

Out[42]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
6	8.77	2.00	Male	No	Sun	Dinner	2.0
7	26.88	3.12	Male	No	Sun	Dinner	4.0
8	15.04	1.96	Male	No	Sun	Dinner	2.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

239 rows × 7 columns

In [43]:
```
df.dropna(how="all")
```

Out[43]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
5	25.29	4.71	Male	No	NaN	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

243 rows × 7 columns

In [44]:
```
# how many NaN contain a particular row
df.dropna(thresh=1)
```

Out[44]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
5	25.29	4.71	Male	No	NaN	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

243 rows × 7 columns

In [45]:
```
# how to drop NaN in a row according to acolumns
df.dropna(subset=["sex"])
```

Out[45]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
5	25.29	4.71	Male	No	NaN	Dinner	4.0
6	8.77	2.00	Male	No	Sun	Dinner	2.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

242 rows × 7 columns

In [46]:
```
# how to fill NaN values
df.fillna(1)
# here 1 replace all NaN values
```

Out[46]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	1	No	1	Dinner	3.0
2	21.01	3.50	Male	No	Sun	1	3.0
3	1.00	1.00	1	1	1	1	1.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [47]:
```
# how to fill NaN values in a particular column
df.fillna({"sex":"other"})
```

Out[47]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	other	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	other	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [48]:
```
# here fill value according to privious value
df.fillna(method="ffill")
```

Out[48]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	Female	No	Sun	Dinner	3.0
2	21.01	3.50	Male	No	Sun	Dinner	3.0
3	21.01	3.50	Male	No	Sun	Dinner	3.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [49]:
```
# here NaN value replace by back value
df.fillna(method="bfill")
```

Out[49]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	Male	No	Sun	Dinner	3.0
2	21.01	3.50	Male	No	Sun	Dinner	3.0
3	24.59	3.61	Female	No	Sun	Dinner	4.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [50]:
```
# here wo take a limit that how many Nan values replace in a particular columns
df.fillna(limit=1,method="bfill")
```

Out[50]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	Male	No	Sun	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	24.59	3.61	Female	No	Sun	Dinner	4.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [51]:
```
# replacing of values
df.replace("Female","2")
```

Out[51]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	2	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	2	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	2	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	2	No	Thur	Dinner	2.0

244 rows × 7 columns

In [52]:
```
df.replace({"day":"Sun"},"Mon")
```

Out[52]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Mon	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Mon	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Mon	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [53]:
```
# here replace all string into integer
df.replace("[A-Za-z]",0,regex=True)
```

Out[53]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	0.0	0.0	0.0	0.0	2.0
1	10.34	1.66	NaN	0.0	NaN	0.0	3.0
2	21.01	3.50	0.0	0.0	0.0	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	0.0	0.0	0.0	0.0	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	0.0	0.0	0.0	0.0	3.0
240	27.18	2.00	0.0	0.0	0.0	0.0	2.0
241	22.67	2.00	0.0	0.0	0.0	0.0	2.0
242	17.82	1.75	0.0	0.0	0.0	0.0	2.0
243	18.78	3.00	0.0	0.0	0.0	0.0	2.0

244 rows × 7 columns

In [54]:
```
df.replace({"sex":"[A-Za-z]"},2,regex=True)
```

Out[54]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	2.0	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	2.0	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	2.0	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	2.0	No	Sat	Dinner	3.0
240	27.18	2.00	2.0	Yes	Sat	Dinner	2.0
241	22.67	2.00	2.0	Yes	Sat	Dinner	2.0
242	17.82	1.75	2.0	No	Sat	Dinner	2.0
243	18.78	3.00	2.0	No	Thur	Dinner	2.0

244 rows × 7 columns

In [55]:
```
df.replace("Sun",method="bfill")
```

Out[55]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	NaN	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	NaN	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	NaN	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Male	Yes	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [56]:
```
df.replace("Yes",method="ffill")
```

Out[56]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3.0
240	27.18	2.00	Female	No	Sat	Dinner	2.0
241	22.67	2.00	Male	No	Sat	Dinner	2.0
242	17.82	1.75	Male	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [57]:
```
df.replace("Male",method="bfill",limit=5)
```

Out[57]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	NaN	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	24.59	3.61	Female	No	Sun	Dinner	4.0
...	...	...	...	...	...	...	...
239	29.03	5.92	Female	No	Sat	Dinner	3.0
240	27.18	2.00	Female	Yes	Sat	Dinner	2.0
241	22.67	2.00	Female	Yes	Sat	Dinner	2.0
242	17.82	1.75	Female	No	Sat	Dinner	2.0
243	18.78	3.00	Female	No	Thur	Dinner	2.0

