In [2]:

```python
#PANDAS Library
import pandas as pd

# Creating a Series
s = pd.Series([11, 21, 31, 41, 51], index=['A','B','C','D','E'])

print("Pandas Series:")
print(s)
print("\nAccess single value:", s['C'])
```

```
Pandas Series:
A    11
B    21
C    31
D    41
E    51
dtype: int64

Access single value: 31
```

In [3]:

```python
type(s)
```

Out[3]:

```
pandas.core.series.Series
```

In [4]:

```python
# Creating a DataFrame
data = {
    'Name': ['Mr. James', 'Mr. Thomas', 'Mr. Albert Francis', 'Mrs. John Pills
    'Sex': ['Male', 'Male','Male','Female'],
    'Age': [88, 92, 76,48]
}

df = pd.DataFrame(data)

print("Pandas DataFrame:")
print(df)
print("\nAccess single column (as Series):")
print(df['Age'])
```

```
Pandas DataFrame:
                  Name     Sex  Age
0            Mr. James    Male   88
1           Mr. Thomas    Male   92
2   Mr. Albert Francis    Male   76
3  Mrs. John Pillsbury  Female   48

Access single column (as Series):
0    88
1    92
2    76
3    48
Name: Age, dtype: int64
```

In [6]:

```python
type(df)
```

```
Out[6]: pandas.core.frame.DataFrame
```

```
In [22]: df1=pd.read_csv("Titanic.csv")         # load CSV file
```

```
In [7]: #perform arithmatic operation
        s = pd.Series([10, 20, 30, 40, 50])
        print("Original:\n", s)
        print("\nMultiplied by 2:\n", s * 2)
```

```
Original:
 0    10
1    20
2    30
3    40
4    50
dtype: int64

Multiplied by 2:
 0     20
1     40
2     60
3     80
4    100
dtype: int64
```

```
In [17]: print(df['Name'])             # select single column

         print(df.iloc[0])             # first row by index

         print(df.iloc[1:4])           # rows by index range

         print(df.loc[df['Age']>85]) # filter condition
```

```
0              Mr. James
1             Mr. Thomas
2     Mr. Albert Francis
3    Mrs. John Pillsbury
Name: Name, dtype: object
Name    Mr. James
Sex          Male
Age            88
Name: 0, dtype: object
                  Name     Sex  Age
1           Mr. Thomas    Male   92
2   Mr. Albert Francis    Male   76
3  Mrs. John Pillsbury  Female   48
         Name   Sex  Age
0   Mr. James  Male   88
1  Mr. Thomas  Male   92
```

```
In [10]: df.head()        # first 5 rows
```

Out[10]:

	Name	Sex	Age
0	Mr. James	Male	88
1	Mr. Thomas	Male	92
2	Mr. Albert Francis	Male	76
3	Mrs. John Pillsbury	Female	48

In [11]:

```
df.tail()        # last 5 rows
```

Out[11]:

	Name	Sex	Age
0	Mr. James	Male	88
1	Mr. Thomas	Male	92
2	Mr. Albert Francis	Male	76
3	Mrs. John Pillsbury	Female	48

In [12]:

```
df.shape         # (rows, columns)
```

Out[12]:

```
(4, 3)
```

In [13]:

```
df.info()        # summary (data types, non-null count)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4 entries, 0 to 3
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   Name    4 non-null      object
 1   Sex     4 non-null      object
 2   Age     4 non-null      int64 
dtypes: int64(1), object(2)
memory usage: 228.0+ bytes
```

In [14]:

```
df.dtypes        # data types of each column
```

Out[14]:

```
Name    object
Sex     object
Age      int64
dtype: object
```

In [15]:

```
df.describe()    # summary statistics (mean, std, min, max, etc.)
```

Out[15]:

	Age
count	4.000000
mean	76.000000
std	19.866219
min	48.000000
25%	69.000000
50%	82.000000
75%	89.000000
max	92.000000

```
In [16]: df.columns       # list of column names
```

```
Out[16]: Index(['Name', 'Sex', 'Age'], dtype='object')
```

```
In [18]: df.describe()          # summary statistics (mean, std, min, max, etc.)
```

Out[18]:

	Age
count	4.000000
mean	76.000000
std	19.866219
min	48.000000
25%	69.000000
50%	82.000000
75%	89.000000
max	92.000000

```
In [24]: df1['Age'].mean()     # average Age
```

```
Out[24]: np.float64(30.272590361445783)
```

```
In [23]: df1['Fare'].median()   # median
```

```
Out[23]: np.float64(14.4542)
```

```
In [26]: df1['PassengerId'].max()      # maximum value
```

```
Out[26]: np.int64(1309)
```

```
In [28]: df1['Age'].min()      # minimum value
```

```
Out[28]: np.float64(0.17)
```