

## Understand basic programming concepts

✓ Video: Programming using RStudio  
2 min

✓ Video: Programming fundamentals  
8 min

✓ Reading: Vectors and lists in R  
10 min

✓ Reading: Dates and times in R  
10 min

✓ Reading: Other common data structures  
10 min

✓ Practice Quiz: Test your knowledge on programming concepts  
4 questions

## Explore coding in R

## Learning about R packages

## Explore the tidyverse

## Weekly challenge 2

# Dates and times in R

In this reading, you will learn how to work with dates and times in R using the **lubridate** package. Coming up, you will use tools in the lubridate package to convert different types of data in R into date and date-time formats.



## Loading tidyverse and lubridate packages

Before you get started working with dates and times, you should load both **tidyverse** and **lubridate**. Lubridate is part of tidyverse.

First, open RStudio.

If you haven't already installed tidyverse, you can use the **install.packages()** function to do so:

```
install.packages("tidyverse")
```

Next, load the tidyverse and lubridate packages using the **library()** function. First, load the core tidyverse to make it available in your current R session:

```
library(tidyverse)
```

Then, load the lubridate package:

```
library(lubridate)
```

Now you're ready to be introduced to the tools in the lubridate package.

## Working with dates and times

This section covers the data types for dates and times in R and how to convert strings to date-time formats.

### Types

In R, there are three types of data that refer to an instant in time:

- A date (`"2016-08-16"`)
- A time within a day (`"~20:11:59 UTC"`)
- And a date-time. This is a date plus a time (`"2018-03-31 18:15:48 UTC"`)

The time is given in UTC, which stands for Universal Time Coordinated, more commonly called Universal Coordinated Time. This is the primary standard by which the world regulates clocks and time.

For example, to get the current date you can run the **today()** function. The date appears as year, month, and day.

```
today()
```

```
#> [1] "2021-01-20"
```

To get the current date-time you can run the **now()** function. Note that the time appears to the nearest second.

```
now()
```

```
#> [1] "2021-01-20 16:25:05 UTC"
```

When working with R, there are three ways you are likely to create date-time formats:

- From a string
- From an individual date
- From an existing date/time object

R creates dates in the standard yyyy-mm-dd format by default.

Let's go over each.

### Converting from strings

Date/time data often comes as strings. You can convert strings into dates and date-times using the tools provided by lubridate. These tools automatically work out the date/time format. First, identify the order in which the year, month, and day appear in your dates. Then, arrange the letters *y*, *m*, and *d* in the same order. That gives you the name of the lubridate function that will parse your date. For example, for the date 2021-01-20, you use the order *ymd*:

```
ymd("2021-01-20")
```

When you run the function, R returns the date in yyyy-mm-dd format.

```
#> [1] "2021-01-20"
```

It works the same way for any order. For example, month, day, and year. R still returns the date in yyyy-mm-dd format.

```
mdy("January 20th, 2021")
```

```
#> [1] "2021-01-20"
```

Or, day, month, and year. R still returns the date in yyyy-mm-dd format.

```
dmy("20-Jan-2021")
```

```
#> [1] "2021-01-20"
```

These functions also take unquoted numbers and convert them into the yyyy-mm-dd format.

These functions also take unquoted numbers and convert them into the `yyyy-mm-dd` format.

```
ymd(20210120)
```

```
#> [1] "2021-01-20"
```

#### Creating date-time components

The `ymd()` function and its variations create dates. To create a date-time from a date, add an underscore and one or more of the letters *h*, *m*, and *s* (hours, minutes, seconds) to the name of the function:

```
ymd_hms("2021-01-20 20:11:59")
```

```
#> [1] "2021-01-20 20:11:59 UTC"
```

```
mdy_hm("01/20/2021 08:01")
```

```
#> [1] "2021-01-20 08:01:00 UTC"
```

#### Optional: Switching between existing date-time objects

Finally, you might want to switch between a date-time and a date.



You can use the function `as_date()` to convert a date-time to a date. For example, put the current date-time—`now()`—in the parentheses of the function.

```
as_date(now())
```

```
#> [1] "2021-01-20"
```

#### Additional resources

To learn more about working with dates and times in R, check out the following resources:

- [lubridate.tidyverse](#) : This is the “lubridate” entry from the official tidyverse documentation, which offers a comprehensive reference guide to the various tidyverse packages. Check out this link for an overview of key concepts and functions.
- [Dates and times with lubridate: Cheat Sheet](#) : This “cheat sheet” gives you a detailed map of all the different things you can do with the lubridate package. You don’t need to know all of this information, but the cheat sheet is a useful reference for any questions you might have about working with dates and times in R.

✓ Completed

Go to next item

 Like  Dislike  Report an issue