

How to use fread() in R to Import Files Faster

Authored by
stats writer

November 28, 2025

RECOMMENDED CITATION

stats writer (2025). *How to use fread() in R to Import Files Faster*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=101248>

The `fread()` function in R is a fast and versatile way to read data into R from flat files like csv, txt, and delimited files. It is especially useful for large datasets because it is much faster than other functions like `read.csv()`. It can also handle a variety of parameters like header, sep, encoding, and quoted parameters to quickly import and manipulate data.

You can use the **`fread()`** function from the **`data.table`** package in R to import files quickly and conveniently.

This function uses the following basic syntax:

```
library(data.table)
```

```
df <- fread("C:\Users\Path\To\My\data.csv")
```

For large files, this function has been shown to be significantly faster than functions like **`read.csv`** from base R.

And in most cases, this function can also automatically detect the delimiter and column types for the dataset you're importing.

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

Example: How to Use fread() to Import Files in R

Suppose I have a CSV file called **`data.csv`** saved in the following location:

```
C:\Users\Bob\Desktop\data.csv
```

And suppose the CSV file contains the following data:

```
team, points, assists
```

```
'A', 78, 12
```

```
'B', 85, 20
```

```
'C', 93, 23
```

```
'D', 90, 8
```

```
'E', 91, 14
```

I can use the **`fread()`** function from the **`data.table`** package to import this file into my current R environment:

```
library(data.table)
```

```
#import data
df <- fread("C:\\Users\\Bob\\Desktop\\data.csv")

#view data
df

team points assists
1 A 78 12
2 B 85 20
3 C 93 23
4 D 90 8
5 E 91 14
```

We're able to successfully import the CSV file using the **fread()** function.

Note: We used double backslashes (\) in the file path to avoid a .

Notice that we didn't have to specify the delimiter either since the **fread()** function automatically detected that it was a comma.

If we use the function to view the structure of the data frame, we can see that the **fread()** function automatically identified the object type for each column as well:

```
#view structure of data
str(df)
```

```
Classes 'data.table' and 'data.frame': 5 obs. of 3 variables:
```

```
$ team : chr "A" "B" "C" "D" ...
```

```
$ points : int 78 85 93 90 91
```

```
$ assists: int 12 20 23 8 14
```

From the output we can see:

The **team** variable is a character.

The **points** variable is an integer.

The **assists** variable is an integer.

In this example we used a small data frame for simplicity (5 rows x 3 columns) but in practice the **fread()** function is able to quickly and efficiently import data frames with tens of thousands of rows, which makes it the preferred import method for large-scale datasets.

The following tutorials explain how to import specific file types into R:

ARABPSYCHOLOGY.COM