

Get alerted when assets are down, slow, or vulnerable to SSL attacks—all free for a month. [Learn more ...](#) [Products](#) [Pricing](#) [Docs](#) [Sign in](#) [Sign Up](#)

[Community](#)[Tutorials](#)[Questions](#)[Learning
Paths](#)[Tech
Talks](#)[Get
Involved](#)[Product
Docs](#)[Sign
Up](#)

CONTENTS

1. Pandas DataFrame dropna() Function
2. Pandas Drop All Rows with any Null/NaN/NaT Values
3. Drop All Columns with Any Missing Value
4. Drop Row/Column Only if All the Values are Null
5. DataFrame Drop Rows/Columns when the threshold of null values is crossed
6. Define Labels to look for null values
7. Dropping Rows with NA inplace
8. References

RELATED

Python time.sleep()

[View](#) [↗](#)

Vectors in Python - A Quick Introduction!

[View](#) [↗](#)

// Tutorial //

Pandas dropna() - Drop Null/NA Values from DataFrame

Published on August 3, 2022

[Pandas](#)[Python](#)

By [Pankaj](#)

Developer and author at DigitalOcean.

We use cookies to provide our services and for analytics and marketing. To find out more about our use of cookies, please see our [Privacy Policy](#). By continuing to browse our website, you agree to our use of cookies.

OK

Pandas DataFrame dropna()

1. Pandas DataFrame **dropna()** function is used to remove rows and columns with Null/NaN/NaT values.
2. The **dropna()** function parameters are:
 - a. **axis**: {0 or 'index', 1 or 'columns'}, default 0. If 0, drop rows with null values. If 1, drop columns with missing values.
 - b. **how**: {'any', 'all'}, default 'any'. If 'any', drop the row/column if any of the values is null. If 'all', drop the row/column if all the values are missing.
 - c. **thresh**: an int value to specify the threshold for the drop operation.
 - d. **subset**: specifies the rows/columns to look for null values.
 - e. **inplace**: a boolean value. If True, the source DataFrame is changed and None is returned.

\$ pip install pandas



While we believe that this content benefits our community, we have not yet thoroughly reviewed it. If you have any suggestions for improvements, please let us know by clicking the "report an issue" button at the bottom of the tutorial.

1. Pandas DataFrame dropna() Function

Pandas DataFrame dropna() function is used to remove rows and columns with Null/NaN values. By default, this function returns a new DataFrame and the source DataFrame remains unchanged. We can create null values using None, pandas.NaT, and numpy.nan variables. The dropna() function syntax is:

```
dropna(self, axis=0, how="any", thresh=None, subset=None, inplace=False)
```

- **axis**: possible values are {0 or 'index', 1 or 'columns'}, default 0. If 0, drop rows with null values. If 1, drop columns with missing values.
- **how**: possible values are {'any', 'all'}, default 'any'. If 'any', drop the row/column if any of the values is null. If 'all', drop the row/column if all the values are missing.
- **thresh**: an int value to specify the threshold for the drop operation.
- **subset**: specifies the rows/columns to look for null values.
- **inplace**: a boolean value. If True, the source DataFrame is changed and None is returned.

Let's look at some examples of using dropna() function.

2. Pandas Drop All Rows with any Null/NaN/NaT Values

This is the default behavior of dropna() function.

```
import pandas as pd
import numpy as np

d1 = {'Name': ['Pankaj', 'Meghna', 'David', 'Lisa'], 'ID': [1, 2, 3, 4], 'Salary': [100, 200, np.nan,
    'Role': ['CEO', None, pd.NaT, pd.NaT]}
```

We use cookies to provide our services and for analytics and marketing. To find out more about our use of cookies, please see our [Privacy Policy](#). By continuing to browse our website, you agree to our use of cookies.

