



Pandas Assignment

```
In [2]: import pandas as pd
```

- 1) Create a series of three different colors

```
In [8]: colors = pd.Series(["Red", "Blue", "Green"])
```

```
In [10]: # 2  
colors
```

```
Out[10]: 0      Red  
         1      Blue  
         2     Green  
        dtype: object
```

- 3) Create a Series of three different car types and view it

```
In [6]: cars = pd.Series(["Sedan", "SUV", "Hatchback"])
```

```
In [7]: cars
```

```
Out[7]: 0      Sedan  
         1      SUV  
         2   Hatchback  
        dtype: object
```

- 4) Combine the Series of cars and colors into a Data Frame

```
In [11]: car_df = pd.DataFrame({"Car_Type": cars, "Color": colors})
```

```
In [12]: car_df
```

```
Out[12]:   Car_Type  Color  
0      Sedan    Red  
1       SUV    Blue  
2  Hatchback  Green
```

- 5) Find the different datatypes of the car DataFrame

```
In [13]: car_df.dtypes
```

```
Out[13]: Car_Type    object  
          Color      object  
        dtype: object
```

- 6) Describe your current car sales DataFrame using describe()

```
In [14]: car_df.describe()
```

Out[14]:

	Car_Type	Color
count	3	3
unique	3	3
top	Sedan	Red
freq	1	1

7) Get information about your DataFrame using info()

```
In [15]: car_df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 2 columns):
 #   Column     Non-Null Count  Dtype  
--- 
 0   Car_Type    3 non-null      object 
 1   Color       3 non-null      object 
dtypes: object(2)
memory usage: 180.0+ bytes
```

8) Create a Series of different numbers and find the mean of them

```
In [16]: numbers = pd.Series([10, 20, 30, 40, 50])
```

```
In [17]: numbers.mean()
```

Out[17]: np.float64(30.0)

9) Create a Series of different numbers and find the sum

```
In [18]: numbers.sum()
```

Out[18]: np.int64(150)

10) List out all the column names of the car sales Data Frame

```
In [19]: car_df.columns
```

Out[19]: Index(['Car_Type', 'Color'], dtype='object')

11) Find the length of the car sales DataFrame

```
In [20]: len(car_df)
```

Out[20]: 3

12) Show the first 5 rows of the DataFrame

```
In [21]: car_df.head()
```

```
Out[21]:   Car_Type  Color
```

	Car_Type	Color
0	Sedan	Red
1	SUV	Blue
2	Hatchback	Green

13) Show the first 7 rows of the DataFrame

```
In [22]: car_df.head(7)
```

```
Out[22]:   Car_Type  Color
```

	Car_Type	Color
0	Sedan	Red
1	SUV	Blue
2	Hatchback	Green

14) Show the bottom 5 rows of the DataFrame

```
In [23]: car_df.tail()
```

```
Out[23]:   Car_Type  Color
```

	Car_Type	Color
0	Sedan	Red
1	SUV	Blue
2	Hatchback	Green

15) Use .loc to select the row at index 3 of the car sales Data Frame

```
In [ ]: car_df.loc[3]
```

16) Use .iloc to select the row at position 3

```
In [ ]: car_df.iloc[3]
```

17) Create a crosstab of the Make and Doors columns

```
In [25]: sales_df = pd.DataFrame({"Make": ["Toyota", "BMW", "Toyota", "Honda", "BMW"], "Doors": [4, 4, 2, 4, 4]})
```

```
In [26]: pd.crosstab(sales_df["Make"], sales_df["Doors"])
```

```
Out[26]:  Doors  2  4
```

Make		
BMW	1	1
Honda	0	1
Toyota	1	1

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

```
thank_you = pd.DataFrame({
    "🐼 Pandas Assignment": ["COMPLETED"],
    "📊 Status": ["SUCCESS"],
    "🙏 Message": [ "THANK YOU"]
})

print("\n" + "="*35)
print("      🐾 THANK YOU 🐾      ")
print("="*35)
print(thank_you)
print("\nAssignment completed successfully!")
```