

```
In [1]: import sys
import os
import pandas as pd
import numpy as np

In [2]: cars_data=pd.read_csv('Toyota.csv')

In [3]: cars_data=cars_data.copy(deep=True)

In [4]: cars_data.index
cars_data.columns
cars_data.shape
cars_data.size
cars_data.head(5)
cars_data.tail(5)
cars_data.iat[5,6]

Out[4]: 0.0

In [5]: cars_data.loc[:, 'FuelType']

Out[5]: 0      Diesel
1      Diesel
2      Diesel
3      Diesel
4      Diesel
...
1431     Petrol
1432     Petrol
1433     Petrol
1434      NaN
1435     Petrol
Name: FuelType, Length: 1436, dtype: object

In [6]: #16
print(np.unique(cars_data['KM']))

['1' '10000' '100123' ... '99865' '99971' '??']

In [7]: print(np.unique(cars_data['HP']))

['107' '110' '116' '192' '69' '71' '72' '73' '86' '90' '97' '98' '????']

In [8]: print(np.unique(cars_data['Doors']))

['2' '3' '4' '5' 'five' 'four' 'three']
```

pandas III

```
In [9]: cars_data=pd.read_csv('Toyota.csv',index_col=0,na_values=["??","????"])

In [10]: cars_data=cars_data.copy(deep=True)

In [11]: cars_data['Automatic']=cars_data['Automatic'].astype('object')

In [12]: cars_data['MetColor']=cars_data['MetColor'].astype('object')

In [13]: cars_data['FuelType'].nbytes

Out[13]: 11488

In [14]: cars_data['FuelType'].astype('category').nbytes

Out[14]: 1460

In [15]: cars_data.dtypes

Out[15]: Price      int64
Age      float64
KM      float64
FuelType   object
HP      float64
MetColor   object
Automatic   object
CC       int64
Doors      object
Weight    int64
dtype: object

In [16]: cars_data.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 1436 entries, 0 to 1435
Data columns (total 10 columns):
Price      1436 non-null int64
Age        1336 non-null float64
KM         1421 non-null float64
FuelType   1336 non-null object
HP         1430 non-null float64
MetColor   1286 non-null float64
Automatic   1436 non-null int64
CC         1436 non-null int64
Doors      1436 non-null object
Weight     1436 non-null int64
dtypes: float64(4), int64(4), object(2)
memory usage: 123.4+ KB

In [17]: print(np.unique(cars_data['Doors']))

['2' '3' '4' '5' 'five' 'four' 'three']

In [18]: cars_data['Doors'].replace('three',3,inplace=True)
cars_data['Doors'].replace('four',4,inplace=True)
cars_data['Doors'].replace('five',5,inplace=True)

In [19]: cars_data['Doors']=cars_data['Doors'].astype('int64')

In [20]: cars_data.isnull().sum()

Out[20]: Price      0
Age      100
KM       15
FuelType  100
HP        6
MetColor  150
Automatic   0
CC         0
Doors      0
Weight     0
dtype: int64
```

control structures#if else for while

```
In [21]: cars_data.insert(10,"Price_Class","")

In [22]: for i in range(0,len(cars_data['Price']),1):
if(cars_data['Price'][i]<=8450):
cars_data['Price_Class'][i]="low"
elif((cars_data['Price'][i]>11950)):
cars_data['Price_Class'][i]="high"
else:cars_data['Price_Class'][i]="medium"

C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
"""
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
This is separate from the ipykernel package so we can avoid doing imports until
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

In [23]: i=0
while i<len(cars_data['Price']):
if(cars_data['Price'][i]<=8450):
cars_data['Price_Class'][i]="low"
elif((cars_data['Price'][i]>11950)):
cars_data['Price_Class'][i]="high"
else:cars_data['Price_Class'][i]="medium"
i=i+1

C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
after removing the cwd from sys.path.
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
import sys

In [24]: cars_data['Price_Class'].value_counts()

Out[24]: medium    751
low             369
high            316
Name: Price_Class, dtype: int64

In [25]: cars_data['Price_Class'].value_counts()

Out[25]: medium    751
low             369
high            316
Name: Price_Class, dtype: int64

In [26]: cars_data.insert(11,"age_converted",0)

In [27]: def c_convert(val):
val_con=val/12
return val_con

In [28]: cars_data["age_converted"]=c_convert(cars_data['Age'])
cars_data["age_converted"]=round(cars_data["age_converted"],1)

In [29]: cars_data.head()

Out[29]:
```

	Price	Age	KM	FuelType	HP	MetColor	Automatic	CC	Doors	Weight	Price_Class	age_converted
0	13500	23.0	46986.0	Diesel	90.0	1	0	2000	3	1165	high	1.9
1	13750	23.0	72937.0	Diesel	90.0	1	0	2000	3	1165	high	1.9
2	13950	24.0	41711.0	Diesel	90.0	NaN	0	2000	3	1165	high	2.0
3	14950	26.0	48000.0	Diesel	90.0	0	0	2000	3	1165	high	2.2
4	13750	30.0	38500.0	Diesel	90.0	0	0	2000	3	1170	high	2.5

```
In [30]: cars_data.insert(12,"kmpermonth",0)

In [31]: def c_convert(val1,val2):
val_converted=val1/12
ratio=val2/val1
return [val_converted,ratio]

In [32]: cars_data["age_converted"],cars_data["kmpermonth"]= \
c_convert(cars_data['Age'],cars_data['KM'])

In [33]: cars_data.head()

Out[33]:
```

	Price	Age	KM	FuelType	HP	MetColor	Automatic	CC	Doors	Weight	Price_Class	age_converted	kmpermon
0	13500	23.0	46986.0	Diesel	90.0	1	0	2000	3	1165	high	1.916667	2042.869565
1	13750	23.0	72937.0	Diesel	90.0	1	0	2000	3	1165	high	1.916667	3171.173913
2	13950	24.0	41711.0	Diesel	90.0	NaN	0	2000	3	1165	high	2.000000	1737.958333
3	14950	26.0	48000.0	Diesel	90.0	0	0	2000	3	1165	high	2.166667	1846.153846
4	13750	30.0	38500.0	Diesel	90.0	0	0	2000	3	1170	high	2.500000	1283.333333