

Introduction to the Tidyverse

Import, wrangle, model, and
communicate data

2024-08-13



Working with data in R

the tidyverse is a collection of *friendly and consistent* tools for data analysis and visualization.

They live as R packages, each of which does one thing well.

library(tidyverse) will load the core packages:



ggplot2, for data visualisation.

dplyr, for data manipulation.

tidyr, for data tidying.

readr, for data import.

purrr, for functional programming.

tibble, for tibbles, a modern re-imagining of data frames.

stringr, for strings.

forcats, for factors.

lubridate, for dates and times.

This course is hands on!

Each section has an
exercises file:
exercises.qmd

exercises.qmd

The image shows the Quarto editor interface. The top toolbar includes icons for navigation, rendering, and execution. The document is titled 'exercises.qmd'. The main editor area shows the following content:

```
1 ---  
2 title: "Import Data"  
3 format: html  
4 ---  
5  
6 ```{r}  
7 #| label: setup  
8 library(tidyverse)  
9 library(haven)  
10 ```  
11  
12 In this section, we will learn about importing and exporting  
13 files from common file formats, including CSV and formats from  
14 other statistical software using the readr and haven packages.  
15  
16 ## readr  
17  
18 readr supplies several related functions, each designed to  
19 read in a specific flat file format.  
20  
21 Function | Reads  
22 -----  
23 (Top Level) |
```

The right sidebar shows the document outline with the following items:

- readr
- Sample data
- Importing Data
- Your Turn 1
- Your Turn 1: Bonus
- Tibbles
- Missing values
- Parsing data types
- Your Turn 2
- haven: read and write S...
- Your Turn 3
- Writing data
- Your Turn 4
- Take Aways

The bottom status bar shows '5:1' and 'Quarto'.

Code chunks

```
```\{r}  
csv_data <- read_csv(
 "a,b,c,d
 1,2,3,4
 5,6,7,8",
 col_types = ""
)




csv_data
```\
```



Running code chunks

```
```{r}
csv_data <- read_csv(
 "a,b,c,d
1,2,3,4
5,6,7,8",
 col_types = ""
)
```

```
csv_data|
```
```



| a | b | c | d |
|----------|----------|----------|----------|
| <dbl> | <dbl> | <dbl> | <dbl> |
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |

2 rows

Outputting to the console

The screenshot shows the Quarto editor interface with a file named 'exercises.qmd'. The 'Source' tab is active, displaying R code. A context menu is open, showing options for how to preview and output the code chunk. The 'Chunk Output in Console' option is selected and highlighted in blue. The code in the background includes a title, an introduction to the 'readr' package, and a code chunk for reading a CSV file with specific column types.

```
102 data.  
103 ## Parsing data types  
104  
105 The read functions in readr sometimes it's wrong. For example, a column called 'ID' that is a number, but we usually want to treat it as a character. readr will show you what function to use. You can use the same function to read other files.  
106  
107 To do this, add the argument 'col_types' to 'read_csv()' and set it equal to a list. readr has several functions that start with 'col' that represent data types. We'll go over data and object types, including lists, later in the week.  
108  
109 ```{r}  
110 #| eval: false  
111 diabetes <- read_csv("diabetes.csv", col_types = list(id =  
112   col_character()))  
113 ```
```

The context menu options are:

- Use Visual Editor (⌘F4)
- Preview in Window
- ✓ Preview in Viewer Pane (No Preview)
- ✓ Preview Images and Equations
- ✓ Show Previews Inline
- Chunk Output Inline
- ✓ Chunk Output in Console

The right sidebar shows the 'Outline' pane with a list of sections:

- readr
- Sample data
- Importing Data
- Your Turn 1
- Your Turn 1: Bonus
- Tibbles
- Missing values
- Parsing data types
- Your Turn 2
- haven: read and write S...
- Your Turn 3
- Writing data
- Your Turn 4
- Take Aways

The status bar at the bottom shows '6:1' and '# Import Data'.

Project contents

```
|— 01-dplyr_5verbs
|   |— cheatsheet_dplyr_5verbs.pdf
|   |— diabetes.csv
|   |— exercises.qmd
|   |— slides.pdf
```