

How can I combine two columns into one in R, and what are some examples of doing so?

Authored by
stats writer

April 20, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I combine two columns into one in R, and what are some examples of doing so?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=137415>

In R, it is possible to combine two columns into one using various methods. This can be achieved through the use of functions such as `cbind()`, `merge()`, and `paste()`, depending on the desired outcome. For example, `cbind()` can be used to create a new column by combining two or more existing columns from a data frame. On the other hand, `merge()` can be used to combine two data frames by matching a specific column. The `paste()` function can be used to merge two columns by concatenating their values. These methods allow for the creation of new data structures that can be used for further analysis or visualization.

Combine Two Columns into One in R (With Examples)

Often you may want to combine two columns into one in R. For example, suppose you have a data frame with three columns:

month year value

10 2019 15

10 2020 13

11 2020 13

11 2021 19

12 2021 22

You may wish to combine the month and year column into a single column called *date*:

date value

2019_10 15

2020_10 13

2020_11 13

2021_11 19

2021_12 22

This tutorial explains two ways to quickly do this in R.

Method 1: Use the Paste Function from Base R

The following code shows how to use the paste function from base R to combine the columns *month* and *year* into a single column called *date*:

```
#create data frame
```

```
data <- data.frame(month=c(10, 10, 11, 11, 12),
```

```
year=c(2019, 2020, 2020, 2021, 2021),
```

```
value=c(15, 13, 13, 19, 22))
```

```
#view data frame
```

```
data
```

```
#combine year and month into one column
```

```
data$date <- paste(data$year, data$month, sep="_")
```

```
#view new data frame
```

```
data
```

month year value date

1 10 2019 15 2019_10

2 10 2020 13 2020_10

3 11 2020 13 2020_11

4 11 2021 19 2021_11

5 12 2021 22 2021_12

Once we've combined the two columns, we can remove the old ones if we'd like:

```
data_new <- data
```

```
data_new
```

date value

1 2019_10 15

2 2020_10 13

3 2020_11 13

4 2021_11 19

5 2021_12 22

Method 2: Use the Unite Function from Tidyr

The following code shows how to use the unite function from the tidyr package to combine the columns *month*

and *year* into a single column called *date*:

```
#load tidy package
```

```
library(tidyr)
```

```
#create data frame
```

```
data <- data.frame(month=c(10, 10, 11, 11, 12),  
year=c(2019, 2020, 2020, 2021, 2021),  
value=c(15, 13, 13, 19, 22))
```

```
#combine year and month into one column  
unite(data, date, c(year, month))
```

```
date value
```

```
1 2019_10 15
```

```
2 2020_10 13
```

```
3 2020_11 13
```

```
4 2021_11 19
```

```
5 2021_12 22
```

Notice that both methods produce identical results.

You can find the complete documentation for the `unite` function [here](#).