

## Series Exercises:

### 1. Create a Series:

```
In [1]: import pandas as pd
```

```
In [3]: fruits_series=pd.Series(['apple', 'banana','cherry', 'date'],  
                                index=['a','b','c','d'])  
print(fruits_series)  
print(fruits_series.index)
```

```
a    apple  
b   banana  
c    cherry  
d     date  
dtype: object  
Index(['a', 'b', 'c', 'd'], dtype='object')
```

### 2. Accessing Elements:

```
In [4]: print(fruits_series["b"])
```

```
banana
```

### 3. Series Operations:

```
In [11]: onetofive_series=pd.Series([1,4,9,16,25],  
                                     index=[1,2,3,4,5])  
updated_onetofive_series = onetofive_series + 10  
print(onetofive_series)  
print(updated_onetofive_series)
```

```
1      1  
2      4  
3      9  
4     16  
5     25  
dtype: int64  
1     11  
2     14  
3     19  
4     26  
5     35  
dtype: int64
```

## DataFrame Exercises:

### 1. Create Dataframe:

```
In [19]: #create ficitional students:  
data = [  
    ['Jerry', 28, 'C'],  
    ['Leslie', 20, 'A'],
```

```

    ['April', 19, 'B']
]

students = pd.DataFrame(data, columns=['Name', 'Age', 'Grade'])

students

```

Out[19]:

	Name	Age	Grade
0	Jerry	28	C
1	Leslie	20	A
2	April	19	B

## 2. Accessing Data:

In [18]:

```

print(students["Name"])

```

```

0    Jerry
1    Leslie
2    April
Name: Name, dtype: object

```

## 3. DataFrame Operations:

In [21]:

```

grad=["False", "True", "False"]

students["Graduated"] = grad
print(students)

```

	Name	Age	Grade	Graduated
0	Jerry	28	C	False
1	Leslie	20	A	True
2	April	19	B	False

## 3. Maths:

In [23]:

```

print(students["Age"].mean())

```

```

22.333333333333332

```

## 5. Sorting:

In [31]:

```

students.sort_values("Age", ascending=False)

```

Out[31]:

	Name	Age	Grade	Graduated
0	Jerry	28	C	False
1	Leslie	20	A	True
2	April	19	B	False

## 6. Tidy Data:

```
In [39]: #make dataframe
cog_health=[
    [10,2,5,3],
    [15,4,9,6],
    [18,7,12,9],
    [13,2,7,4]
]

cog_health_df=pd.DataFrame(cog_health, columns=['happiness', 'language','energy', 'memory'],
                           index = ['Subject 1', 'Subject 2','Subject 3', 'Subject 4'])

cog_health_df
```

```
Out[39]:
```

	happiness	language	energy	memory
Subject 1	10	2	5	3
Subject 2	15	4	9	6
Subject 3	18	7	12	9
Subject 4	13	2	7	4

```
In [40]: ## mean/standard deviation

#happyness:
print(cog_health_df["happiness"].mean())
print(cog_health_df["happiness"].std())

#language:
print(cog_health_df["language"].mean())
print(cog_health_df["language"].std())

#energy:
print(cog_health_df["energy"].mean())
print(cog_health_df["energy"].std())

#memory:
print(cog_health_df["memory"].mean())
print(cog_health_df["memory"].std())
```

```
14.0
3.366501646120693
3.75
2.362907813126304
8.25
2.9860788111948193
5.5
2.6457513110645907
```

```
In [44]: ## mean/standard error of happiness:

print(cog_health_df["happiness"].mean())
print(cog_health_df["happiness"].sem())
```

```
14.0
1.6832508230603465
```