

How can I replace NaN values with a specific string in a Pandas dataframe?

Authored by
stats writer

July 2, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I replace NaN values with a specific string in a Pandas dataframe?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165683>

The process of replacing NaN (not a number) values with a specific string in a Pandas dataframe involves using the `.fillna()` function with the desired string as the parameter. This will replace all NaN values in the dataframe with the specified string, allowing for easier data manipulation and analysis.

Pandas: Replace NaN Values with String

You can use the following methods to replace NaN values with strings in a pandas DataFrame:

Method 1: Replace NaN Values with String in Entire DataFrame

```
df.fillna("", inplace=True)
```

Method 2: Replace NaN Values with String in Specific Columns

```
df[df.columns] = df[df.columns].fillna("")
```

Method 3: Replace NaN Values with String in One Column

```
df.col1 = df.col1.fillna("")
```

The following examples show how to use each method with the following pandas DataFrame:

```
import pandas as pd
import numpy as np

#create DataFrame with some NaN values
df = pd.DataFrame({'team': ,
'points': ,
'assists': ,
'rebounds': })#view DataFrame
```

```
team points assists rebounds
0 A NaN 5.0 11.0
1 A 11.0 NaN 8.0
2 A 7.0 7.0 10.0
3 A 7.0 9.0 NaN
4 B 8.0 12.0 6.0
5 B 6.0 9.0 5.0
6 B 14.0 9.0 9.0
7 B 15.0 4.0 NaN
```

Method 1: Replace NaN Values with String in Entire DataFrame

The following code shows how to replace every NaN value in an entire DataFrame with an empty string:

```
#replace NaN values in all columns with empty string
```

```
df.fillna("", inplace=True)
```

```
#view updated DataFrame
```

```
df
```

```
team points assists rebounds
```

```
0 A 5.0 11.0
```

```
1 A 11.0 8.0
```

```
2 A 7.0 7.0 10.0
```

```
3 A 7.0 9.0
```

```
4 B 8.0 12.0 6.0
```

```
5 B 6.0 9.0 5.0
```

```
6 B 14.0 9.0 9.0
```

```
7 B 15.0 4.0
```

Notice that every NaN value in each column has been replaced with an empty string.

Method 2: Replace NaN Values with String in Specific Columns

The following code shows how to replace NaN values in specific columns with a specific string:

```
#replace NaN values in 'points' and 'rebounds' columns  
with 'none'
```

```
df] = df].fillna('none')
```

```
#view updated DataFrame
```

```
df
```

```
team points assists rebounds
```

```
0 A none 5.0 11.0
```

```
1 A 11.0 NaN 8.0
```

```
2 A 7.0 7.0 10.0
```

```
3 A 7.0 9.0 none
```

```
4 B 8.0 12.0 6.0
```

```
5 B 6.0 9.0 5.0
```

```
6 B 14.0 9.0 9.0
```

```
7 B 15.0 4.0 none
```

Notice that the NaN values in the 'points' and 'rebounds' columns were replaced with the string 'none', but the NaN values in the 'assists' column remained unchanged.

Method 3: Replace NaN Values with String in One Column

The following code shows how to replace NaN values in one column with a specific string:

```
#replace NaN values in 'points' column with 'zero'
```

```
df.points = df.points.fillna('zero')
```

```
#view updated DataFrame
```

```
df
```

```
team points assists rebounds
```

```
0 A zero 5.0 11.0
```

```
1 A 11.0 NaN 8.0
```

```
2 A 7.0 7.0 10.0
```

```
3 A 7.0 9.0 NaN
```

```
4 B 8.0 12.0 6.0
```

```
5 B 6.0 9.0 5.0
```

```
6 B 14.0 9.0 9.0
```

```
7 B 15.0 4.0 NaN
```

Notice that the NaN value in the 'points' column was replaced with the string 'zero', but the NaN values in the 'assists' and 'rebounds' columns remained unchanged.

Additional Resources

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