```
import os
os.getcwd()
```

```
    '/content'
```

```
import pandas as pd

df = pd.read_excel('/content/data.xlsx')
```

```
df
```

	roll	name	class	marks	age
0	1	Ravi	TE	67.50	23
1	2	Amit	BE	73.25	21
2	3	Deepika	TE	61.80	25
3	4	Rahul	BE	58.90	22
4	5	Priya	TE	70.10	24
5	6	Vivek	BE	64.70	20
6	7	Anita	TE	77.60	23
7	8	Rajesh	BE	69.45	22
8	9	Sneha	TE	66.30	21
9	10	Mohit	BE	72.15	24
10	11	Anjali	TE	75.90	20
11	12	Sanjay	BE	59.75	23
12	13	Neha	TE	63.40	22
13	14	Manoj	BE	68.20	24
14	15	Simran	TE	71.60	21
15	16	Arjun	BE	60.50	20
16	17	Kritika	TE	79.30	25
17	18	Abhishek	BE	65.85	24
18	19	Shivani	TE	76.70	22
19	20	Prateek	BE	55.20	23

Next steps: **Generate code with `df`**   ◑ **View recommended plots**

to check the shape of our data frame

```
df.shape
```

```
    (20, 5)
```

describe is for, to get the description of the data.

```
df.describe()
```

	roll	marks	age
count	20.00000	20.000000	20.000000
mean	10.50000	67.907500	22.450000
std	5.91608	6.771812	1.605091
min	1.00000	55.200000	20.000000
25%	5.75000	63.000000	21.000000
50%	10.50000	67.850000	22.500000
75%	15.25000	72.425000	24.000000
max	20.00000	79.300000	25.000000

head is for to display the first 5 rows of data sheet

```
df.head()
```

	roll	name	class	marks	age
0	1	Ravi	TE	67.50	23
1	2	Amit	BE	73.25	21
2	3	Deepika	TE	61.80	25
3	4	Rahul	BE	58.90	22
4	5	Priya	TE	70.10	24

Next steps: Generate code with `df`   View recommended plots

tail is used for displaying the last five rows of the whole data.

```
df.tail()
```

	roll	name	class	marks	age
15	16	Arjun	BE	60.50	20
16	17	Kritika	TE	79.30	25
17	18	Abhishek	BE	65.85	24
18	19	Shivani	TE	76.70	22
19	20	Prateek	BE	55.20	23

count: counts the total no of value of each attributes.

```
df.count()
```

```
roll     20
name     20
class    20
marks    20
age      20
dtype: int64
```

info used for getting the whole information like datatype, null count etc.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   roll    20 non-null     int64
 1   name    20 non-null     object
 2   class   20 non-null     object
 3   marks   20 non-null     float64
 4   age     20 non-null     int64
dtypes: float64(1), int64(2), object(2)
memory usage: 928.0+ bytes
```

Double-click (or enter) to edit

```
df.isnull()
```

	roll	name	class	marks	age
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
5	False	False	False	False	False
6	False	False	False	False	False
7	False	False	False	False	False
8	False	False	False	False	False
9	False	False	False	False	False
10	False	False	False	False	False
11	False	False	False	False	False
12	False	False	False	False	False
13	False	False	False	False	False
14	False	False	False	False	False
15	False	False	False	False	False
16	False	False	False	False	False
17	False	False	False	False	False
18	False	False	False	False	False
19	False	False	False	False	False

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

```
roll     0
name     0
class    0
marks    0
age      0
dtype: int64
```

```
print(True+True+True)
```

```
3
```

drop all rows that having no value

```
df.dropna()
```

	roll	name	class	marks	age
0	1	Ravi	TE	67.50	23
1	2	Amit	BE	73.25	21
2	3	Deepika	TE	61.80	25
3	4	Rahul	BE	58.90	22
4	5	Priya	TE	70.10	24
5	6	Vivek	BE	64.70	20
6	7	Anita	TE	77.60	23
7	8	Rajesh	BE	69.45	22
8	9	Sneha	TE	66.30	21
9	10	Mohit	BE	72.15	24
10	11	Anjali	TE	75.90	20
11	12	Sanjay	BE	59.75	23
12	13	Neha	TE	63.40	22
13	14	Manoj	BE	68.20	24
14	15	Simran	TE	71.60	21
15	16	Arjun	BE	60.50	20
16	17	Kritika	TE	79.30	25
17	18	Abhishek	BE	65.85	24
18	19	Shivani	TE	76.70	22
19	20	Prateek	BE	55.20	23

```
df.fillna(0)
```

	roll	name	class	marks	age
0	1	Ravi	TE	67.50	23
1	2	Amit	BE	73.25	21
2	3	Deepika	TE	61.80	25
3	4	Rahul	BE	58.90	22
4	5	Priya	TE	70.10	24
5	6	Vivek	BE	64.70	20
6	7	Anita	TE	77.60	23
7	8	Rajesh	BE	69.45	22
8	9	Sneha	TE	66.30	21
9	10	Mohit	BE	72.15	24
10	11	Anjali	TE	75.90	20
11	12	Sanjay	BE	59.75	23
12	13	Neha	TE	63.40	22
13	14	Manoj	BE	68.20	24
14	15	Simran	TE	71.60	21
15	16	Arjun	BE	60.50	20
16	17	Kritika	TE	79.30	25
17	18	Abhishek	BE	65.85	24
18	19	Shivani	TE	76.70	22
19	20	Prateek	BE	55.20	23

Double-click (or enter) to edit

```
df['class'].fillna('TE')
```

```
0     TE
1     BE
2     TE
3     BE
4     TE
5     BE
6     TE
7     BE
8     TE
9     BE
10    TE
11    BE
12    TE
13    BE
14    TE
15    BE
16    TE
17    BE
18    TE
19    BE
Name: class, dtype: object
```

```
df['marks'].fillna(df['marks'].mean())
```

```
0     67.50
1     73.25
2     61.80
3     58.90
4     70.10
5     64.70
6     77.60
7     69.45
8     66.30
9     72.15
10    75.90
11    59.75
12    63.40
13    68.20
14    71.60
15    60.50
16    79.30
17    65.85
18    76.70
19    55.20
Name: marks, dtype: float64
```

```
df['age'].fillna(df['age'].median())
```

```
0    23
1    21
2    25
```

```
3    22
4    24
5    20
6    23
7    22
8    21
9    24
10   20
11   23
12   22
13   24
14   21
15   20
16   25
17   24
18   22
19   23
Name: age, dtype: int64
```

```
df['class'].value_counts()
```

```
TE    10
BE    10
Name: class, dtype: int64
```

```
df['class'].fillna(df['class'].mode()[0])
```

```
0     TE
1     BE
2     TE
3     BE
4     TE
5     BE
6     TE
7     BE
8     TE
9     BE
10    TE
11    BE
12    TE
13    BE
14    TE
15    BE
16    TE
17    BE
18    TE
19    BE
Name: class, dtype: object
```

```
df.fillna(method='backfill')
```

	roll	name	class	marks	age
0	1	Ravi	TE	67.50	23
1	2	Amit	BE	73.25	21
2	3	Deepika	TE	61.80	25
3	4	Rahul	BE	58.90	22
4	5	Priya	TE	70.10	24
5	6	Vivek	BE	64.70	20
6	7	Anita	TE	77.60	23
7	8	Rajesh	BE	69.45	22
8	9	Sneha	TE	66.30	21
9	10	Mohit	BE	72.15	24
10	11	Anjali	TE	75.90	20
11	12	Sanjay	BE	59.75	23
12	13	Neha	TE	63.40	22
13	14	Manoj	BE	68.20	24
14	15	Simran	TE	71.60	21
15	16	Arjun	BE	60.50	20
16	17	Kritika	TE	79.30	25
17	18	Abhishek	BE	65.85	24
18	19	Shivani	TE	76.70	22
19	20	Prateek	BE	55.20	23

initializing the operations for the plotting, means using of matplotlib

```
import numpy as np
x=np.array([5,4,3,2,7,8,98,28])
np.mean(x)
```

```
19.375
```

```
np.median(x)
```

```
6.0
```

```
import matplotlib.pyplot as plt
plt.boxplot(x)
```

```
{'whiskers': [<matplotlib.lines.Line2D at 0x78b8b63873a0>,
  <matplotlib.lines.Line2D at 0x78b8b6387640>],
 'caps': [<matplotlib.lines.Line2D at 0x78b8b63878e0>,
  <matplotlib.lines.Line2D at 0x78b8b6387b80>],
 'boxes': [<matplotlib.lines.Line2D at 0x78b8b6387100>],
 'medians': [<matplotlib.lines.Line2D at 0x78b8b6387e20>],
 'fliers': [<matplotlib.lines.Line2D at 0x78b8b63c0100>],
 'means': []}
```
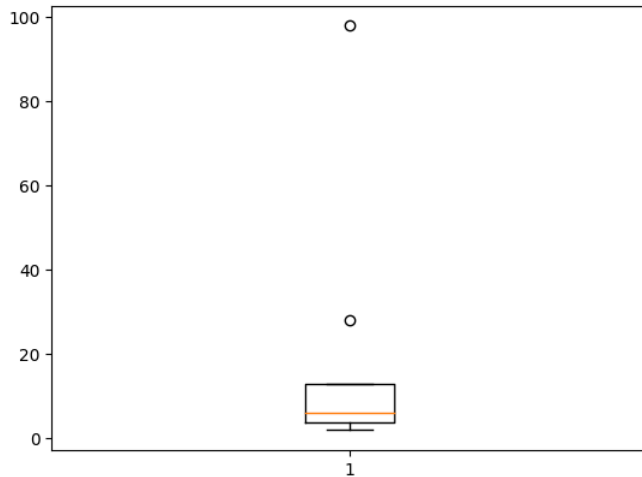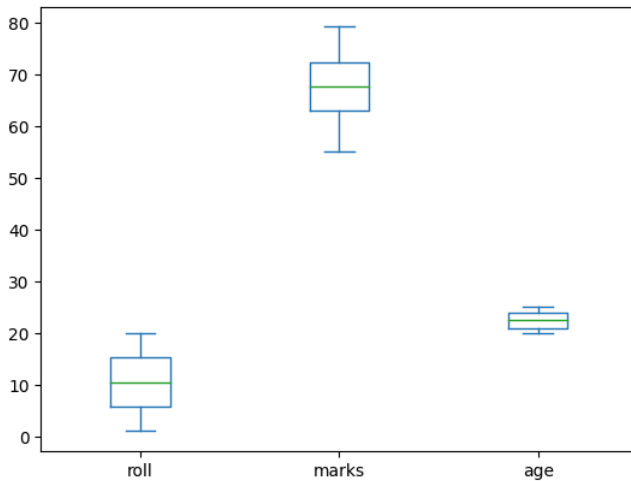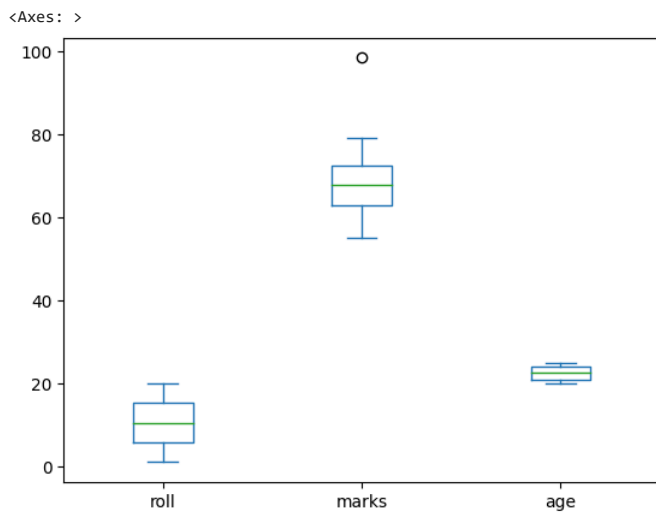


