```
import pandas as pd
```

```
headers=['symboling','normalized-losses','make','fuel-type','aspiration','num-of-doors','body-style','drive-wheels',
         'engine-location','wheel-base','length','width','height','curb-weight','engine-type',
         'num-of-cylinders','engine-size','fuel-system','bore','stroke','compression-ratio','horsepower','peak-rpm','city-mpg','highway-mp
ds='/content/imports-85.data'
df=pd.read_csv(ds,sep=",",names=headers)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine locatic
0	3	?	alfa-romero	gas	std	two	convertible	rwd	fro
1	3	?	alfa-romero	gas	std	two	convertible	rwd	fro
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	fro
3	2	164	audi	gas	std	four	sedan	fwd	fro
4	2	164	audi	gas	std	four	sedan	4wd	fro
...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	fro
201	-1	95	volvo	gas	turbo	four	sedan	rwd	fro
202	-1	95	volvo	gas	std	four	sedan	rwd	fro
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	fro
204	-1	95	volvo	gas	turbo	four	sedan	rwd	fro

205 rows × 26 columns

```
df.head()
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	whee ba
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	8
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	8
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	9
3	2	164	audi	gas	std	four	sedan	fwd	front	9
4	2	164	audi	gas	std	four	sedan	4wd	front	9

5 rows × 26 columns

```
import numpy as np
df.replace("?",np.nan,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	NaN	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	NaN	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	NaN	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1

```
missing_data=df.isnull()
missing_data.head()
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base	.
0	False	True	False	False	False	False	False	False	False	False	
1	False	True	False	False	False	False	False	False	False	False	
2	False	True	False	False	False	False	False	False	False	False	
3	False	False	False	False	False	False	False	False	False	False	
4	False	False	False	False	False	False	False	False	False	False	

5 rows × 26 columns

```
missing_data=df.isnull()
missing_data.tail()
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
200	False	False	False	False	False	False	False	False	False	False
201	False	False	False	False	False	False	False	False	False	False
202	False	False	False	False	False	False	False	False	False	False
203	False	False	False	False	False	False	False	False	False	False
204	False	False	False	False	False	False	False	False	False	False

5 rows × 26 columns

```
missing_data=df.isnull()
missing_data
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base
0	False	True	False	False	False	False	False	False	False	False
1	False	True	False	False	False	False	False	False	False	False
2	False	True	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...
200	False	False	False	False	False	False	False	False	False	False
201	False	False	False	False	False	False	False	False	False	False
202	False	False	False	False	False	False	False	False	False	False

```
for i in missing_data.columns.values.tolist():
  print(i)
  print(missing_data[i].value_counts())
  print(" ")
```

```
symboling
False    205
Name: symboling, dtype: int64

normalized-losses
False    164
True      41
Name: normalized-losses, dtype: int64

make
False    205
Name: make, dtype: int64

fuel-type
False    205
Name: fuel-type, dtype: int64

aspiration
False    205
Name: aspiration, dtype: int64

num-of-doors
False    203
True       2
Name: num-of-doors, dtype: int64

body-style
False    205
Name: body-style, dtype: int64

drive-wheels
False    205
Name: drive-wheels, dtype: int64

engine-location
False    205
Name: engine-location, dtype: int64

wheel-base
False    205
Name: wheel-base, dtype: int64

length
False    205
Name: length, dtype: int64

width
False    205
Name: width, dtype: int64

height
False    205
Name: height, dtype: int64

curb-weight
False    205
Name: curb-weight, dtype: int64
```

```
#Replace with Mean
avg_norm_losses=df["normalized-losses"].astype("float").mean(axis=0)
df["normalized-losses"].replace(np.nan,avg_norm_losses,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 26 columns

```
#Replace with Median
avg_norm_losses=df["normalized-losses"].astype("float").median
df["normalized-losses"].replace(np.nan,avg_norm_losses,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 26 columns

```
#Replace with Mean
avg_norm_losses=df["bore"].astype("float").mean(axis=0)
df["bore"].replace(np.nan,avg_norm_losses,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 26 columns

```
#Replace with Mean
avg_norm_losses=df["peak-rpm"].astype("float").mean(axis=0)
df["peak-rpm"].replace(np.nan,avg_norm_losses,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 26 columns

```
#Replace with Mean
avg_norm_losses=df["stroke"].astype("float").mean(axis=0)
df["stroke"].replace(np.nan,avg_norm_losses,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 26 columns

```
#Replace with Mean
avg_norm_losses=df["horsepower"].astype("float").mean(axis=0)
df["horsepower"].replace(np.nan,avg_norm_losses,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 26 columns

```
#Replace with Mean
avg_norm_losses=df["price"].astype("float").mean(axis=0)
df["price"].replace(np.nan,avg_norm_losses,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 26 columns

```
df['num-of-doors'].value_counts()
```

```
four    114
two      89
Name: num-of-doors, dtype: int64
```

```
df["num-of-doors"].value_counts().idxmax()
```

```
'four'
```

```
df['num-of-doors'].replace(np.nan,"Four",inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	

```
df.dtypes
```

```
symboling             int64
normalized-losses    object
make                 object
fuel-type            object
aspiration           object
num-of-doors         object
body-style           object
drive-wheels         object
engine-location      object
wheel-base          float64
length              float64
width               float64
height              float64
curb-weight           int64
engine-type          object
num-of-cylinders     object
engine-size           int64
fuel-system          object
bore                 object
stroke               object
compression-ratio   float64
horsepower           object
peak-rpm             object
city-mpg              int64
highway-mpg           int64
price                object
dtype: object
```

```
df.dropna(subset=["price"],axis=0,inplace=True)
df.reset_index(drop=True,inplace=True)
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 26 columns

```
df["city-L/100km"]=235/df["city-mpg"]
df.head()
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	whee ba
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	8
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	8
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	9
3	2	164	audi	gas	std	four	sedan	fwd	front	9
4	2	164	audi	gas	std	four	sedan	4wd	front	9

5 rows × 27 columns

```
df["highway-L/100km"]=235/df["highway-mpg"]
df.head()
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	whee ba
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	8
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	8
2	1	122.0	alfa-romero	gas	std	two	hatchback	rwd	front	9
3	2	164	audi	gas	std	four	sedan	fwd	front	9
4	2	164	audi	gas	std	four	sedan	4wd	front	9

5 rows × 28 columns

```
df['length']=df['length']/df['length'].max()
df['width']=df['width']/df['width'].max()
df['height']=df['height']/df['height'].max()
```

```
df
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wh
0	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
1	3	122.0	alfa-romero	gas	std	two	convertible	rwd	front	
3	2	164	audi	gas	std	four	sedan	fwd	front	
4	2	164	audi	gas	std	four	sedan	4wd	front	
...	...	...	...	...	...	...	...	...	...	
200	-1	95	volvo	gas	std	four	sedan	rwd	front	1
201	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1
202	-1	95	volvo	gas	std	four	sedan	rwd	front	1
203	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	1
204	-1	95	volvo	gas	turbo	four	sedan	rwd	front	1

205 rows × 28 columns