Python | Pandas dataframe.replace()

Last Updated : 19 Feb, 2021

Python is a great language for doing data analysis, primarily because of the fantastic ecosystem of data-centric python packages. ***Pandas*** is one of those packages and makes importing and analyzing data much easier.

Pandas **`dataframe.replace()`** function is used to replace a string, regex, list, dictionary, series, number etc. from a dataframe. This is a very rich function as it has many variations.
The most powerful thing about this function is that it can work with Python regex (regular expressions).

Syntax: *DataFrame.replace(to_replace=None, value=None, inplace=False, limit=None, regex=False, method='pad', axis=None)*

Parameters:
to_replace : *[str, regex, list, dict, Series, numeric, or None] pattern that we are trying to replace in dataframe.*
value : *Value to use to fill holes (e.g. 0), alternately a dict of values specifying which value to use for each column (columns not in the dict will not be filled). Regular expressions, strings and lists or dic*

this is True.

limit : *Maximum size gap to forward or backward fill*

regex : *Whether to interpret to_replace and/or value as regular ex pressions. If this is True then to_replace must be a string. Otherwi se, to_replace must be None because this parameter will be interp reted as a regular expression or a list, dict, or array of regular expr essions.*

method : *Method to use when for replacement, when to_replace is a list.*

Returns: *filled : NDFrame*

For link to CSV file Used in Code, click here

Example #1: Replace team "Boston Celtics" with "Omega Warrior" in the nba.csv file

```
# importing pandas as pd
import pandas as pd

# Making data frame from the csv file
df = pd.read_csv("nba.csv")

# Printing the first 10 rows of the data frame for visualization
df[:10]
```

Output:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	NaN	5000000.0
5	Amir Johnson	Boston Celtics	90.0	PF	29.0	6-9	240.0	NaN	12000000.0
6	Jordan Mickey	Boston Celtics	55.0	PF	21.0	6-8	235.0	LSU	1170960.0
7	Kelly Olynyk	Boston Celtics	41.0	C	25.0	7-0	238.0	Gonzaga	2165160.0
8	Terry Rozier	Boston Celtics	12.0	PG	22.0	6-2	190.0	Louisville	1824360.0
9	Marcus Smart	Boston Celtics	36.0	PG	22.0	6-4	220.0	Oklahoma State	3431040.0

We are going to replace team “Boston Celtics” with “Omega Warrior” in the ‘df’ data frame

```
# this will replace "Boston Celtics" with "Omega Warrior"
df.replace(to_replace ="Boston Celtics",
                 value ="Omega Warrior")
```

Output:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Omega Warrior	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Omega Warrior	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Omega Warrior	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Omega Warrior	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Omega Warrior	8.0	PF	29.0	6-10	231.0	NaN	5000000.0
5	Amir Johnson	Omega Warrior	90.0	PF	29.0	6-9	240.0	NaN	12000000.0
6	Jordan Mickey	Omega Warrior	55.0	PF	21.0	6-8	235.0	LSU	1170960.0
7	Kelly Olynyk	Omega Warrior	41.0	C	25.0	7-0	238.0	Gonzaga	2165160.0
8	Terry Rozier	Omega Warrior	12.0	PG	22.0	6-2	190.0	Louisville	1824360.0
9	Marcus Smart	Omega Warrior	36.0	PG	22.0	6-4	220.0	Oklahoma State	3431040.0
10	Jared Sullinger	Omega Warrior	7.0	C	24.0	6-9	260.0	Ohio State	2569260.0
11	Isaiah Thomas	Omega Warrior	4.0	PG	27.0	5-9	185.0	Washington	6912869.0
12	Evan Turner	Omega Warrior	11.0	SG	27.0	6-7	220.0	Ohio State	3425510.0
13	James Young	Omega Warrior	13.0	SG	20.0	6-6	215.0	Kentucky	1749840.0
14	Tyler Zeller	Omega Warrior	44.0	C	26.0	7-0	253.0	North Carolina	2616975.0

Example #2: Replacing more than one value at a time. Using python list as an argument

We are going to replace team "Boston Celtics" and "Texas" with "Omega Warrior" in the 'df' dataframe.

```
# importing pandas as pd
import pandas as pd

# Making data frame from the csv file
df = pd.read_csv("nba.csv")

# this will replace "Boston Celtics" and "Texas" with "Omega Warrior"
df.replace(to_replace =["Boston Celtics", "Texas"],
                            value ="Omega Warrior")
```

Output:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Omega Warrior	0.0	PG	25.0	6-2	180.0	Omega Warrior	7730337.0
1	Jae Crowder	Omega Warrior	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Omega Warrior	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Omega Warrior	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Omega Warrior	8.0	PF	29.0	6-10	231.0	NaN	5000000.0
5	Amir Johnson	Omega Warrior	90.0	PF	29.0	6-9	240.0	NaN	12000000.0
6	Jordan Mickey	Omega Warrior	55.0	PF	21.0	6-8	235.0	LSU	1170960.0
7	Kelly Olynyk	Omega Warrior	41.0	C	25.0	7-0	238.0	Gonzaga	2165160.0
8	Terry Rozier	Omega Warrior	12.0	PG	22.0	6-2	190.0	Louisville	1824360.0
9	Marcus Smart	Omega Warrior	36.0	PG	22.0	6-4	220.0	Oklahoma State	3431040.0
10	Jared Sullinger	Omega Warrior	7.0	C	24.0	6-9	260.0	Ohio State	2569260.0
11	Isaiah Thomas	Omega Warrior	4.0	PG	27.0	5-9	185.0	Washington	6912869.0
12	Evan Turner	Omega Warrior	11.0	SG	27.0	6-7	220.0	Ohio State	3425510.0
13	James Young	Omega Warrior	13.0	SG	20.0	6-6	215.0	Kentucky	1749840.0
14	Tyler Zeller	Omega Warrior	44.0	C	26.0	7-0	253.0	North Carolina	2616975.0
15	Bojan Bogdanovic	Brooklyn Nets	44.0	SG	27.0	6-8	216.0	NaN	3425510.0
16	Markel Brown	Brooklyn Nets	22.0	SG	24.0	6-3	190.0	Oklahoma State	845059.0

Notice the College column in the first row, "Texas" has been replaced with "Omega Warriors"

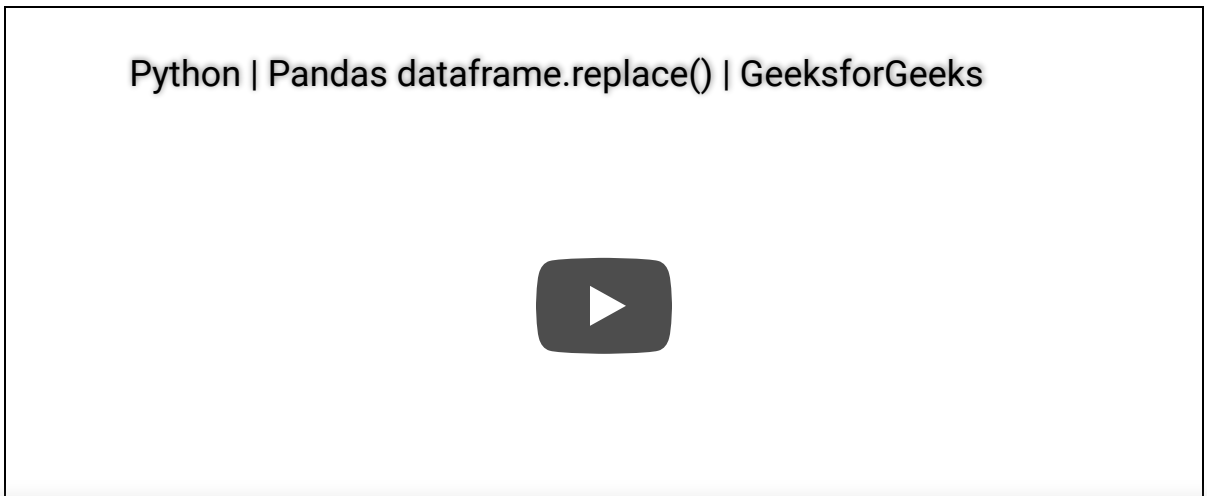
```
# Making data frame from the csv file
df = pd.read_csv("nba.csv")

# will replace  Nan value in dataframe with value -99999
df.replace(to_replace = np.nan, value =-99999)
```

Output:

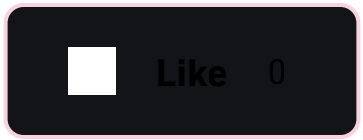
	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	-99999.0
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	-99999	5000000.0
5	Amir Johnson	Boston Celtics	90.0	PF	29.0	6-9	240.0	-99999	12000000.0
6	Jordan Mickey	Boston Celtics	55.0	PF	21.0	6-8	235.0	LSU	1170960.0
7	Kelly Olynyk	Boston Celtics	41.0	C	25.0	7-0	238.0	Gonzaga	2165160.0
8	Terry Rozier	Boston Celtics	12.0	PG	22.0	6-2	190.0	Louisville	1824360.0
9	Marcus Smart	Boston Celtics	36.0	PG	22.0	6-4	220.0	Oklahoma State	3431040.0
10	Jared Sullinger	Boston Celtics	7.0	C	24.0	6-9	260.0	Ohio State	2569260.0
11	Isaiah Thomas	Boston Celtics	4.0	PG	27.0	5-9	185.0	Washington	6912869.0
12	Evan Turner	Boston Celtics	11.0	SG	27.0	6-7	220.0	Ohio State	3425510.0

Notice all the `Nan` value in the data frame has been replaced by -99999. Though for practical purposes we should be careful with what value we are replacing `nan` value.



Programming Foundation Course and learn the basics.

To begin with, your interview preparations Enhance your Data Structures concepts with the **Python DS** Course. And to begin with your Machine Learning Journey, join the **Machine Learning - Basic Level Course**



Previous

Python | Pandas Series.str.replace() to replace text in a series

Next

Python program to find number of days between two given dates

RECOMMENDED ARTICLES

Page : 1 2 3

01 **Python | pandas.to_markdown() in Pandas**
02, Apr 20

02 **Add a Pandas series to another Pandas series**
14, Jul 20

03 **Python | Pandas Index.insert()**
17, Dec 18

04 **Python | Pandas DatetimeIndex.inferred_freq**
24, Dec 18

05 **Python | Pandas PeriodIndex.start_time**
04, Jan 19

06 **Python | Pandas PeriodIndex.week**
04, Jan 19

07 **Python | Pandas Timestamp.second**
08, Jan 19

08 **Python | Pandas Series.asobject**
25, Jan 19

Article Contributed By :

Shubham__Ranjan
@Shubham__Ranjan

Vote for difficulty

Easy | Normal | Medium | Hard | Expert

Article Tags : Python pandas-dataFrame,

Improve Article

5th Floor, A-118,
Sector-136, Noida, Uttar Pradesh - 201305

feedback@geeksforgeeks.org