

KEY_Practice10_Pandas-Intro

July 17, 2019

1 Practice: Intro to Pandas

First we need to import the `pandas` package. Let's give it the same special nickname we used before.

```
[0]: # import the pandas package with the nickname pd

import pandas as pd
```

Now we'll create a DataFrame. Modify this cell to save it to a variable called `car_data`.

```
[0]: # save the DataFrame to a variable
car_data = pd.DataFrame({'acceleration': [12.0,11.5,11.0,12.0,10.5,10.0,9.0,8.
↳5,10.0,8.5,10.0,8.0,9.5,10.0,15.0,15.5,15.5,16.0,14.5,20.5],
'cylinders': [8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 4, 6, 6, 6, 4, 4],
'displacement': [307.0,350.0,318.0,304.0,302.0,429.0,454.0,440.0,455.0,390.
↳0,383.0,340.0,400.0,455.0,113.0,198.0,199.0,200.0,97.0,97.0],
'horsepower': [130.0,165.0,150.0,150.0,140.0,198.0,220.0,215.0,225.0,190.0,170.
↳0,160.0,150.0,225.0,95.0,95.0,97.0,85.0,88.0,46.0],
'model_year': [70,70,70,70,70,70,70,70,70,70,70,70,70,70,70,70,70,70,70,70],
'mpg': [18.0,15.0,18.0,16.0,17.0,15.0,14.0,14.0,14.0,15.0,15.0,14.0,15.0,14.
↳0,24.0,22.0,18.0,21.0,27.0,26.0],
'name': ['chevrolet chevelle malibu','buick skylark 320','plymouth_
↳satellite','amc rebel sst','ford torino','ford galaxie 500','chevrolet_
↳impala','plymouth fury iii','pontiac catalina','amc ambassador dpl','dodge_
↳challenger se','plymouth cuda 340','chevrolet monte carlo','buick estate_
↳wagon (sw)','toyota corona mark ii','plymouth duster','amc hornet','ford_
↳maverick','datsun pl510','volkswagen 1131 deluxe sedan'],
'origin':_
↳['usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa','usa'],
'weight':_
↳[3504,3693,3436,3433,3449,4341,4354,4312,4425,3850,3563,3609,3761,3086,2372,2833,2774,2587,
```

Now explore the `car_data` DataFrame. View the first few rows:

```
[0]: # view the first 5 rows

car_data.head(5)
```

```
[0]: acceleration cylinders ... origin weight
0      12.0          8 ...    usa    3504
1      11.5          8 ...    usa    3693
2      11.0          8 ...    usa    3436
3      12.0          8 ...    usa    3433
4      10.5          8 ...    usa    3449
```

[5 rows x 9 columns]

```
[0]: # view the first 10 rows
```

```
car_data.head(10)
```

```
[0]: acceleration cylinders ... origin weight
0      12.0          8 ...    usa    3504
1      11.5          8 ...    usa    3693
2      11.0          8 ...    usa    3436
3      12.0          8 ...    usa    3433
4      10.5          8 ...    usa    3449
5      10.0          8 ...    usa    4341
6       9.0          8 ...    usa    4354
7       8.5          8 ...    usa    4312
8      10.0          8 ...    usa    4425
9       8.5          8 ...    usa    3850
```

[10 rows x 9 columns]

What does the end of the DataFrame look like? Try viewing the last few rows:

```
[0]: # view the last 5 rows
```

```
car_data.tail(5)
```

```
[0]: acceleration cylinders ... origin weight
15      15.5          6 ...    usa    2833
16      15.5          6 ...    usa    2774
17      16.0          6 ...    usa    2587
18      14.5          4 ...   japan    2130
19      20.5          4 ...  europe    1835
```

[5 rows x 9 columns]

What happens when you call head or tail without putting a number between the parentheses?

```
[0]: # try using head without a number
```

```
car_data.head()
```

```
[0]: acceleration cylinders ... origin weight
0      12.0          8 ...    usa    3504
1      11.5          8 ...    usa    3693
2      11.0          8 ...    usa    3436
```

3	12.0	8	...	usa	3433
4	10.5	8	...	usa	3449

[5 rows x 9 columns]

[0]: *# try using tail without a number*

```
car_data.tail()
```

[0]:

	acceleration	cylinders	...	origin	weight
15	15.5	6	...	usa	2833
16	15.5	6	...	usa	2774
17	16.0	6	...	usa	2587
18	14.5	4	...	japan	2130
19	20.5	4	...	europa	1835

[5 rows x 9 columns]

What do the rows of the DataFrame represent?

Answers: Different types of cars

What measurements or observations do we have about the data?

Answer: acceleration, cylinder count, engine displacement, horsepower, model year, mpg, name, country of origin, and weight

What types of variables are in each of the columns?

- **acceleration:** float
- **cylinders:** int
- **displacement:** float
- **horsepower:** float
- **model_year:** int
- **mpg:** float
- **name:** string
- **origin:** string
- **weight:** int