244 rows × 7 columns

In [58]:
```
# access a group of rows and columns
df.loc[2]
```

Out[58]:
```
total_bill    21.01
tip             3.5
sex            Male
smoker           No
day             Sun
time            NaN
size              3
Name: 2, dtype: object
```

In [59]:
```
df.loc[1:3]
```

Out[59]:

	total_bill	tip	sex	smoker	day	time	size
1	10.34	1.66	NaN	No	NaN	Dinner	3.0
2	21.01	3.50	Male	No	Sun	NaN	3.0
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [60]:
```
# for choose particular row
df.loc[[2,4]]
```

Out[60]:

	total_bill	tip	sex	smoker	day	time	size
2	21.01	3.50	Male	No	Sun	NaN	3.0
4	24.59	3.61	Female	No	Sun	Dinner	4.0

In [61]:
```
# find out value of row's in a particular columns
df.loc[1:3,"sex"]
```

Out[61]:
```
1     NaN
2    Male
3     NaN
Name: sex, dtype: object
```

In [62]:
```
df.loc[df["tip"] < 2,["day"]]
```

Out[62]:

	day
0	Sun
1	NaN
8	Sun
10	Sun
12	Sun
16	Sun
30	Sat
43	Sun
53	Sun
57	Sat
58	Sat
62	Sat
67	Sat
70	Sat
75	Sat
82	Thur
92	Fri
97	Fri
99	Fri
105	Sat
111	Sat
117	Thur
118	Thur
121	Thur
126	Thur
130	Thur
132	Thur
135	Thur
145	Thur
146	Thur
147	Thur
148	Thur
168	Sat
190	Sun
195	Thur
215	Sat
217	Sat
218	Sat

	day
222	Fri
224	Fri
233	Sat
235	Sat
236	Sat
237	Sat
242	Sat

In [63]:
```
df.iloc[4]
```

Out[63]:
```
total_bill     24.59
tip             3.61
sex           Female
smoker            No
day              Sun
time          Dinner
size               4
Name: 4, dtype: object
```

In [64]:
```
df.iloc[1:5,1:4]
```

Out[64]:

	tip	sex	smoker
1	1.66	NaN	No
2	3.50	Male	No
3	NaN	NaN	NaN
4	3.61	Female	No

In [65]:
```
df.iloc[[0,1]]
```

Out[65]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0

In [66]:
```
df.iloc[0:2,:]
```

Out[66]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2.0
1	10.34	1.66	NaN	No	NaN	Dinner	3.0

In [67]:
```
gr=df.groupby(by="sex")
gr.groups
```

Out[67]:
```
{'Female': Int64Index([  0,   4,  11,  14,  16,  18,  21,  22,  29,  32,  33,  37,  51,
                52,  57,  66,  67,  71,  72,  73,  74,  82,  85,  92,  93,  94,
               100, 101, 102, 103, 104, 109, 111, 114, 115, 117, 118, 119, 121,
               124, 125, 127, 128, 131, 132, 133, 134, 135, 136, 137, 139, 140,
               143, 144, 145, 146, 147, 155, 157, 158, 162, 164, 168, 169, 178,
               186, 188, 191, 197, 198, 201, 202, 203, 205, 209, 213, 214, 215,
               219, 221, 223, 225, 226, 229, 238, 240, 243],
              dtype='int64'),
 'Male': Int64Index([  2,   5,   6,   7,   8,   9,  10,  12,  13,  15,
               ...
               231, 232, 233, 234, 235, 236, 237, 239, 241, 242],
              dtype='int64', length=155)}
```

In [68]:
```
list(gr.groups)
```

Out[68]:
```
['Female', 'Male']
```

In [69]:
```
dict(gr.groups)
```

Out[69]:
```
{'Female': Int64Index([  0,   4,  11,  14,  16,  18,  21,  22,  29,  32,  33,  37,  51,
                52,  57,  66,  67,  71,  72,  73,  74,  82,  85,  92,  93,  94,
               100, 101, 102, 103, 104, 109, 111, 114, 115, 117, 118, 119, 121,
               124, 125, 127, 128, 131, 132, 133, 134, 135, 136, 137, 139, 140,
               143, 144, 145, 146, 147, 155, 157, 158, 162, 164, 168, 169, 178,
               186, 188, 191, 197, 198, 201, 202, 203, 205, 209, 213, 214, 215,
               219, 221, 223, 225, 226, 229, 238, 240, 243],
              dtype='int64'),
 'Male': Int64Index([  2,   5,   6,   7,   8,   9,  10,  12,  13,  15,
               ...
               231, 232, 233, 234, 235, 236, 237, 239, 241, 242],
              dtype='int64', length=155)}
```

In [70]:
```
gr.mean()
```

Out[70]:

sex	total_bill	tip	size
Female	18.056897	2.833448	2.459770
Male	20.792258	3.097419	2.636364

