How can I set a column in a data frame as the index in R, and can you provide an example?

Authored by

stats writer

June 28, 2024

To set a column as the index in a data frame using R, the "setDF" function can be used. This function takes two arguments, the first being the data frame and the second being the column to be set as the index. This function will return a new data frame with the specified column as the index. An example of this process would be as follows:

df

Set Data Frame Column as Index in R (With Example)

Data frames in R do not have an "index" column like data frames in pandas might.

However, data frames in R do have row names, which act similar to an index column.

You can use one of the following methods to set an existing data frame column as the row names for a data frame in R:

Method 1: Set Row Names Using Base R

#set specific column as row names
rownames(df) <- df$my_column

#remove original column from data frame
df$my_column <- NULL

Method 2: Set Row Names Using Tidyverse Package

```
library(tidyverse)#set specific column as row names
df <- df %>% column_to_rownames(., var = 'my_column')
```

Method 3: Set Row Names When Importing Data

```
#import CSV file and specify column to use as row names
df <- read.csv('my_data.csv', row.names='my_column')
```

The following examples show how to use each method in practice.

Example 1: Set Row Names Using Base R

Suppose we have the following data frame in R:

```
#create data frame
df <- data.frame(ID=c(101, 102, 103, 104, 105),
points=c(99, 90, 86, 88, 95),
assists=c(33, 28, 31, 39, 34),
rebounds=c(30, 28, 24, 24, 28))

#view data frame
df
```

ID points assists rebounds

1 101 99 33 30

2 102 90 28 28

3 103 86 31 24

4 104 88 39 24

5 105 95 34 28

We can use the following code to set the ID column as the row names:

```
#set ID column as row names
rownames(df) <- df$ID

#remove original ID column from data frame
df$ID <- NULL

#view updated data frame
df
```

points assists rebounds

101 99 33 30

102 90 28 28

103 86 31 24

104 88 39 24

105 95 34 28

The values from the ID column are now the row names for the data frame.

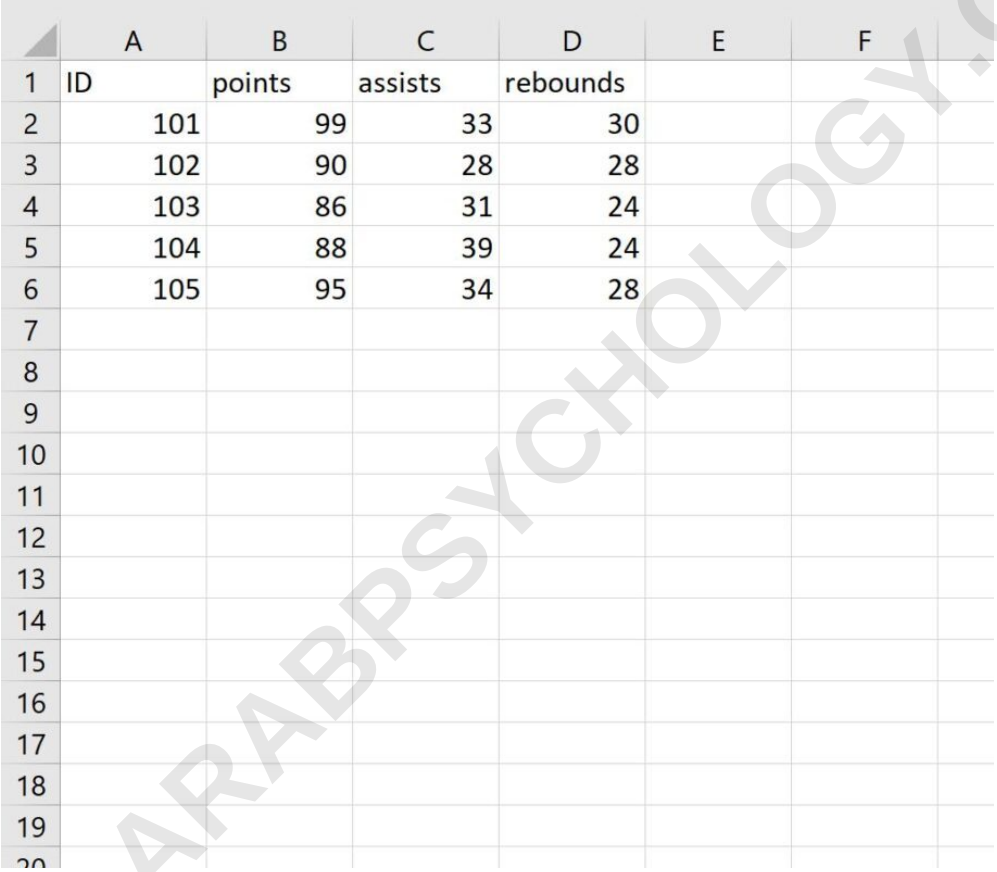
Example 2: Set Row Names Using Tidyverse Package

```
library(tidyverse)

#create data frame
df <- data.frame(ID=c(101, 102, 103, 104, 105),
points=c(99, 90, 86, 88, 95),
assists=c(33, 28, 31, 39, 34),
rebounds=c(30, 28, 24, 24, 28))

#set ID column as row names
df <- df %>% column_to_rownames(., var = 'ID')

#view updated data frame
df

points assists rebounds
101 99 33 30
102 90 28 28
103 86 31 24
104 88 39 24
105 95 34 28
```

Notice that this result matches the one from the previous example.

Example 3: Set Row Names When Importing Data

Suppose we have the following CSV file called my_data.csv:



	A	B	C	D	E	F
1	ID	points	assists	rebounds		
2	101	99	33	30		
3	102	90	28	28		
4	103	86	31	24		
5	104	88	39	24		
6	105	95	34	28		
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						

We can use the following code to import the CSV file and set the row names to be equal to the ID column when importing:

```
#import CSV file and specify ID column to use as row names
df <- read.csv('my_data.csv', row.names='ID')

#view data frame
df

points assists rebounds
101 99 33 30
102 90 28 28
103 86 31 24
104 88 39 24
105 95 34 28
```

Notice that the values from the ID column are used as the row names in the data frame.

Additional Resources

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