

How to Fix the “Argument is of Length Zero” Error in R: A Simple Guide

Authored by
stats writer

November 29, 2025

RECOMMENDED CITATION

stats writer (2025). *How to Fix the “Argument is of Length Zero” Error in R: A Simple Guide*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=101741>

The "argument is of length zero" error in R occurs when you try to pass an empty argument to a function. To fix this, you need to pass a valid argument to the function, either a value, vector, or other data type. If the argument is intended to be empty, you can explicitly set the argument to NULL.

One error message you may encounter when using R is:

Error in if (x < 10) { : argument is of length zero

This error usually occurs when you attempt to make some logical comparison within an if statement in R, but the variable that you're using in the comparison is of length zero.

Two examples of variables with length zero are **numeric()** or **character()**.

The following example shows how to resolve this error in practice.

How to Reproduce the Error

Suppose we create the following numeric variable in R with a length of zero:

```
#create numeric variable with length of zero
x <- numeric()
```

Now suppose we attempt to use this variable in an if statement:

```
#if x is less than 10, print x to console
if(x < 10) {
  x
}
```

Error in if (x < 10) { : argument is of length zero

We receive an error because the variable that we defined has a length of zero.

If we simply created a numeric variable with an actual value, we would never receive this error when using the if statement:

```
#create numeric variable
y <- 5
```

```
#if y is less than 10, print y to console
```

```
if(y < 10) {  
  y  
}
```

5

How to Avoid the Error

To avoid the **argument is of length zero** error, we must include an **isTRUE** function, which uses the following logic:

```
is.logical(x) && length(x) == 1 && !is.na(x) && x
```

If we use this function in the if statement, we won't receive an error when comparing our variable to some value:

```
if(isTRUE(x) && x < 10) {  
  x  
}
```

Instead of receiving an error, we simply receive no output because the **isTRUE(x)** function evaluates to **FALSE**, which means the value of x is never printed.

The following tutorials explain how to fix other common errors in R: