```
import pandas as pd
data=[10,20,30,40,50]
index_labels=['a','b','c','d','e']
series=pd.Series(data,index=index_labels)
print(series)
```

```
a    10
b    20
c    30
d    40
e    50
dtype: int64
```

```
import pandas as pd
data={
'name':['Alice','Bob','Claire','David'],
'age':[25,30,35,40],
'City':['hyderabad','Delhi','Chenni','Mumbai']
}
df=pd.DataFrame(data)
print("pandas dataframe")
print(df)
```

```
pandas dataframe
     name  age       City
0   Alice   25  hyderabad
1     Bob   30      Delhi
2  Claire   35     Chenni
3   David   40     Mumbai
```

```
print(df["name"])
```

```
0     Alice
1       Bob
2    Claire
3     David
Name: name, dtype: object
```

```
print(df[["name","City"]])
```

```
     name       City
0   Alice  hyderabad
1     Bob      Delhi
2  Claire     Chenni
3   David     Mumbai
```

```
print(df.iloc[1])
```

```
name      Bob
age        30
City    Delhi
Name: 1, dtype: object
```

```
print(df.loc[2])
```

```
name    Claire
age         35
City    Chenni
Name: 2, dtype: object
```

```
df["salary"]=[40000,50000,60000,70000]
```

```
df=df.drop("City",axis=1)
```

```
print(df)
```

```
     name  age  salary
0   Alice   25   40000
1     Bob   30   50000
2  Claire   35   60000
3   David   40   70000
```

```
print(df[df['age']>30])
```

```
     name  age  salary
2  Claire   35   60000
3   David   40   70000
```

```
print(df['age'].mean())
```

```
32.5
```

```
print(df['age'].max())
```

```
40
```

```
print(df['age'].min())
```

```
25
```

```
import pandas as pd
data={
'Student_id':[101,102,103,104],
'Name':['Alice','Bob','Claire','David'],
'marks':[85,90,78,92],
'department':['Cse','ee','Me','Cse']
}
df=pd.DataFrame(data)
print("Student data")
print(df)
```

```
Student data
   Student_id    Name  marks department
0         101   Alice     85        Cse
1         102     Bob     90         ee
2         103  Claire     78         Me
3         104   David     92        Cse
```

```
import numpy as np
import pandas as pd
data={
    'name':['Alice','Bob','Claire','David'],
    'age':[25,30,35,40],
    'marks':[20,45,60,80]
}
df=pd.DataFrame(data)
print(df)
print
```

```
     name  age  marks
0   Alice   25     20
1     Bob   30     45
2  Claire   35     60
3   David   40     80
<function print(*args, sep=' ', end='\n', file=None, flush=False)>
```

Start coding or generate with AI.

```
df_filled=df.fillna(0)
print("filled with 0")
print(df_filled)
df['age'] = df['age'].fillna(df['age'].mean())
df['marks'] = df['marks'].fillna(df['marks'].mean())
print('\nfilled with mean')
print(df)
```

```
filled with 0
     name  age  marks
0   Alice   25     20
1     Bob   30     45
2  Claire   35     60
3   David   40     80

filled with mean
     name  age  marks
0   Alice   25     20
1     Bob   30     45
2  Claire   35     60
3   David   40     80
```

```
data={
    'Id':[1,2,3,4,4],
    'name':['Alice','Bob','Claire','David','David'],
    'marks':[20,45,60,80,80]
}
df=pd.DataFrame(data)
print("orginal Dataframe with Duplicate")
print(df)
df_no_dup=df.drop_duplicates()
print("after removing  Duplicate")
print(df_no_dup)
```

```
orginal Dataframe with Duplicate
   Id    name  marks
0   1   Alice     20
1   2     Bob     45
2   3  Claire     60
3   4   David     80
4   4   David     80
after removing  Duplicate
   Id    name  marks
0   1   Alice     20
1   2     Bob     45
2   3  Claire     60
3   4   David     80
```

```
data={
    'Id':['1','2','3','4','4'],

    'marks':['20','45','60','80','80']
}
df=pd.DataFrame(data)
print("Original Data types")
print(df.dtypes)
df['Id']=pd.to_numeric(df['Id'])
```

```
df['marks']=pd.to_numeric(df['marks'])
print("\nAfter converting ")
print(df.dtypes)
```

```
Original Data types
Id       object
marks    object
dtype: object

After converting
Id       int64
marks    int64
dtype: object
```

```
df['marks']=df['marks'].apply(lambda x:x+10)
print("After adding 2 Bonus Marks")
print(df)
```

```
After adding 2 Bonus Marks
   Id  marks  Marks_normalized
0   1    180          0.000000
1   2    205          0.416667
2   3    220          0.666667
3   4    240          1.000000
4   4    240          1.000000
```

```
df['Marks_normalized']=(df['marks']-df['marks'].min())/(df['marks'].max()-df['marks'].min())
print("normalized Marks")
print(df)
```

```
normalized Marks
   Id  marks  Marks_normalized
0   1     40          0.000000
1   2     65          0.416667
2   3     80          0.666667
3   4    100          1.000000
4   4    100          1.000000
```