

Pandas Exercise Assignment No.01

In [1]: `import numpy as np`

In [2]: `import pandas as pd`

In [3]: `cars=pd.read_csv('https://github.com/YBI-Foundation/Dataset/raw/main/MPG.csv')`

In [4]: `cars`

Out[4]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	name
0	18.0	8	307.0	130.0	3504	12.0	70	usa	chevrolet chevelle malibu
1	15.0	8	350.0	165.0	3693	11.5	70	usa	buick skylark 320
2	18.0	8	318.0	150.0	3436	11.0	70	usa	plymouth satellite
3	16.0	8	304.0	150.0	3433	12.0	70	usa	amc rebel s/s
4	17.0	8	302.0	140.0	3449	10.5	70	usa	ford torino
...
393	27.0	4	140.0	86.0	2790	15.6	82	usa	ford mustang
394	44.0	4	97.0	52.0	2130	24.6	82	europa	vw pickup
395	32.0	4	135.0	84.0	2295	11.6	82	usa	dodge rampage
396	28.0	4	120.0	79.0	2625	18.6	82	usa	ford range
397	31.0	4	119.0	82.0	2720	19.4	82	usa	chevy s-10

398 rows × 9 columns



```
In [5]: cars.head(10)
```

```
Out[5]:
```

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	name
0	18.0	8	307.0	130.0	3504	12.0	70	usa	chevrolet chevelle malibu
1	15.0	8	350.0	165.0	3693	11.5	70	usa	buick skylark 320
2	18.0	8	318.0	150.0	3436	11.0	70	usa	plymouth satellite
3	16.0	8	304.0	150.0	3433	12.0	70	usa	amc rebel ss
4	17.0	8	302.0	140.0	3449	10.5	70	usa	ford torino
5	15.0	8	429.0	198.0	4341	10.0	70	usa	ford galaxie 500
6	14.0	8	454.0	220.0	4354	9.0	70	usa	chevrolet impala
7	14.0	8	440.0	215.0	4312	8.5	70	usa	plymouth fury ii
8	14.0	8	455.0	225.0	4425	10.0	70	usa	pontiac catalina
9	15.0	8	390.0	190.0	3850	8.5	70	usa	amc ambassador dp



Display Information Of Dataframe

In [6]: `cars.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 398 entries, 0 to 397
Data columns (total 9 columns):
#   Column          Non-Null Count  Dtype
---  -
0   mpg             398 non-null    float64
1   cylinders       398 non-null    int64
2   displacement    398 non-null    float64
3   horsepower      392 non-null    float64
4   weight          398 non-null    int64
5   acceleration    398 non-null    float64
6   model_year      398 non-null    int64
7   origin          398 non-null    object
8   name            398 non-null    object
dtypes: float64(4), int64(3), object(2)
memory usage: 28.1+ KB
```

Displaying the summary statistics of the dataframe

In [8]: `cars.describe()`

Out[8]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year
count	398.000000	398.000000	398.000000	392.000000	398.000000	398.000000	398.000000
mean	23.514573	5.454774	193.425879	104.469388	2970.424623	15.568090	76.010050
std	7.815984	1.701004	104.269838	38.491160	846.841774	2.757689	3.697627
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000
25%	17.500000	4.000000	104.250000	75.000000	2223.750000	13.825000	73.000000
50%	23.000000	4.000000	148.500000	93.500000	2803.500000	15.500000	76.000000
75%	29.000000	8.000000	262.000000	126.000000	3608.000000	17.175000	79.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000

Display Summary of all column

In [9]: `cars.describe(include="all")`

Out[9]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year
count	398.000000	398.000000	398.000000	392.000000	398.000000	398.000000	398.000000
unique	NaN	NaN	NaN	NaN	NaN	NaN	NaN
top	NaN	NaN	NaN	NaN	NaN	NaN	NaN
freq	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mean	23.514573	5.454774	193.425879	104.469388	2970.424623	15.568090	76.010050
std	7.815984	1.701004	104.269838	38.491160	846.841774	2.757689	3.697627
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000
25%	17.500000	4.000000	104.250000	75.000000	2223.750000	13.825000	73.000000
50%	23.000000	4.000000	148.500000	93.500000	2803.500000	15.500000	76.000000
75%	29.000000	8.000000	262.000000	126.000000	3608.000000	17.175000	79.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000

Displaying of the Matrix

In [10]: `cars.corr()`

Out[10]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year
mpg	1.000000	-0.775396	-0.804203	-0.778427	-0.831741	0.420289	0.579267
cylinders	-0.775396	1.000000	0.950721	0.842983	0.896017	-0.505419	-0.348746
displacement	-0.804203	0.950721	1.000000	0.897257	0.932824	-0.543684	-0.370164
horsepower	-0.778427	0.842983	0.897257	1.000000	0.864538	-0.689196	-0.416361
weight	-0.831741	0.896017	0.932824	0.864538	1.000000	-0.417457	-0.306564
acceleration	0.420289	-0.505419	-0.543684	-0.689196	-0.417457	1.000000	0.288137
model_year	0.579267	-0.348746	-0.370164	-0.416361	-0.306564	0.288137	1.000000

In [12]: `cars.shape`

Out[12]: (398, 9)

```
In [14]: cars.columns
```

```
Out[14]: Index(['mpg', 'cylinders', 'displacement', 'horsepower', 'weight',  
              'acceleration', 'model_year', 'origin', 'name'],  
              dtype='object')
```

Displaying of unique valus in DataFrame

```
In [17]: cars.nunique()
```

```
Out[17]: mpg          129  
cylinders           5  
displacement       82  
horsepower         93  
weight            351  
acceleration       95  
model_year         13  
origin              3  
name              305  
dtype: int64
```

```
In [20]: cars['origin'].value_counts()
```

```
Out[20]: usa          249  
japan           79  
europe          70  
Name: origin, dtype: int64
```

Displaying of missing values

```
In [22]: cars.isna().sum()
```

```
Out[22]: mpg          0  
cylinders           0  
displacement       0  
horsepower         6  
weight            0  
acceleration       0  
model_year         0  
origin             0  
name              0  
dtype: int64
```

Display random sample of three rows

```
In [23]: cars.sample()
```

```
Out[23]:
```

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	name
99	18.0	6	232.0	100.0	2945	16.0	73	usa	amc hornet

```
In [25]: cars.displacement[13]
```

```
Out[25]: 455.0
```

Display first ten rows of columns second and third using loc.function

```
In [27]: cars.loc[0:9,["cylinders", 'displacement']]
```

```
Out[27]:
```

	cylinders	displacement
0	8	307.0
1	8	350.0
2	8	318.0
3	8	304.0
4	8	302.0
5	8	429.0
6	8	454.0
7	8	440.0
8	8	455.0
9	8	390.0

```
In [28]: cars.iloc[-11:-1,[1,2]]
```

```
Out[28]:
```

	cylinders	displacement
387	6	262.0
388	4	156.0
389	6	232.0
390	4	144.0
391	4	135.0
392	4	151.0
393	4	140.0
394	4	97.0
395	4	135.0
396	4	120.0

sub_sample

```
In [30]: sub_sample=cars.iloc[:,[1,2]]
```

```
In [31]: sub_sample
```

```
Out[31]:
```

	cylinders	displacement
0	8	307.0
1	8	350.0
2	8	318.0
3	8	304.0
4	8	302.0
...
393	4	140.0
394	4	97.0
395	4	135.0
396	4	120.0
397	4	119.0

398 rows × 2 columns

```
In [ ]:
```

