DATE : 21 june 2024

DAY : Friday

TOPICS : Worksheet of numpy and pandas

```
import numpy as np
import pandas as pd
```

```
l1 = [1,2,3,4,5,6]
labels = ['a','b','c','d','e','f']
d1 = {"A":10,"B":20,"C":30,"D":40,"E":50}
```

```
s1 = pd.Series(l1)
s1
```

```
   0
0  1
1  2
2  3
3  4
4  5
5  6

dtype: int64
```

```
s1[4]
```

```
5
```

```
s1 = pd.Series(labels)
s1
```

```
   0
0  a
1  b
2  c
3  d
4  e
5  f
```

```
s3 = pd.Series(data = l1,index = labels)
s3
```

```
   0
a  1
b  2
c  3
d  4
e  5
f  6
```

```
s3['a']
```

```
1
```

```
s3[0]
```

```
<ipython-input-8-e48481fcb92e>:1: FutureWarning: Series.__getitem__ treating keys as positions is deprecated. In a future version, integ
  s3[0]
1
```

```
pd.Series(d1)
```

	0
A	10
B	20
C	30
D	40
E	50

```
arr = np.random.randint(1,100,size = (5,6))
arr
```

```
array([[61, 95, 49, 43, 76, 79],
       [21, 90, 82, 56, 31, 33],
       [ 9,  7, 20, 85, 59, 26],
       [22, 62, 15, 29,  2, 17],
       [81, 13, 78, 62,  4, 68]])
```

```
type(arr)
```

```
numpy.ndarray
```

```
pd.DataFrame(arr)
```

	0	1	2	3	4	5
0	61	95	49	43	76	79
1	21	90	82	56	31	33
2	9	7	20	85	59	26
3	22	62	15	29	2	17
4	81	13	78	62	4	68

```
df = pd.DataFrame(arr, index=["A","B","C","D","E"], columns= ["U","V","W","X","Y","Z"])
df
```

	U	V	W	X	Y	Z
A	61	95	49	43	76	79
B	21	90	82	56	31	33
C	9	7	20	85	59	26
D	22	62	15	29	2	17
E	81	13	78	62	4	68

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
type(df)
```

```
pandas.core.frame.DataFrame
def __init__(data=None, index: Axes | None=None, columns: Axes | None=None, dtype: Dtype |
None=None, copy: bool | None=None) -> None

/usr/local/lib/python3.10/dist-packages/pandas/core/frame.py
Two-dimensional, size-mutable, potentially heterogeneous tabular data.

Data structure also contains labeled axes (rows and columns).
Arithmetic operations align on both row and column labels. Can be
thought of as a dict-like container for Series objects. The primary
```

```python
df['X']
```

```
   X
A  43
B  56
C  85
D  29
E  62

dtype: int64
```

```python
df[["X","Y","Z"]]
```

```
   X   Y   Z
A  43  76  79
B  56  31  33
C  85  59  26
D  29   2  17
E  62   4  68
```

```python
np.__version__
```

```
'1.26.4'
```

```python
pd.__version__
```

```
'2.2.2'
```

```python
df.loc["A"]
```

```
   A
U  61
V  95
W  49
X  43
Y  76
Z  79

dtype: int64
```

```python
df.iloc[0]
```

```
   A
U  61
V  95
W  49
X  43
Y  76
Z  79
```

```
#To add new column
df['new']=[100,200,300,400,500]
df
```

```
    U   V   W   X   Y   Z  new
A  61  95  49  43  76  79  100
B  21  90  82  56  31  33  200
C   9   7  20  85  59  26  300
D  22  62  15  29   2  17  400
E  81  13  78  62   4  68  500
```

Next steps: Generate code with df View recommended plots New interactive sheet

```
#To remove any column but it is not permanent
df.drop('new',axis=1)
```

```
    U   V   W   X   Y   Z
A  61  95  49  43  76  79
B  21  90  82  56  31  33
C   9   7  20  85  59  26
D  22  62  15  29   2  17
E  81  13  78  62   4  68
```

```
df
```

```
    U   V   W   X   Y   Z  new
A  61  95  49  43  76  79  100
B  21  90  82  56  31  33  200
C   9   7  20  85  59  26  300
D  22  62  15  29   2  17  400
E  81  13  78  62   4  68  500
```

Next steps: Generate code with df View recommended plots New interactive sheet

```
#remove any column permanently
df.drop('new',axis=1,inplace=True)
df
```

```
    U   V   W   X   Y   Z
A  61  95  49  43  76  79
```

Next   Generate code with df   View recommended plots   New interactive sheet

```
#conditional selection
df["X"] % 2 == 0
```

```
       X
A  False
B   True
C  False
D  False
E   True
```

```
df[df["X"] % 2 == 0]
```

```
    U   V   W   X   Y   Z
B  21  90  82  56  31  33
E  81  13  78  62   4  68
```

```
df[df["X"] % 2 == 0]["Y"]
```

```
A    83
B    49
C    77
Name: Y, dtype: int64
```

```
(df["X"] % 2 == 0) & (df["X"] > 50)
```

```
       X
A  False
B   True
C  False
D  False
E   True
```

```
df[(df["X"] % 2 == 0) & (df["X"] > 50)]
```

```
    U   V   W   X   Y   Z
B  21  90  82  56  31  33
E  81  13  78  62   4  68
```