OK

Output:

```

      Name  ID  Salary  Role
0  Pankaj   1    100    CEO
1  Meghna   2    200   None
2  David    3     NaN   NaT
3   Lisa    4     NaN   NaT

      Name  ID  Salary  Role
0  Pankaj   1    100    CEO

```

3. Drop All Columns with Any Missing Value

We can pass `axis=1` to drop columns with the missing values.

```
df1 = df.dropna(axis=1)
print(df1)
```

Output:

```

      Name  ID
0  Pankaj   1
1  Meghna   2
2   David   3
3   Lisa    4

```

4. Drop Row/Column Only if All the Values are Null

```

import pandas as pd
import numpy as np

d1 = {'Name': ['Pankaj', 'Meghna', 'David', pd.NaT], 'ID': [1, 2, 3, pd.NaT], 'Salary': [100, 200, np.
      'Role': [np.nan, np.nan, pd.NaT, pd.NaT]}

df = pd.DataFrame(d1)

print(df)

df1 = df.dropna(how='all')
print(df1)

df1 = df.dropna(how='all', axis=1)
print(df1)

```

Output:

```

      Name  ID  Salary  Role
0  Pankaj   1    100   NaT
1  Meghna   2    200   NaT
2  David    3     NaN   NaT
3     NaT  NaT     NaN   NaT

      Name  ID  Salary  Role
0  Pankaj   1    100   NaT
1  Meghna   2    200   NaT
2  David    3     NaN   NaT

```

We use cookies to provide our services and for analytics and marketing. To find out more about our use of cookies, please see our [Privacy Policy](#). By continuing to browse our website, you agree to our use of cookies.

OK

5. DataFrame Drop rows/columns when the threshold of null values is crossed

```
import pandas as pd
import numpy as np

d1 = {'Name': ['Pankaj', 'Meghna', 'David', pd.NaT], 'ID': [1, 2, pd.NaT, pd.NaT], 'Salary': [100, 200, pd.NaT, pd.NaT], 'Role': [np.nan, np.nan, pd.NaT, pd.NaT]}

df = pd.DataFrame(d1)

print(df)

df1 = df.dropna(thresh=2)
print(df1)
```

Output:

```
   Name  ID  Salary  Role
0 Pankaj   1    100   NaT
1 Meghna   2    200   NaT
2 David  NaT    NaN   NaT
3   NaT  NaT    NaT   NaT

   Name ID  Salary  Role
0 Pankaj  1    100   NaT
1 Meghna  2    200   NaT
```

The rows with 2 or more null values are dropped.

6. Define Labels to look for null values

```
import pandas as pd
import numpy as np

d1 = {'Name': ['Pankaj', 'Meghna', 'David', 'Lisa'], 'ID': [1, 2, 3, pd.NaT], 'Salary': [100, 200, np.NaT, np.NaT], 'Role': ['CEO', np.nan, pd.NaT, pd.NaT]}

df = pd.DataFrame(d1)

print(df)

df1 = df.dropna(subset=['ID'])
print(df1)
```

Output:

```
   Name  ID  Salary  Role
0 Pankaj   1    100   CEO
1 Meghna   2    200   NaN
2 David   3    NaN   NaT
3  Lisa  NaT    NaT   NaT

   Name ID  Salary  Role
0 Pankaj  1    100   CEO
1 Meghna  2    200   NaN
2 David   3    NaN   NaT
```

We can specify the index values in the subset when dropping columns from the DataFrame.

We use cookies to provide our services and for analytics and marketing. To find out more about our use of cookies, please see our [Privacy Policy](#). By continuing to browse our website, you agree to our use of cookies.

OK

```
   Name  ID
0  Pankaj  1
1  Meghna  2
2   David  3
3   Lisa  NaT
```

The 'ID' column is not dropped because the missing value is looked only in index 1 and 2.

7. Dropping Rows with NA inplace

We can pass `inplace=True` to change the source DataFrame itself. It's useful when the DataFrame size is huge and we want to save some memory.

```
import pandas as pd

d1 = {'Name': ['Pankaj', 'Meghna'], 'ID': [1, 2], 'Salary': [100, pd.NaT]}

df = pd.DataFrame(d1)

print(df)

df.dropna(inplace=True)
print(df)
```

Output:

```
   Name  ID  Salary
0  Pankaj  1   100.0
1  Meghna  2     NaN

   Name  ID  Salary
0  Pankaj  1   100.0
```

8. References

- [Python Pandas Module Tutorial](#)
- [Pandas Drop Duplicate Rows](#)
- [Pandas DataFrame dropna\(\) API Doc](#)

Want to learn more? Join the DigitalOcean Community!

Join our DigitalOcean community of over a million developers for free! Get help and share knowledge in our Questions & Answers section, find tutorials and tools that will help you grow as a developer and scale your project or business, and subscribe to topics of interest.

Sign up →

We use cookies to provide our services and for analytics and marketing. To find out more about our use of cookies, please see our [Privacy Policy](#). By continuing to browse our website, you agree to our use of cookies.

OK

Still looking for an answer?

Ask a question

Search for more help

Was this helpful?

Yes

No



Comments

[JournalDev](#) • May 12, 2021

Thank u bro, well explained in very simple way

- KHAJA MOINUDDIN KHAN

[JournalDev](#) • February 12, 2021

thats very comprehensive. out of all drop explanation ... this is the best thank you

- johny



This work is licensed under a Creative Commons Attribution-NonCommercial- ShareAlike 4.0 International License.

 Try DigitalOcean for free

Click below to sign up and get **\$100 of credit** to try our products over 60 days!

Sign up →



Popular Topics

Ubuntu

Linux Basics

JavaScript

We use cookies to provide our services and for analytics and marketing. To find out more about our use of cookies, please see our [Privacy Policy](#). By continuing to browse our website, you agree to our use of cookies.

OK

[Databases](#)[Docker](#)[Kubernetes](#)[Ebooks](#)[Browse all topic tags](#)[All tutorials →](#)

Questions

[Q&A](#)[Ask a question](#)[DigitalOcean Product Docs](#)[DigitalOcean Support](#)

Events

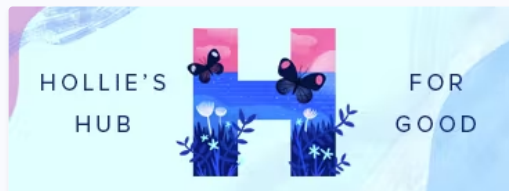
[Tech Talks](#)[Hacktoberfest](#)[Deploy](#)

Get involved

[Community Newsletter](#)[Hollie's Hub for Good](#)[Write for DOnations](#)[Community tools and integrations](#)[Hatch Startup program](#)

GET OUR BIWEEKLY NEWSLETTER

Sign up for Infrastructure as a Newsletter.



HOLLIE'S HUB FOR GOOD

Working on improving health and education, reducing inequality, and spurring economic growth? We'd like to help.



We use cookies to provide our services and for analytics and marketing. To find out more about our use of cookies, please see our [Privacy Policy](#). By continuing to browse our website, you agree to our use of cookies.

OK

Featured on Community

Kubernetes Course

Learn Python 3

Machine Learning in Python

Getting started with Go

Intro to Kubernetes

DigitalOcean Products

Virtual Machines

Managed Databases

Managed Kubernetes

Block Storage

Object Storage

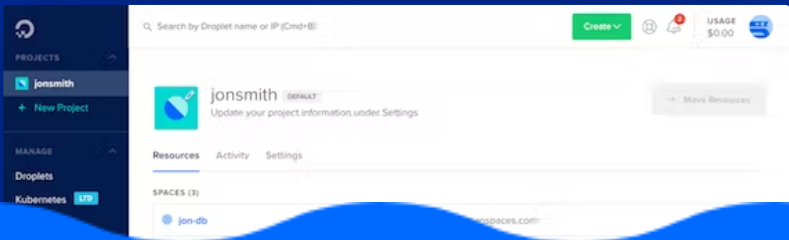
Marketplace

VPC Load Balancers

Welcome to the developer cloud

DigitalOcean makes it simple to launch in the cloud and scale up as you grow – whether you’re running one virtual machine or ten thousand.

Learn More



Company

About

Leadership

Blog

Careers

Customers

Partners

Referral Program

Press

Legal

Trust Platform

Investor Relations

DO Impact

Products

Products Overview

Droplets

Kubernetes

App Platform

Functions

Managed Databases

Spaces

Marketplace

Load Balancers

Block Storage

Tools & Integrations

API

Pricing

Documentation

Release Notes

Uptime

Community

Tutorials

Meetups

Q&A

CSS-Tricks

Write for DONations

Droplets for Demos

Hatch Startup Program

deploy by DigitalOcean

Shop Swag

Research Program

Currents Research

Open Source

Code of Conduct

Newsletter Signup

Solutions

Web & Mobile Apps

Website Hosting

Game Development

Streaming

VPN

Startups

SaaS Solutions

Agency & Web Dev Shops

Managed Cloud Hosting Providers

Big Data

Business Solutions

Cloud Hosting for Blockchain

Contact

Support

Sales

Report Abuse

System Status

Share your ideas



© 2022 DigitalOcean, LLC. All rights reserved.



We use cookies to provide our services and for analytics and marketing. To find out more about our use of cookies, please see our [Privacy Policy](#). By continuing to browse our website, you agree to our use of cookies.

OK