In [3]:
```
import pandas as pd
import numpy as np
arr=np.array(['p','a','n','d','a','s'])
a=pd.series(arr)
print("series from array:")
print(a)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[3], line 4
      2 import numpy as np
      3 arr=np.array(['p','a','n','d','a','s'])
----> 4 a=pd.series(arr)
      5 print("series from array:")
      6 print(a)

AttributeError: module 'pandas' has no attribute 'series'
```

In [4]:
```
import pandas as pd
import numpy as np
arr=np.array(['p','a','n','d','a','s'])
a=pd.Series(arr)
print("series from array:")
print(a)
```

```
series from array:
0    p
1    a
2    n
3    d
4    a
5    s
dtype: object
```

In [5]:
```
 x=pd.Series(4,index=[0,1,2,3])
print("\nSeries using scalar\n")
print("\nSeries using scalar\n")
print(x)
```

```

Series using scalar


Series using scalar

0    4
1    4
2    4
3    4
dtype: int64
```

In [14]:
```
a=pd.Series(data=[1,2,3,4])
print("\n a Series:
```

```
 a Series:
 0    1
1    2
2    3
3    4
dtype: int64

---------------------------------------------------------------------------
-
NameError                                 Traceback (most recent call las
t)
Cell In[14], line 5
      1 a=pd.Series(data=[1,2,3,4])
      2 print("\n a Series:\n",a)
----> 5 D,D

NameError: name 'D' is not defined
```

In [6]:
```
x=pd.Series([1,2,3],index=['a','d','c'])
print("\nSeries through index:")
print(x)
```

```
Series through index:
a    1
d    2
c    3
dtype: int64
```

In [8]:
```
print("\nIndex: \n",a.index)
print("\nValues: \n",a.values)
print("\nShape: \n",a.shape)
print("\nDimension: \n",a.ndim)
print("\nSize: \n",a.size)
```

```
Index:
 RangeIndex(start=0, stop=4, step=1)

Values:
 [1 2 3 4]

Shape:
 (4,)

Dimension:
 1

Size:
 4
```

In [9]:
```
print("\nAdding new column:\n")
result['Age']=pd.Series([35,24,40,38])
print(result)
```

```
Adding new column:


---------------------------------------------------------------------------
NameError                                 Traceback (most recent call las
t)
Cell In[9], line 2
      1 print("\nAdding new column:\n")
----> 2 result['Age']=pd.Series([35,24,40,38])
      3 print(result)

NameError: name 'result' is not defined
```

In [1]:
```
emp=pd.Series(['Parker','John','Smith','William'])
id=pd.Series([102,107,109,114])
frame={'Emp':emp,'ID':id}
result=pd.DataFrame(frame)
print("\nSeries to Data frame\n")
print(result)
print("\nExtracting one column:\n")
print(result['Emp'])
print("\nExtracting the third row:\n")
print(result.loc[2])
print("\nAdding new column:\n")
result['Age']=pd.Series([35,24,40,38])
print(result)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call las
t)
Cell In[1], line 1
----> 1 emp=pd.Series(['Parker','John','Smith','William'])
      2 id=pd.Series([102,107,109,114])
      3 frame={'Emp':emp,'ID':id}

NameError: name 'pd' is not defined
```

In [10]:
```
emp=pd.Series(['Parker','John','Smith','William'])
id=pd.Series([102,107,109,114])
frame={'Emp':emp,'ID':id}
result=pd.DataFrame(frame)
print("\nSeries to Data frame\n")
print(result)
print("\nExtracting one column:\n")
print(result['Emp'])
print("\nExtracting the third row:\n")
print(result.loc[2])
```

```
Series to Data frame

       Emp   ID
0   Parker  102
1     John  107
2    Smith  109
3  William  114

Extracting one column:

0     Parker
1       John
2      Smith
3    William
Name: Emp, dtype: object

Extracting the third row:
```

In [13]:
```
arr={'x':0,'y':1,'z':2.}
b=pd.Series(arr)
print("\n\n Series from dictionary:\n")
print(b)
```

```
 Series from dictionary:

x    0.0
y    1.0
z    2.0
dtype: float64
```

