How can I use the `isin()` and `query()` methods in Pandas to filter data in a DataFrame?

Authored by

stats writer

June 24, 2024

The `isin()` and `query()` methods in Pandas are useful tools for filtering data in a DataFrame. The `isin()` method allows you to check if a value is present in a column or not, while the `query()` method allows you to filter data based on specific conditions. By using these methods, you can easily extract the desired data from a DataFrame, making it easier to analyze and manipulate the data. These methods are efficient and user-friendly, making them essential tools for data analysis in Pandas.

Pandas: Use isin() with query() Method

Often you may want to use the isin() function within the query() method in pandas to filter for rows in a DataFrame where a column contains a value in a list.

You can use the following syntax to do so:

df.query('team in ')

This particular query filters for rows in a pandas DataFrame where the team column is equal to A, B, or D.

Note: We must use in instead of isin when using the pandas query() method.

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

Example: Use query() Method to Filter for Values in List

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

```
import pandas as pd

#create DataFrame
df = pd.DataFrame({'team': ,
'points': ,
'assists': ,
'rebounds': })

#view DataFrame
print(df)
```

```
team points assists rebounds
0 A 18 5 11
1 A 22 7 8
2 B 19 7 10
3 B 14 9 6
4 C 14 12 6
5 C 11 9 5
6 D 20 9 9
7 E 28 4 12
```

Now suppose that we would like to query for the rows where the value in the team column is equal to either A, B, or D.

We can use the following syntax to do so:

#query for rows where team is in list of specific teams
df.query('team in ')

team points assists rebounds
0 A 18 5 11
1 A 22 7 8
2 B 19 7 10
3 B 14 9 6
6 D 20 9 9

Notice that the query() function returns all rows where the value in the team column is equal to A, B, or D.

Also note that we can store a list of team names in a variable and then reference the variable in the query() function using the @ operator:

#create variable to hold specific team names
team_names = #query for rows where team is equal to a

team name in team_names variable

df.query('team in @team_names')

team points assists rebounds

0 A 18 5 11

1 A 22 7 8

2 B 19 7 10

3 B 14 9 6

6 D 20 9 92

The query returns all of the rows in the DataFrame where team is equal to one of the team names stored in the team_names variable.

Notice that the results of this query match the one from the previous example.

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