

How can I create an empty data frame in R?

Authored by
stats writer

April 21, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I create an empty data frame in R?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=137586>

Creating an empty data frame in R involves using the "data.frame()" function and specifying the number of rows and columns desired. This will create a blank data frame with the specified dimensions, ready to be populated with data. Additionally, the "data.frame()" function allows for specifying column names and data types, if desired. An empty data frame serves as a starting point for data manipulation and analysis in R.

Create an Empty Data Frame in R (With Examples)

There are two basic ways to create an empty data frame in R:

Method 1: Matrix with Column Names

```
#create data frame with 0 rows and 3 columns
```

```
df <- data.frame(matrix(ncol = 3, nrow = 0))
```

```
#provide column names
```

```
colnames(df) <- c('var1', 'var2', 'var3')
```

Method 2: Initialize Empty Vectors

```
#create data frame with 5 empty vectors
```

```
df2 <- data.frame(Doubles=double(),
```

```
Integers=integer(),
```

```
Factors=factor(),
```

```
Logicals=logical(),
```

```
Characters=character(),
```

stringsAsFactors=FALSE)

This tutorial shows examples of how to use these two methods in practice.

Method 1: Matrix with Column Names

The first way to create an empty data frame is by using the following steps:

Define a matrix with 0 rows and however many columns you'd like. Then use the `data.frame()` function to convert it to a data frame and the `colnames()` function to give it column names. Then use the `str()` function to analyze the structure of the resulting data frame.

For example:

```
#create data frame with 0 rows and 5 columns
```

```
df <- data.frame(matrix(ncol = 5, nrow = 0))
```

```
#provide column names
```

```
colnames(df) <- c('var1', 'var2', 'var3', 'var4', 'var5')
```

```
#view structure of the data frame
```

```
str(df)
```

'data.frame': 0 obs. of 5 variables:

\$ var1: logi

\$ var2: logi

\$ var3: logi

\$ var4: logi

\$ var5: logi

We can see that the resulting data frame has 0 observations (i.e. rows), 5 variables (i.e. columns), and each of the variables are of the class *logical*.

Although each variable is of the class *logical*, you can still add rows to the variables that are of different types.

Method 2: Initialize Empty Vectors

The second way to create an empty data frame is by using the following steps:

Define a data frame as a set of empty vectors with specific class types. Specify `stringsAsFactors=False` so that any character vectors are treated as strings, not factors.

For example:

```
#create data frame with 5 empty vectors
```

```
df2 <- data.frame(Doubles=double(),
```

```
Integers=integer(),
```

```
Factors=factor(),
```

```
Logicals=logical(),
```

```
Characters=character(),
```

```
stringsAsFactors=FALSE)
```

```
#view structure of the data frame
```

```
str(df2)
```

```
'data.frame': 0 obs. of 5 variables:
```

```
$ Doubles : num
```

```
$ Integers : int
```

```
$ Factors : Factor w/ 0 levels:
```

```
$ Logicals : logi
```

```
$ Characters: chr
```

We can see that the resulting data frame has 0 observations (i.e. rows), 5 variables (i.e. columns), and each of the variables are five different classes.

Note that we were also able to provide column names for the data frame in just one step (e.g. the first column name is "Doubles", the second column name is

"Integers" and so on.

The following tutorials explain how to create other empty objects in R:

How to Create an Empty List in R

ARABPSYCHOLOGY.COM