In [25]:
```
mport pandas as pd
a=pd.dataframe([101,94,91,98,94,96],[102,94,98,97,99,96],[103,90,98,96,94,97
print(a.arr(['min','max']))
a[result]=[pass,pass,pass,pass,fail]
print[a.drop(103)]
a[['sai']]
print(a[,name])
a=np.dataframe([1101,'Dharun','EEE'],[1102,'Suriya','ECE'],[1103,'Vetri','CS
b=pd.dataframe([101,95,93],[102,98,99],[103,96,97])
columns=['rollno','m1','m2']
c=pd.merge(a,b='rollno')
print(c)
e.to_exl('work data.xlsx')
```

```
  Cell In[25], line 4
    a[result]=[pass,pass,pass,pass,fail]
                ^
SyntaxError: invalid syntax
```

In [20]:
```
import matplotlib.pyplot as plt
x=np.array([1,2,3])
y=np.array([5,4,3])
plt.plot=(x)
plt.plot=(y)
plt.show()
```

In [21]:
```
import matplotlib.pyplot as plt
x=np.array([1,2,3])
y=np.array([5,4,3])
```

In [1]:
```
import datetime as d
r=d.datetime.now()
print(r)
```

```
2024-08-21 10:36:42.213893
```

In [2]:
```
import datetime as d
r=d.datetime.today()
print(r)
```

```
2024-08-21 10:37:01.644634
```

In [8]:
```
  import datetime as d
r=d.datetime.now()
tomo=r+d.timedelta(days=1)
print(tomo)
```

```
2024-08-22 10:44:22.325396
```

In [6]:
```
  import datetime as d
r=d.datetime.now()
tomo=r+d.timedelta(days=2)
print(tomo)
```

```
2024-08-23 10:41:00.756494
```

In [7]:
```
  import datetime as d
r=d.datetime.now()
tomo=r-d.timedelta(days=2)
print(tomo)
```

```
2024-08-19 10:41:22.796116
```

In [10]:
```
import datetime as d
r1=d.datetime(2020,8,8,23,10,25,404040)
print(r1)
```

```
2020-08-08 23:10:25.404040
```

In [11]:
```
  import datetime as d
r2=d.datetime(2024,8,8,10,5,26,5485)
print(r2)
```

```
2024-08-08 10:05:26.005485
```

In [13]:
```
  import datetime as d
r2=d.datetime(2024,8,8,10,5,26,548500)
print(r2)
```

```
2024-08-08 10:05:26.548500
```

In [16]:
```
  import datetime as d
r2=d.datetime(2023,7,16,5,30,26,975485)
print(r2)
```

```
2023-07-16 05:30:26.975485
```

In [17]:
```
print(r2.replace(day=15))
```

```
2023-07-15 05:30:26.975485
```

In [18]:
```
print(r2.replace(month=4))
```

```
2023-04-16 05:30:26.975485
```

In [19]:
```
print(r2.replace(year=2024))
```

```
2024-07-16 05:30:26.975485
```

In [20]:
```
print(r2.replace(year=2024,month=5,day=30))
```

```
2024-05-30 05:30:26.975485
```

In [26]:
```
from datetime import date
print(date(2023,7,16))
```

```
2023-07-16
```

In [28]:
```
from datetime import date
print(date(2023,7,16).ctime())
```

```
Sun Jul 16 00:00:00 2023
```

In [29]:
```
print(r.strftime("%Y"))
```

```
2024
```

In [30]:
```
print(r.strftime("%y"))
```

```
24
```

In [32]:
```
print(r.strftime("%m"))
```

```
08
```

In [35]:
```
print(r.strftime("%b"))
```

```
Aug
```

In [36]:
```
print(r.strftime("%B"))
```

```
August
```

In [37]:
```
print(r.strftime("%j"))
```

```
234
```

In [39]:
```
print(r2.strftime("%D"))
```

```
07/16/23
```

In [40]:
```
print(r2.strftime("%j"))
```

```
197
```

In [41]:
```
print(r2.strftime("%d"))
```

```
16
```

In [43]:
```
print(r2.strftime("%a"))
```

```
Sun
```

In [44]:
```
print(r2.strftime("%A"))
```

```
Sunday
```

In [45]:
```
print(r2.strftime("%H"))
```

```
05
```

In [46]:
```
print(r2.strftime("%h"))
```

```
Jul
```

In [48]:
```
print(r2.strftime("%S"))
```

```
26
```

In [49]:
```
print(r2.strftime("%F"))
```

```
2023-07-16
```

In [50]:
```
print(r2.strftime("%p"))
```

```
AM
```

In [51]:
```
print(r2.strftime("%x"))
```

```
07/16/23
```

In [52]:
```
print(r2.strftime("%f"))
```

```
975485
```

In [53]:
```
print(r2.strftime("%X"))
```

```
05:30:26
```

In [ ]:
```

```