```
df.plot.box()
```

```
<Axes: >
```



```
df.loc[6,'marks']=98.45
df.plot.box()
```
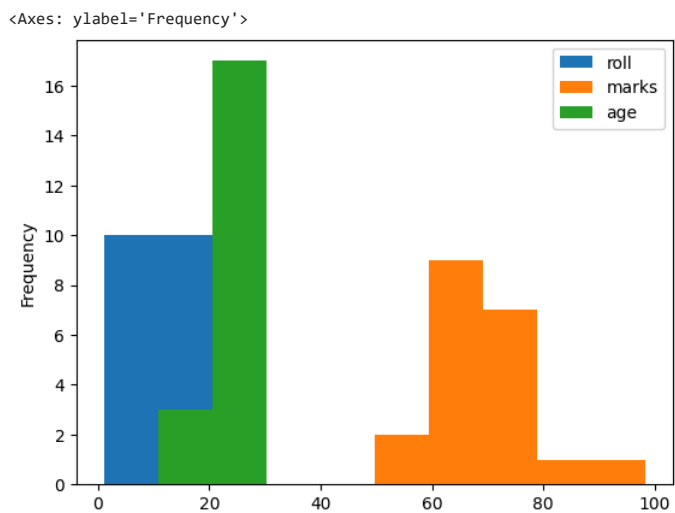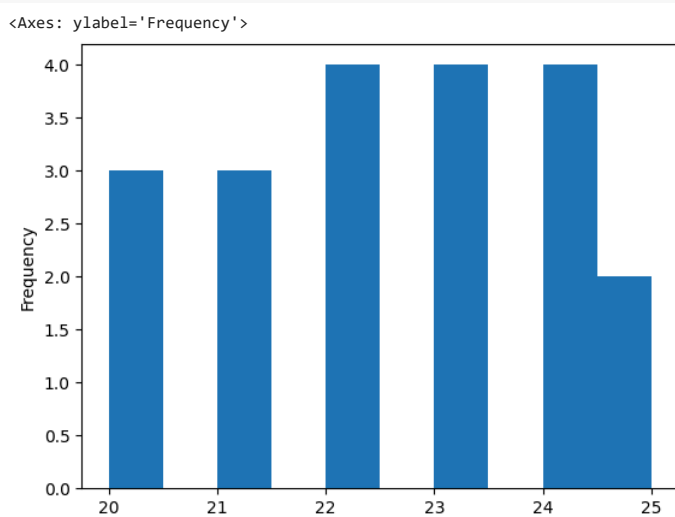
`<Axes: >`


```
df.plot.hist()
```

`<Axes: ylabel='Frequency'>`


```
df['age'].plot.hist()
```

`<Axes: ylabel='Frequency'>`


```
x = df[['age','marks']]
x.describe()
```

	age	marks	⊞
count	20.000000	20.000000	📊

```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
x_scaled = scaler.fit_transform(x)
```

```
pd.DataFrame(x_scaled).describe()
```

	0	1	⊞
count	20.000000	20.000000	📊
mean	0.490000	0.317919	
std	0.321018	0.217963	
min	0.000000	0.000000	
25%	0.200000	0.180347	
50%	0.500000	0.292486	
75%	0.800000	0.398266	