

# How can NaN values be handled in R, and what are some examples?

Authored by  
**stats writer**

June 28, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can NaN values be handled in R, and what are some examples?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=156872>

NaN (Not a Number) values are a common occurrence in data analysis and can cause disruptions in data processing. In R, there are various methods to handle NaN values and ensure their proper treatment. One way is to use the "is.na" function, which identifies and flags the NaN values in a dataset. These flagged values can then be replaced with appropriate values using the "na.omit" or "na.exclude" functions. Another approach is to impute the NaN values with the mean, median, or mode of the corresponding column using the "na.aggregate" function. Additionally, specialized packages such as "mice" and "missMDA" offer advanced techniques for handling NaN values. Some examples of NaN values that can be encountered in R data analysis include missing values, infinite values, and non-numeric values such as "NA", "NaN", and "Inf". Proper handling of NaN values is crucial for accurate and reliable data analysis.

## Handle NaN Values in R (With Examples)

**In R, NaN stands for Not a Number.**

**Typically NaN values occur when you attempt to perform some calculation that results in an invalid result.**

**For example, dividing by zero or calculating the log of a negative number both produce NaN values:**

**#attempt to divide by zero**

**0 / 0**

**NaN**

**#attempt to calculate log of negative value**

**log(-12)**

## NaN

**Note that NaN values are different from NA values, which simply represent missing values.**

**You can use the following methods to handle NaN values in R:**

**#identify positions in vector with NaN values  
which(is.nan(x))**

**#count total NaN values in vector  
sum(is.nan(x))**

**#remove NaN values in vector  
x\_new <- x**

**#replace NaN values in vector  
x <- 0**

**The following examples show how to use each of these methods in practice.**

**Example 1: Identify Positions in Vector with NaN Values**

**The following code shows how to identify the positions**

**in a vector that contain NaN values:**

```
#create vector with some NaN values
```

```
x <- c(1, NaN, 12, NaN, 50, 30)
```

```
#identify positions with NaN values
```

```
which(is.nan(x))
```

```
2 4
```

**From the output we can see that the elements in positions 2 and 4 in the vector are NaN values.**

**Example 2: Count Total NaN Values in Vector**

**The following code shows how to count the total number of NaN values in a vector in R:**

```
#create vector with some NaN values
```

```
x <- c(1, NaN, 12, NaN, 50, 30)
```

```
#identify positions with NaN values
```

```
sum(is.nan(x))
```

```
2
```

**From the output we can see that there are 2 total NaN values in the vector.**

**Example 3: Remove NaN Values in Vector**

```
#create vector with some NaN values
```

```
x <- c(1, NaN, 12, NaN, 50, 30)
```

```
#define new vector with NaN values removed
```

```
x_new <- x
```

```
#view new vector
```

```
x_new
```

```
1 12 50 30
```

**Notice that both NaN values have been removed from the vector.**

**Example 4: Replace NaN Values in Vector**

**The following code shows how to replace NaN values in a vector with zeros:**

```
#create vector with some NaN values
```

```
x <- c(1, NaN, 12, NaN, 50, 30)
```

```
#replace NaN values with zero
```

```
x <- 0
```

```
#view updated vector
```

```
x
```

```
1 0 12 0 50 30
```

**Notice that both NaN values have been replaced by zeros in the vector.**

**Additional Resources**

**The following tutorials explain how to perform other common tasks in R:**