How to Check if a Row Exists in Another DataFrame

Authored by

stats writer

November 22, 2025

This is a common task when working with data frames, where one needs to check if a row in one dataframe exists in another dataframe. This can be done by comparing the values in the respective columns of the two data frames and checking if the values in the columns match. If a match is found, then the row exists in the other data frame. Otherwise, the row does not exist in the other data frame.

You can use the following syntax to add a new column to a pandas DataFrame that shows if each row exists in another DataFrame:

#merge two DataFrames on specific columns
all_df = pd.merge(df1, df2, on=, how='left', indicator='exists')

#drop unwanted columns
all_df = all_df.drop('column3', axis=1)

#add column that shows if each row in one DataFrame exists in another
all_df = np.where(all_df.exists == 'both', True, False)

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

Example: Check if Row in One Pandas DataFrame Exist in Another

Suppose we have the following two pandas DataFrames:

import pandas as pd

#create first DataFrame
df1 = pd.DataFrame({'team' : ,
'points' : })

print(df1)

team points
0 A 12
1 B 15
2 C 22
3 D 29
4 E 24

#create second DataFrame
df2 = pd.DataFrame({'team' : ,

'points' : ,
'assists' : })

print(df2)

team points assists
0 A 12 4
1 D 29 7
2 F 15 7
3 G 19 10
4 H 10 12

We can use the following syntax to add a column called **exists** to the first DataFrame that shows if each value in the **team** and **points** column of each row exists in the second DataFrame:

import numpy as np

#merge two dataFrames and add indicator column
all_df = pd.merge(df1, df2, on=, how='left', indicator='exists')

#drop assists columns
all_df = all_df.drop('assists', axis=1)

#add column to show if each row in first DataFrame exists in second
all_df = np.where(all_df.exists == 'both', True, False)

#view updated DataFrame
print (all_df)

team points exists
0 A 12 True
1 B 15 False
2 C 22 False
3 D 29 True
4 E 24 False

The new **exists** column shows if each value in the **team** and **points** column of each row exists in the second DataFrame.

From the output we can see:

A Team value of **A** and points value of **12** does exist in the second DataFrame.

A Team value of **B** and points value of **15** does not exist in the second DataFrame.
A Team value of **C** and points value of **22** does not exist in the second DataFrame.
A Team value of **D** and points value of **29** does exist in the second DataFrame.
A Team value of **E** and points value of **24** does not exist in the second DataFrame.

Also note that you can specify values other than True and False in the **exists** column by changing the values in the NumPy **where()** function.

For example, you could instead use 'exists' and 'not exists' as follows:

#add column to show if each row in first DataFrame exists in second
all_df = np.where(all_df.exists == 'both', 'exists', 'not exists')

#view updated DataFrame
print (all_df)

team points exists
0 A 12 exists
1 B 15 not exists
2 C 22 not exists
3 D 29 exists
4 E 24 not exists

Notice that the values in the **exists** column have been changed.