In [71]:
```
# how merge 2 or more dataframe
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                 "class":["a","b","c","d"]})
pd.merge(df1,df2,on="id")
```

Out[71]:

	id	sn	class
0	1	5	a
1	2	6	b
2	3	7	c
3	4	8	d

In [72]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                 "class":["a","b","c","d"]})
pd.merge(df1,df2,on="id",how="left")
```

Out[72]:

	id	sn	class
0	1	5	a
1	2	6	b
2	3	7	c
3	4	8	d

In [73]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                 "class":["a","b","c","d"]})
pd.merge(df1,df2,how="right")
```

Out[73]:

	id	sn	class
0	1	5	a
1	2	6	b
2	3	7	c
3	4	8	d

In [74]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                 "class":["a","b","c","d"]})
pd.merge(df1,df2,how="outer",indicator=True)
```

Out[74]:

	id	sn	class	_merge
0	1	5	a	both
1	2	6	b	both
2	3	7	c	both
3	4	8	d	both

In [75]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1.1,2,3,4],
                 "class":["a","b","c","d"]})
pd.merge(df1,df2,left_index=True,right_index=True)
```

Out[75]:

	id_x	sn	id_y	class
0	1	5	1.1	a
1	2	6	2.0	b
2	3	7	3.0	c
3	4	8	4.0	d

In [76]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                 "class":["a","b","c","d"]})
pd.merge(df1,df2,on="id",suffixes=("x","y"))
```

Out[76]:

	id	sn	class
0	1	5	a
1	2	6	b
2	3	7	c
3	4	8	d

In [77]:
```
# how to combining data
sr1=pd.Series([1,2,3,4])
sr2=pd.Series([5,6,7,8,9])
pd.concat([sr1,sr2],ignore_index=True)
```

Out[77]:
```
0    1
1    2
2    3
3    4
4    5
5    6
6    7
7    8
8    9
dtype: int64
```

In [78]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                  "sn":["a","b","c","d"]})
pd.concat([df1,df2],ignore_index=True)
```

Out[78]:

	id	sn
0	1	5
1	2	6
2	3	7
3	4	8
4	1	a
5	2	b
6	3	c
7	4	d

In [79]:
```
pd.concat([df1,df2],ignore_index=True,axis=1)
```

Out[79]:

	0	1	2	3
0	1	5	1	a
1	2	6	2	b
2	3	7	3	c
3	4	8	4	d

In [80]:
```
pd.concat([df1,df2],ignore_index=True,axis=0)
```

Out[80]:

	id	sn
0	1	5
1	2	6
2	3	7
3	4	8
4	1	a
5	2	b
6	3	c
7	4	d

In [81]:
```
pd.concat([df1,df2],keys=["df1","df2"],axis=1)
```

Out[81]:

	df1		df2	
	id	**sn**	**id**	**sn**
0	1	5	1	a
1	2	6	2	b
2	3	7	3	c
3	4	8	4	d

In [82]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                  "sn1":["a","b","c","d"]})
pd.concat([df1,df2],sort=True)
```

Out[82]:

	id	sn	sn1
0	1	5.0	NaN
1	2	6.0	NaN
2	3	7.0	NaN
3	4	8.0	NaN
0	1	NaN	a
1	2	NaN	b
2	3	NaN	c
3	4	NaN	d

In [83]:
```
# how to join data
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id1":[1,2,3,4],
                  "sn1":["a","b","c","d"]})
df1.join(df2)
```

Out[83]:

	id	sn	id1	sn1
0	1	5	1	a
1	2	6	2	b
2	3	7	3	c
3	4	8	4	d

In [84]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id1":[1,2,3,4],
                  "sn1":["a","b","c","d"]})
df1.join(df2,how="left")
```

Out[84]:

	id	sn	id1	sn1
0	1	5	1	a
1	2	6	2	b
2	3	7	3	c
3	4	8	4	d

In [85]:
```
# how to append data
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                  "sn":["a","b","c","d"]})
df1.append(df2)
```

Out[85]:

	id	sn
0	1	5
1	2	6
2	3	7
3	4	8
0	1	a
1	2	b
2	3	c
3	4	d

In [86]:
```
df1=pd.DataFrame({"id":[1,2,3,4],
                  "sn":[5,6,7,8]})
df2=pd.DataFrame({"id":[1,2,3,4],
                 "sn":["a","b","c","d"]})
df1.append(df2,ignore_index=True)
```

Out[86]:

	id	sn
0	1	5
1	2	6
2	3	7
3	4	8
4	1	a
5	2	b
6	3	c
7	4	d

In [87]:
```
# when both df name is same
df1.append(df2,sort=False)
```

Out[87]:

	id	sn
0	1	5
1	2	6
2	3	7
3	4	8
0	1	a
1	2	b
2	3	c
3	4	d

In [ ]:
```

```