

How can empty strings be replaced with NaN in Pandas?

Authored by
stats writer

June 30, 2024

RECOMMENDED CITATION

stats writer (2024). *How can empty strings be replaced with NaN in Pandas?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=160605>

Pandas, a popular data analysis library in Python, provides a simple and efficient way to replace empty strings with NaN (Not a Number) values. This can be achieved by using the `replace()` function with the parameter `''` (empty string) and specifying `NaN` as the replacement value. This method allows for easy handling of missing or incomplete data in a dataset, which is a common scenario in data analysis. By replacing empty strings with NaN, Pandas allows for easier data manipulation and calculation without the need for complex conditional statements. This feature makes Pandas a useful tool for handling and cleaning data in various data analysis tasks.

Pandas: Replace Empty Strings with NaN

You can use the following syntax to replace empty strings with NaN values in pandas:

```
df = df.replace(r'^s*$', np.nan, regex=True)
```

The following example shows how to use this syntax in practice.

Related:

Example: Replace Empty Strings with NaN

Suppose we have the following pandas DataFrame that contains information about various basketball players:

```
import pandas as pd
```

```
#create DataFrame
```

```
df = pd.DataFrame({'team': ,
```

```
'position': ,  
'points': ,  
'rebounds': })
```

```
#view DataFrame
```

```
df
```

```
team position points rebounds
```

```
0 A 5 11
```

```
1 B G 7 8
```

```
2 G 7 10
```

```
3 D F 9 6
```

```
4 E F 12 6
```

```
5 9 5
```

```
6 G C 9 9
```

```
7 H C 4 12
```

Notice that there are several empty strings in both the team and position columns.

We can use the following syntax to replace these empty strings with NaN values:

```
import numpy as np#replace empty values with NaN  
df = df.replace(r'^s*$', np.nan, regex=True)
```

```
#view updated DataFrame  
df  
team positionpoints rebounds  
0 A NaN 5 11  
1 B G 7 8  
2 NaN G 7 10  
3 D F 9 6  
4 E F 12 6  
5 NaN NaN 9 5  
6 G C 9 9  
7 H C 4 127
```

Notice that each of the empty strings have been replaced with NaN.

Note: You can find the complete documentation for the replace function in pandas .

Additional Resources

The following tutorials explain how to perform other common tasks in pandas: