

# How can I list files by date in R, with an example?

Authored by  
**stats writer**

June 23, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can I list files by date in R, with an example?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=148532>

To list files by date in R, one can use the function "file.info()" which provides information about files in a specified directory. This function returns a data frame with details such as file size, date of creation, and date of last modification. One can then use the "order()" function to sort the data frame by the desired date column. An example of this process would be using the "file.info()" function on a directory containing various files, and then using the "order()" function to sort the data frame by the modified date column, resulting in a list of files sorted by date. This method can be useful for organizing files and identifying the most recently modified files.

## List Files by Date in R (With Example)

You can use the following basic syntax to list the files in the current working directory in R by date:

```
#extract all CSV files in working directory
```

```
file_info = file.info(list.files(pattern="*.csv"))
```

```
#sort files based on mtime (modification date and time)
```

```
file_info = file_info
```

```
#view only file names with modification date and time
```

```
file_info
```

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

**Example: How to List Files by Date in R**

**Suppose I would like to list all of the CSV files in the in R by date.**

I can use the following syntax to first extract all CSV files (with file information) from the current working directory:

```
#extract all CSV files in working directory
```

```
file_info = file.info(list.files(pattern="*.csv"))
```

```
#view all CSV files
```

```
file_info
```

```
size isdir mode mtime ctime atime exe
```

```
basketball_data.csv 55 FALSE 666 2023-01-06 11:07:43
```

```
2022-07-12 09:07:26 2023-04-18 09:42:19 no
```

```
df1.csv 126 FALSE 666 2022-04-21 10:48:24 2022-04-21
```

```
10:48:24 2023-04-18 09:42:19 no
```

```
df2.csv 126 FALSE 666 2022-04-21 10:48:30 2022-04-21
```

```
10:48:29 2023-04-18 09:42:19 no
```

```
df3.csv 126 FALSE 666 2022-04-21 10:48:34 2022-04-21
```

```
10:48:34 2023-04-18 09:42:19 no
```

```
my_data.csv 53 FALSE 666 2022-09-09 09:02:21
```

```
2022-04-22 09:00:13 2023-04-18 09:42:19 no
```

```
my_list.csv 90 FALSE 666 2022-04-21 09:40:01
```

```
2022-04-21 09:39:59 2023-04-18 09:42:19 no
```

```
my_test.csv 146 FALSE 666 2022-04-21 09:42:25
```

```
2022-04-21 09:42:25 2023-04-18 09:42:19 no
```

```
player_stats.csv 137 FALSE 666 2023-04-11 09:07:20
2023-04-11 09:07:20 2023-04-18 09:42:19 no
players_data.csv 50 FALSE 666 2023-01-06 09:44:12
2023-01-06 09:44:12 2023-04-18 09:42:19 no
team_info.csv 131 FALSE 666 2023-04-11 09:07:21
2023-04-11 09:07:21 2023-04-18 09:42:19 no
test.csv 18059168 FALSE 666 2022-09-07 09:07:34
2020-02-01 13:44:03 2023-04-18 09:42:19 no
uneven_data.csv 43 FALSE 666 2023-01-06 14:02:17
2023-01-06 14:00:27 2023-04-18 09:42:19 no
```

I can then use the `order()` function to order the files by `mtime`, which represents the date and time the files were most recently modified:

```
#sort files based on mtime (modification date and time)
```

```
file_info = file_info
```

```
#view sorted files
```

```
file_info
```

```
size isdir mode mtime ctime atime exe
```

```
my_list.csv 90 FALSE 666 2022-04-21 09:40:01
2022-04-21 09:39:59 2023-04-18 09:42:19 no
```

```
my_test.csv 146 FALSE 666 2022-04-21 09:42:25
```

```
2022-04-21 09:42:25 2023-04-18 09:42:19 no
df1.csv 126 FALSE 666 2022-04-21 10:48:24 2022-04-21
10:48:24 2023-04-18 09:42:19 no
df2.csv 126 FALSE 666 2022-04-21 10:48:30 2022-04-21
10:48:29 2023-04-18 09:42:19 no
df3.csv 126 FALSE 666 2022-04-21 10:48:34 2022-04-21
10:48:34 2023-04-18 09:42:19 no
test.csv 18059168 FALSE 666 2022-09-07 09:07:34
2020-02-01 13:44:03 2023-04-18 09:42:19 no
my_data.csv 53 FALSE 666 2022-09-09 09:02:21
2022-04-22 09:00:13 2023-04-18 09:42:19 no
players_data.csv 50 FALSE 666 2023-01-06 09:44:12
2023-01-06 09:44:12 2023-04-18 09:42:19 no
basketball_data.csv 55 FALSE 666 2023-01-06 11:07:43
2022-07-12 09:07:26 2023-04-18 09:42:19 no
uneven_data.csv 43 FALSE 666 2023-01-06 14:02:17
2023-01-06 14:00:27 2023-04-18 09:42:19 no
player_stats.csv 137 FALSE 666 2023-04-11 09:07:20
2023-04-11 09:07:20 2023-04-18 09:42:19 no
team_info.csv 131 FALSE 666 2023-04-11 09:07:21
2023-04-11 09:07:21 2023-04-18 09:42:19 no
```

Note that you could instead use `ctime` if you'd like to order the files based on *creation date* instead or `atime`

to order the files based on *accessed date*.

Lastly, we can subset the data frame to only show the file names along with the date and time they were most recently modified:

```
#view only file names with modification date and time  
file_info
```

```
mtime
```

```
my_list.csv 2022-04-21 09:40:01  
my_test.csv 2022-04-21 09:42:25  
df1.csv 2022-04-21 10:48:24  
df2.csv 2022-04-21 10:48:30  
df3.csv 2022-04-21 10:48:34  
test.csv 2022-09-07 09:07:34  
my_data.csv 2022-09-09 09:02:21  
players_data.csv 2023-01-06 09:44:12  
basketball_data.csv 2023-01-06 11:07:43  
uneven_data.csv 2023-01-06 14:02:17  
player_stats.csv 2023-04-11 09:07:20  
team_info.csv 2023-04-11 09:07:21
```

If you'd like, you could also only view the file names in order by date:

**#view only file names**

**rownames(file\_info)**

**"my\_list.csv" "my\_test.csv" "df1.csv" "df2.csv"**

**"df3.csv"**

**"test.csv" "my\_data.csv" "players\_data.csv"**

**"basketball\_data.csv" "uneven\_data.csv"**

**"player\_stats.csv" "team\_info.csv"**

**The twelve CSV file names are listed in order by date.**