WILEY

Data Engineering

Module: Python programming

Topic: Working with pandas Dataframe

Scenario

- Txt/CSV/JSON files in python can be loaded using pandas.read_csv/txt/json.
- Using pandas, we can extract valuable information
- Using pandas we can Easily handles missing data
- Pandas provides an efficient way to slice the data
- Pandas provides a flexible way to merge, concatenate or reshape the data
- pandas includes a powerful time series tool to work with

Background

Objective

After the completing this exercise, the learner will be able to –

- Understand how to show head and tail of the dataset using pandas
- Understand how to check null value in each column
- Extract valuable information using pandas
- Understand how to use count, max, min, average function in pandas
- Understand how to sort and group column data
- Understand how to merge two dataset using pandas

Problem statement

- From the given dataset print the first and last five rows
- Check sum of total null value in every column in dataset
- Find the most expensive car company name
- Print All Toyota Cars details
- Count total cars per company
- Find each company's Higesht price car
- Find the average mileage of each car making company
- Sort all cars by horsepower column
- Concatenate two data frames using the following conditions

Dataset information

In this exercise, we are using Automobile Dataset for data analysis. This Dataset has different characteristics of an auto such as body-style, wheel-base, engine-type, price, mileage, horsepower, etc.

Execution

Step 1

From the given dataset print the first and last five rows

Print first five rows

```
1. import pandas as pd
2. df = pd.read_csv("Automobile_data.csv")
3. df.head(5)
```

	index	company	body-style	wheel-base	length	engine-type	num-of-cylinders	horsepower	average-mileage	price
0	0	alfa-romero	convertible	88.6	168.8	dohc	four	111	21	13495.0
1	1	alfa-romero	convertible	88.6	168.8	dohc	four	111	21	16500.0
2	2	alfa-romero	hatchback	94.5	171.2	ohcv	six	154	19	16500.0
3	3	audi	sedan	99.8	176.6	ohc	four	102	24	13950.0
4	4	audi	sedan	99.4	176.6	ohc	five	115	18	17450.0

Print last five rows

```
1. import pandas as pd
2. df = pd.read_csv("Automobile_data.csv")
3. df.tail(5)
```

	index	company	body-style	wheel-base	length	engine-type	num-of-cylinders	horsepower	average-mileage	price
56	81	volkswagen	sedan	97.3	171.7	ohc	four	85	27	7975.0
57	82	volkswagen	sedan	97.3	171.7	ohc	four	52	37	7995.0
58	86	volkswagen	sedan	97.3	171.7	ohc	four	100	26	9995.0
59	87	volvo	sedan	104.3	188.8	ohc	four	114	23	12940.0
60	88	volvo	wagon	104.3	188.8	ohc	four	114	23	13415.0

Step 2

Check sum of total null value in every column in dataset

```
1. df.isna().sum()
```

```
Unnamed: 0          0
index               0
company             0
body-style          0
wheel-base          0
length              0
engine-type         0
num-of-cylinders    0
horsepower          0
average-mileage     0
price               3
dtype: int64
```

Step 3

Find the most expensive car company name

Print most expensive car's company name and price.

```
1. df = df [['company','price']][df.price==df['price'].max()]
2. df
```

	company	price
35	mercedes-benz	45400.0

Step 4

Print All Toyota Cars details

```
1. df = pd.read_csv("Automobile_data.csv")
2. car_Manufacturers = df.groupby("company")
3. toyotaDf = car_Manufacturers.get_group("toyota")
4. toyotaDf
```

	Unnamed: 0	Unnamed: 0.1	index	company	body-style	wheel-base	length	engine-type	num-of-cylinders	horsepower	average-mileage	price
48	48	48	66	toyota	hatchback	95.7	158.7	ohc	four	62	35	5348.0
49	49	49	67	toyota	hatchback	95.7	158.7	ohc	four	62	31	6338.0
50	50	50	68	toyota	hatchback	95.7	158.7	ohc	four	62	31	6488.0
51	51	51	69	toyota	wagon	95.7	169.7	ohc	four	62	31	6918.0
52	52	52	70	toyota	wagon	95.7	169.7	ohc	four	62	27	7898.0
53	53	53	71	toyota	wagon	95.7	169.7	ohc	four	62	27	8778.0
54	54	54	79	toyota	wagon	104.5	187.8	dohc	six	156	19	15750.0

Step 5

Count total cars per company

```
1. df = pd.read_csv("Automobile_data.csv")
2. df['company'].value_counts()
```

```
toyota           7
bmw              6
mazda            5
nissan           5
audi             4
mercedes-benz    4
mitsubishi       4
volkswagen       4
alfa-romero      3
chevrolet        3
honda            3
isuzu            3
jaguar           3
porsche          3
dodge            2
volvo            2
Name: company, dtype: int64
```

Step 6

Find each company's Higesht price car

```
1. df = pd.read_csv("Automobile_data.csv")
2. car_Manufacturers = df.groupby('company')
3. priceDf = car_Manufacturers['company','price'].max()
4. priceDf
```

company	company	price
alfa-romero	alfa-romero	16500.0
audi	audi	18920.0
bmw	bmw	41315.0
chevrolet	chevrolet	6575.0
dodge	dodge	6377.0
honda	honda	12945.0
isuzu	isuzu	6785.0
jaguar	jaguar	36000.0
mazda	mazda	18344.0
mercedes-benz	mercedes-benz	45400.0
mitsubishi	mitsubishi	8189.0
nissan	nissan	13499.0
porsche	porsche	37028.0
toyota	toyota	15750.0
volkswagen	volkswagen	9995.0
volvo	volvo	13415.0

Step 7

Find the average mileage of each car making company

```
1. df = pd.read_csv("Automobile_data.csv")
2. car_Manufacturers = df.groupby('company')
3. mileageDf = car_Manufacturers['company','average-mileage'].mean()
4. mileageDf
```

company	average-mileage
alfa-romero	20.333333
audi	20.000000
bmw	19.000000
chevrolet	41.000000
dodge	31.000000
honda	26.333333
isuzu	33.333333
jaguar	14.333333
mazda	28.000000
mercedes-benz	18.000000
mitsubishi	29.500000
nissan	31.400000
porsche	17.000000
toyota	28.714286
volkswagen	31.750000
volvo	23.000000

Step 8

Sort all cars by horsepower column

```
1. carsDf = pd.read_csv("Automobile_data.csv")
2. carsDf = carsDf.sort_values(by=['horsepower'], ascending=False)
3. carsDf.head(5)
```

	Unnamed: 0	Unnamed: 0.1	index	company	body-style	wheel-base	length	engine-type	num-of-cylinders	horsepower	average-mileage	price
47	47	47	63	porsche	hatchback	98.4	175.7	dohcv	eight	288	17	NaN
26	26	26	35	jaguar	sedan	102.0	191.7	ohcv	twelve	262	13	36000.0
46	46	46	62	porsche	convertible	89.5	168.9	ohcf	six	207	17	37028.0
45	45	45	61	porsche	hardtop	89.5	168.9	ohcf	six	207	17	34028.0
34	34	34	46	mercedes-benz	sedan	120.9	208.1	ohcv	eight	184	14	40960.0

Step-9

Merge following two data frames

carPriceDf

	Company	Price
0	Toyota	23845
1	Honda	17995
2	BMV	135925
3	Audi	71400

carsHorsepowerDf

	Company	horsepower
0	Toyota	141
1	Honda	80
2	BMV	182
3	Audi	160

```
1. Car_Price = {'Company': ['Toyota', 'Honda', 'BMV', 'Audi'], 'Price':
   [23845, 17995, 135925 , 71400]}
2. carPriceDf = pd.DataFrame.from_dict(Car_Price)
3.
4. car_Horsepower = {'Company': ['Toyota', 'Honda', 'BMV', 'Audi'],
   'horsepower': [141, 80, 182 , 160]}
5. carsHorsepowerDf = pd.DataFrame.from_dict(car_Horsepower)
6.
7. carsDf = pd.merge(carPriceDf, carsHorsepowerDf, on="Company")
8. carsDf
```

	Company	Price	horsepower
0	Toyota	23845	141
1	Honda	17995	80
2	BMV	135925	182
3	Audi	71400	160

Conclusion

We have learnt

- How to show head and tail of the dataset using pandas
- How to check null value in each column
- Extract valuable information using pandas
- How to use count, max, min, average function in pandas
- How to sort and group column data
- How to merge two dataset using pandas