

18.3: Conditional Statements

A second kind of flow control that programming languages provide is the ability to evaluate **conditional statements**. Unlike loops, which can repeat over and over again, a conditional statement only executes once, but it can switch between different possible commands depending on a **CONDITION** that is specified by the programmer. The power of these commands is that they allow the program itself to make choices, and in particular, to make different choices depending on the context in which the program is run. The most prominent example of a conditional statement is the `if` statement, and the accompanying `else` statement. The basic format of an `if` statement in R is as follows:

```
if ( CONDITION ) {  
    STATEMENT1  
    STATEMENT2  
    ETC  
}
```

And the execution of the statement is pretty straightforward. If the **CONDITION** is true, then R will execute the statements contained in the curly braces. If the **CONDITION** is false, then it does not. If you want to, you can extend the `if` statement to include an `else` statement as well, leading to the following syntax:

```
if ( CONDITION ) {  
    STATEMENT1  
    STATEMENT2  
    ETC  
} else {  
    STATEMENT3  
    STATEMENT4  
    ETC  
}
```

As you'd expect, the interpretation of this version is similar. If the **CONDITION** is true, then the contents of the first block of code (i.e., `STATEMENT1`, `STATEMENT2`, `ETC`) are executed; but if it is false, then the contents of the second block of code (i.e., `STATEMENT3`, `STATEMENT4`, `ETC`) are executed instead.

To give you a feel for how you can use `if` and `else` to do something useful, the example that I'll show you is a script that prints out a different message depending on what day of the week you run it. We can do this making use of some of the tools that we discussed in Section 7.11.3. Here's the script:

```
## --- ifelseexample.R  
# find out what day it is...  
today <- Sys.Date()          # pull the date from the system clock  
day <- weekdays( today )     # what day of the week it is_  
  
# now make a choice depending on the day...  
if ( day == "Monday" ) {  
    print( "I don't like Mondays" )  
} else {  
    print( "I'm a happy little automaton" )  
}
```

```
## [1] "I'm a happy little automaton"
```

Since today happens to be a Sunday, when I run the script here's what happens:

```
source( "./rbook-master/scripts/ifelseexample.R" )
```

```
## [1] "I'm a happy little automaton"
```

There are other ways of making conditional statements in R. In particular, the `ifelse()` function and the `switch()` functions can be very useful in different contexts. However, my main aim in this chapter is to briefly cover the very basics, so I'll move on.

This page titled [18.3: Conditional Statements](#) is shared under a [CC BY-SA 4.0](#) license and was authored, remixed, and/or curated by [Danielle Navarro](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **8.3: Conditional Statements** by [Danielle Navarro](#) is licensed [CC BY-SA 4.0](#). Original source: <https://bookdown.org/ekothe/navarro26/>.