

How can I add a new column to a dataframe in R?

Authored by
stats writer

June 24, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I add a new column to a dataframe in R?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=149616>

Adding a new column to a dataframe in R is a process of creating and inserting a new column of data into an existing dataframe. This can be achieved by using the "cbind()" function and providing the name and values of the new column to be added. Alternatively, the "mutate()" function from the dplyr package can also be used to add a new column based on calculations or transformations of existing columns. Both methods allow for efficient and flexible ways to add new data to an existing dataframe in R.

To add a new column or variable to the DataFrame (data.frame) in R use either the \$ operator, notation, cbin() base function, or add_column() function from the tibble package.

In this article, I will explain how to insert a new columns of single/multiple in R, and insert a column with a constant value, insert before or after the column e.t.c with examples.

The following are different methods to add new columns to Data Frame in R that I covered in this article.

\$ operator notation cbin() R base function add_column() function from tibble. mutate() function from dplyr.

1. Quick Examples of Add Column to DataFrame

Following are quick examples of how to add a single column or multiple columns to DataFrame in R.

```
# Below are the quick examples

# Example 1: Using $ notation
chapters = c(76,86)
df$chapters = chapters

# Example 2: By using bracket notation
chapters = c(76,86)
df <- chapters

# Example 3: Add new column
chapters = c(76,86)
df2 <- cbind(df, chapters)

# Example 4: Add multiple columns to dataframe
chapters = c(76,86)
price=c(144,553)
```

```
df3 <- cbind(df, chapters, price)

# Example 5: Add new column from existing column
df$new_pages <- df$pages-2

library('tidyverse')
# using add_column()
df2 <- df %>%
add_column(add_column = "constant_value")
```

What is DataFrame in R

A **data frame** in R is a fundamental data structure that is used for storing and manipulating structured data which is in the format of rows and columns similar to an RDBMS table or spreadsheet. It is a two-dimensional data structure such that one dimension refers to the row and another dimension refers to a column. Each column in the data frame is a Vector of the same length, in other words, all columns in the data frame should have the same length.

Let's [create an R DataFrame](#), [run these examples](#), and explore the output. If you already have data in CSV you can easily [import CSV files to R DataFrame](#). Also, refer to [Import Excel File into R](#).

```
# Create dataframe
df=data.frame(id=c(11,22),
pages=c(32,45),
name=c("spark","python"))
df
```

Yields below output.

```
  id pages  name
1 11   32 spark
2 22   45 python
```

2. Add Column in R using \$ Operator

You can use the [dollar\(\\$\)](#) in R to add a new column in R. We know that the dataframe can be formed using a list of vectors of the same length. The vectors in a list represent columns of

dataFrame. Hence, you can add a new column to the dataframe by simply using `$` notation specifying the vector of the same length and adding it to the dataframe.

Note: The length of the new vector (column) must be the length of existing vectors(columns).

Let's create a new column using a vector and assign this to the `df$new_colname`. The following example adds a new column `chapters` to the DataFrame.

```
# Using $ notation to add new column
chapters = c(76,86)
df$chapters <- chapters
df
```

Yields below output.

	id	pages	name	chapters
1	11	32	spark	76
2	22	45	python	86

3. Add Column in R using Notation

Similarly, another way to refer to a column name is by using `df`, assigning a vector to this to have a new column. This gives the same result as the `$` operator however, this is the best approach to use as it is more user-friendly and the code is easy to read.

```
# By using bracket notation
chapters = c(76,86)
df <- chapters
df
```

Yields the same output as above.

4. Add Column Using `cbin()` Function

To add a new column in R, use `cbin()` function. This function takes a DataFrame as a first argument and for the second argument, use the vector which creates a new column on the DataFrame. Note that this doesn't update the existing DataFrame instead it returns a copy of the DataFrame after adding a new column.

```
# Add new column
chapters = c(76,86)
df2 <- cbind(df, chapters)
df2
```

Yields the same output as above.

5. Add Multiple Columns to DataFrame

By using the same `cbind()` function you can add multiple columns to the DataFrame in R. The following example adds columns `chapters` and `price` to the DataFrame (`data.frame`).

```
# Add multiple columns to dataframe
chapters = c(76,86)
price=c(144,553)
df3 <- cbind(df, chapters, price)
```

Output

```
# id pages name chapters price
# 1 11 32 spark 76 144
# 2 22 45 python 86 553
```

6. Add a New Column from the Existing

Finally using the R base let's see how to add a new column from the existing DataFrame column.

```
# Add new column from existing column
df$new_pages <- df$pages-2
df
```

Output

```
# id pages name new_pages
#1 11 32 spark 30
#2 22 45 python 43
```

7. By using `add_column()` from tidyverse

Use `add_column()` function from `tidyverse` to add a column to the DataFrame in R. This is a

convenient way to add one or more columns to an existing data frame. To use this function first, you need to install R package by using `install.packages("tidyverse")` and load it using the `library("tidyverse")`.

The following example adds a constant column to DataFrame. All rows will have the same value.

```
# Load tidyverse
library("tidyverse")

# Using add_column()
df2 <- df %>%
  add_column(add_column = "constant_value")
df2

# Output
# id pages name add_column
#1 11 32 spark constant_value
#2 22 45 python constant_value
```

By using the methods explained above you can also add an empty column to DataFrame in R.

Conclusion

In this article, you have learned how to add new single/multiple columns in R and add a column with a constant value, e.t.c with multiple examples.

You can also find examples explained in this article at [R GitHub Project](#).

Related Articles