```
import pandas as pd
```

Summary Statistics

```
df.describe()
```

	total_bill	tip	size
count	244.000000	244.000000	244.000000
mean	19.785943	2.998279	2.569672
std	8.902412	1.383638	0.951100
min	3.070000	1.000000	1.000000
25%	13.347500	2.000000	2.000000
50%	17.795000	2.900000	2.000000
75%	24.127500	3.562500	3.000000
max	50.810000	10.000000	6.000000

Making a Series

```
a=pd.Series([1,2,3,4,5],index=["A","B","C","D","E"])
a
```

```
A    1
B    2
C    3
D    4
E    5
dtype: int64
```

Making a DataFrame

```
b=pd.DataFrame({"sher ali":21,"wasif":19,"umair":13},index=["A","B","C"])
b
```

	sher ali	wasif	umair
A	21	19	13

✓ 0s completed at 6:08 PM

C	21	19	13

Working on a dataset from seaborn library

```
import seaborn as sns
df=sns.load_dataset("tips")
df
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

244 rows × 7 columns

Checking information about data

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 244 entries, 0 to 243
Data columns (total 7 columns):
 #   Column      Non-Null Count  Dtype   
---  ------      --------------  -----   
 0   total_bill  244 non-null    float64 
 1   tip         244 non-null    float64 
 2   sex         244 non-null    category
 3   smoker      244 non-null    category
 4   day         244 non-null    category
```

```
 5   time        244 non-null    category
 6   size        244 non-null    int64
dtypes: category(4), float64(2), int64(1)
memory usage: 7.4 KB
```

checking first 5 enteries

```
df.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

Checking last 5 enteries

```
df.tail()
```

	total_bill	tip	sex	smoker	day	time	size
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

Checking no of rows and columns

```
df.shape
```

```
(244, 7)
```

Checking no of rows and columns by putting 0 and 1

```
df.shape[0]
```

```
244
```

Checking no of rows and columns by putting 0 and 1

```
df.shape[1]
```

```
7
```

Checking column names

```
df.columns
```

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

Checking row headings

```
df.index
```

```
RangeIndex(start=0, stop=244, step=1)
```

Removing specific columns

```
df1=df.drop(["day","smoker"],axis=1)
df1
```

	total_bill	tip	sex	time	size
0	16.99	1.01	Female	Dinner	2
1	10.34	1.66	Male	Dinner	3
2	21.01	3.50	Male	Dinner	3
3	23.68	3.31	Male	Dinner	2
4	24.59	3.61	Female	Dinner	4
...	...	...	...	...	...
239	29.03	5.92	Male	Dinner	3
240	27.18	2.00	Female	Dinner	2

241	22.67	2.00	Male	Dinner	2
242	17.82	1.75	Male	Dinner	2
243	18.78	3.00	Female	Dinner	2

244 rows × 5 columns

Checking missing values

```
df.isnull()
```

	total_bill	tip	sex	smoker	day	time	size
0	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False
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

Checking missing values totally

```
df.isnull().sum()
```

```
total_bill    0
tip           0
sex           0
smoker        0
day           0
time          0
size          0
dtype: int64
```

Checking unique values

code=df.column name.unique()

```
df.day.unique()
```

```
['Sun', 'Sat', 'Thur', 'Fri']
Categories (4, object): ['Thur', 'Fri', 'Sat', 'Sun']
```

Double-click (or enter) to edit