

How can an If statement be used with multiple conditions in order to rewrite code using the -R option?

Authored by
stats writer

June 29, 2024

RECOMMENDED CITATION

stats writer (2024). *How can an If statement be used with multiple conditions in order to rewrite code using the -R option?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=159654>

An If statement can be used with multiple conditions in order to rewrite code using the -R option. This allows for more efficient and concise code by eliminating the need for multiple separate If statements. The -R option allows for the evaluation of all conditions simultaneously, resulting in a more streamlined and organized code structure. Additionally, using an If statement with multiple conditions allows for better control and customization of program flow, as different actions can be taken based on the combination of conditions being met. Overall, utilizing an If statement with multiple conditions and the -R option can greatly improve the efficiency and effectiveness of code.

R: Use If Statement with Multiple Conditions

You can use the following methods to create a new column in R using an IF statement with multiple conditions:

Method 1: If Statement with Multiple Conditions Using OR

```
df$new_var <- ifelse(df$var1>15 | df$var2>8, "value1", "value2")
```

Method 2: If Statement with Multiple Conditions Using AND

```
df$new_var <- ifelse(df$var1>15 & df$var2>8, "value1", "value2")
```

The following examples show how to use each method in practice with the following data frame:

```
#create data frame
```

```
df <- data.frame(team=c('A', 'A', 'A', 'A', 'B', 'B', 'B', 'B'),  
points=c(8, 8, 10, 13, 17, 19, 22, 25),  
assists=c(5, 10, 9, 6, 8, 10, 11, 12))
```

```
#view data frame
```

```
df
```

```
team points assists
```

```
1 A 8 5  
2 A 8 10  
3 A 10 9  
4 A 13 6  
5 B 17 8  
6 B 19 10  
7 B 22 11  
8 B 25 12
```

Example 1: If Statement with Multiple Conditions Using OR

The following code shows how to create a new column called rating that assigns a value of "good" if the points column is greater than 15 or the assists column is greater than 8.

Otherwise it assigns a value of "bad":

#create new "rating" column using if statement with multiple conditions

```
df$rating <- ifelse(df$points>15 | df$assists>8, "good", "bad")
```

#view updated data frame

df

team points assists rating

1 A 8 5 bad

2 A 8 10 good

3 A 10 9 good

4 A 13 6 bad

5 B 17 8 good

6 B 19 10 good

7 B 22 11 good

8 B 25 12 good

Each player receives a value of "good" or "bad" in the newly created rating column.

Note that the | operator is used as an "or" statement in R.

Example 2: If Statement with Multiple Conditions Using AND

The following code shows how to create a new column called rating that assigns a value of "good" if the points column is greater than 15 and the assists column is greater than 8.

Otherwise it assigns a value of "bad":

```
#create new "rating" column using if statement with multiple conditions
```

```
df$rating <- ifelse(df$points>15 & df$assists>8, "good", "bad")
```

```
#view updated data frame
```

```
df
```

```
team points assists rating
```

```
1 A 8 5 bad
```

```
2 A 8 10 bad
```

```
3 A 10 9 bad
```

```
4 A 13 6 bad
```

```
5 B 17 8 bad
```

```
6 B 19 10 good
```

```
7 B 22 11 good
```

```
8 B 25 12 good
```

Note that the & operator is used as an "and" statement in R.

Additional Resources

ARABPSYCHOLOGY.COM