



Difference between loc() and iloc() in Pandas DataFrame

Difficulty Level : Easy • Last Updated : 08 Aug, 2022



Read

Discuss

Practice

Video

Courses

Pandas library of python is very useful for the manipulation of mathematical data and is widely used in the field of machine learning. It comprises many methods for its proper functioning. [loc\(\)](#) and [iloc\(\)](#) are one of those methods. These are used in slicing data from the [Pandas DataFrame](#). They help in the convenient selection of data from the DataFrame in [Python](#). They are used in filtering the data according to some conditions.

Creating a sample dataframe

The working of both of these methods is explained in the sample dataset of cars.

python3

```
# importing the module
import pandas as pd

# creating a sample dataframe
data = pd.DataFrame({'Brand': ['Maruti', 'Hyundai', 'Tata',
                               'Mahindra', 'Maruti', 'Hyundai',
                               'Renault', 'Tata', 'Maruti'],
                     'Year': [2012, 2014, 2011, 2015, 2012,
                              2016, 2014, 2018, 2019],
                     'Kms Driven': [50000, 30000, 60000,
                                    25000, 10000, 46000,
                                    31000, 15000, 12000],
                     'City': ['Gurgaon', 'Delhi', 'Mumbai',
                              'Delhi', 'Mumbai', 'Delhi',
                              'Delhi', 'Mumbai', 'Delhi']})
```

```
display(data)
```

Output :

	Brand	Year	Kms Driven	City	Mileage
0	Maruti	2012	50000	Gurgaon	28
1	Hyundai	2014	30000	Delhi	27
2	Tata	2011	60000	Mumbai	25
3	Mahindra	2015	25000	Delhi	26
4	Maruti	2012	10000	Mumbai	28
5	Hyundai	2016	46000	Delhi	29
6	Renault	2014	31000	Mumbai	24
7	Tata	2018	15000	Chennai	21
8	Maruti	2019	12000	Ghaziabad	24

Python loc() function

The [loc\(\) function](#) is label based data selecting method which means that we have to pass the name of the row or column which we want to select. This method includes the last element of the range passed in it, unlike `iloc()`. `loc()` can accept the boolean data unlike `iloc()`. Many operations can be performed using the `loc()` method like

Example 1:

Start Your Coding Journey Now!

```
# selecting cars with brand 'Maruti' and Mileage > 25
display(data.loc[(data.Brand == 'Maruti') & amp
                 (data.Mileage & gt
                  25)])
```

Output :

	Brand	Year	Kms Driven	City	Mileage
0	Maruti	2012	50000	Gurgaon	28
4	Maruti	2012	10000	Mumbai	28

Example 2:

Selecting a range of rows from the DataFrame

python3

```
# selecting range of rows from 2 to 5
display(data.loc[2: 5])
```

Output :

	Brand	Year	Kms Driven	City	Mileage
2	Tata	2011	60000	Mumbai	25
3	Mahindra	2015	25000	Delhi	26
4	Maruti	2012	10000	Mumbai	28
5	Hyundai	2016	46000	Delhi	29

Example 3:

Updating the value of any column

python3

Start Your Coding Journey Now!

Output :

	Brand	Year	Kms Driven	City	Mileage
0	Maruti	2012	50000	Gurgaon	22
1	Hyundai	2014	30000	Delhi	22
2	Tata	2011	60000	Mumbai	22
3	Mahindra	2015	25000	Delhi	26
4	Maruti	2012	10000	Mumbai	22
5	Hyundai	2016	46000	Delhi	29
6	Renault	2014	31000	Mumbai	22
7	Tata	2018	15000	Chennai	21
8	Maruti	2019	12000	Ghaziabad	24

Python iloc() function

The [iloc\(\) function](#) is an indexed-based selecting method which means that we have to pass an integer index in the method to select a specific row/column. This method does not include the last element of the range passed in it unlike loc(). iloc() does not accept the boolean data unlike loc(). Operations performed using iloc() are:

Example 1:

Selecting rows using integer indices

python3

```
# selecting 0th, 2th, 4th, and 7th index rows
display(data.iloc[[0, 2, 4, 7]])
```

Output :

	Brand	Year	Kms Driven	City	Mileage
0	Maruti	2012	50000	Gurgaon	22
2	Tata	2011	60000	Mumbai	22

Start Your Coding Journey Now!

Example 2:

Selecting a range of columns and rows simultaneously

python3

```
# selecting rows from 1 to 4 and columns from 2 to 4
display(data.iloc[1: 5, 2: 5])
```

Output :

	Kms Driven	City	Mileage
1	30000	Delhi	22
2	60000	Mumbai	22
3	25000	Delhi	26
4	10000	Mumbai	22

WHY LEARN DATA SCIENCE?

✓ Highly Paid Career ✓ Increasing Roles

Enrol Today



♡ Like 38

Next >

Select Rows & Columns by Name
or Index in Pandas DataFrame
using [], loc & iloc

Start Your Coding Journey Now!

using [], loc & iloc

2. Select any row from a Dataframe using iloc[] and iat[] in Pandas

3. Python | Pandas DataFrame.loc[]

4. Python | Pandas Series.iloc

5. Python | Extracting rows using Pandas .iloc[]

6. Python | Pandas Extracting rows using .loc[]

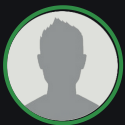
7. Python | Pandas Series.loc

8. Difference Between Spark DataFrame and Pandas DataFrame

9. Convert given Pandas series into a dataframe with its index as another column on the dataframe

10. How to Convert Wide Dataframe to Tidy Dataframe with Pandas stack()?

Article Contributed By :



vipul1501

@vipul1501

Vote for difficulty

Current difficulty : [Easy](#)

Easy

Normal

Medium

Hard

Expert

Start Your Coding Journey Now!

[Improve Article](#)

[Report Issue](#)



 A-143, 9th Floor, Sovereign Corporate Tower,
Sector-136, Noida, Uttar Pradesh - 201305

 feedback@geeksforgeeks.org



Company

[About Us](#)

[Careers](#)

[In Media](#)

[Contact Us](#)

[Privacy Policy](#)

[Copyright Policy](#)

[Advertise with us](#)

Learn

[DSA](#)

[Algorithms](#)

[Data Structures](#)

[SDE Cheat Sheet](#)

[Machine learning](#)

[CS Subjects](#)

[Video Tutorials](#)

[Courses](#)

News

[Top News](#)

[Technology](#)

[Work & Career](#)

[Business](#)

[Finance](#)

[Lifestyle](#)

[Knowledge](#)

Languages

[Python](#)

[Java](#)

[CPP](#)

[Golang](#)

[C#](#)

[SQL](#)

[Kotlin](#)

Start Your Coding Journey Now!

HTML

JavaScript

Bootstrap

ReactJS

NodeJS

Pick Topics to Write

Write Interview Experience

Internships

Video Internship

@geeksforgeeks , Some rights reserved