In [ ]:

```
Python for data analysis Lab 5
Nurshanov Dias
IT3-2208
```

In [11]:

```python
import pandas as pd
#1 exercise Load auto-mpg.data into a DataFrame autodf.

def proccess_bad_lines(lines):
    """
    function which deletes extra element in bad lines
    """
    return [line for line in lines if line.strip() and not line.strip() == '']


autodf = pd.read_csv('auto-mpg.csv', sep=',', on_bad_lines=proccess_bad_lines, engine='python')

#1) replace '?' on nan
#2) replace nan on mean
autodf['horsepower'] = pd.to_numeric(autodf['horsepower'], errors='coerce')
horsepower_mean = autodf['horsepower'].mean().round()
autodf['horsepower'] = autodf['horsepower'].fillna(horsepower_mean)
print(autodf.head())
```

```
   %mpg  cylinders  displacement  horsepower  weight  acceleration  \
0  18.0          8         307.0       130.0  3504.0          12.0   
1  15.0          8         350.0       165.0  3693.0          11.5   
2  18.0          8         318.0       150.0  3436.0          11.0   
3  16.0          8         304.0       150.0  3433.0          12.0   
4  17.0          8         302.0       140.0  3449.0          10.5   

   model year  origin                   car name  
0          70       1  chevrolet chevelle malibu  
1          70       1          buick skylark 320  
2          70       1         plymouth satellite  
3          70       1              amc rebel sst  
4          70       1                ford torino  
```

In [13]:

```python
#2 exercise Give description of the generated DataFrame autodf. (Description-> Main Statistics) .describe() OR a
print('Give description of the generated DataFrame autodf. (Description-> Main Statistics) .describe() OR aggreg
# print(autodf.describe())

#3 exercise Display the first 10 rows of the DataFrame autodf. head(10)
print('Display the first 10 rows of the DataFrame autodf. head(10)')
# print(autodf.head(10))


'''4 exercise
Find the attributes which have missing values. Handle the missing values using following two ways:
i. Replace the missing values by a value before that. method = 'pad'
ii. Remove the rows having missing values from the original dataset dropna()
'''

autodf.fillna(method='pad', inplace=True)

autodf.dropna(inplace=True)

print('{4} exercise')
print(autodf.info())



# 5 Print the details of the car which gave the maximum mileage.

max_mileage_car = autodf[autodf['%mpg'] == autodf['%mpg'].max()]
print('{5} Details of the car with the maximum mileage:')
print(max_mileage_car)

# 6 Find the average displacement of the car given the number of cylinders. mean



average_displacement = autodf.groupby('cylinders')['displacement'].mean()
print(6)
print(average_displacement)
# 7 What is the average number of cylinders in a car? [cylinder] -> mean
average_cylinders = autodf['cylinders'].mean()
print(7)
print(average_cylinders)

# 8 Determine the no. of cars with weight greater than the average weight. count
average_weight = autodf['weight'].mean()
cars_above_average_weight = autodf[autodf['weight'] > average_weight].count()
```

```python
print('8')
print(cars_above_average_weight)
```

```
Give description of the generated DataFrame autodf. (Description-> Main Statistics) .describe() OR aggregate()
Display the first 10 rows of the DataFrame autodf. head(10)
{4} exercise
<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    398 non-null    float64
 4   weight        398 non-null    float64
 5   acceleration  398 non-null    float64
 6   model year    398 non-null    int64  
 7   origin        398 non-null    int64  
 8   car name      398 non-null    object 
dtypes: float64(5), int64(3), object(1)
memory usage: 28.1+ KB
None
{5} Details of the car with the maximum mileage:
     %mpg  cylinders  displacement  horsepower  weight  acceleration  \
322  46.6          4          86.0        65.0  2110.0          17.9   

     model year  origin   car name  
322          80       3  mazda glc  
6
cylinders
3     72.500000
4    109.796569
5    145.000000
6    218.142857
8    345.009709
Name: displacement, dtype: float64
7
5.454773869346734
8
%mpg            171
cylinders       171
displacement    171
horsepower      171
weight          171
acceleration    171
model year      171
origin          171
car name        171
dtype: int64
```

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js