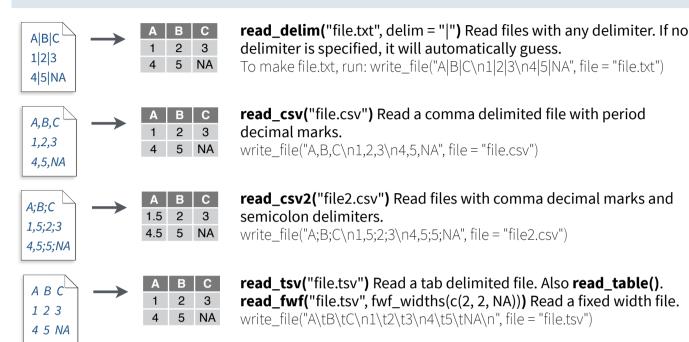
Data Import :: CHEAT SHEET

readr

Read Tabular Data with readr

read_*(file, col_names = TRUE, col_types = NULL, col_select = NULL, id = NULL, locale, n_max = Inf,
 skip = 0, na = c("", "NA"), guess_max = min(1000, n_max), show_col_types = TRUE) See ?read_delim

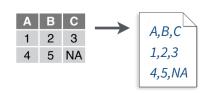


ISEEIII DEAD ADGIIMENTS

USEFUL READ ARGUMENTS								
Α	В	С	No header	1 2 3	Skip lines			
1	2	3	read_csv("file.csv", col_names = FALSE)	4 5 NA	read_csv("file.csv", skip = 1)			
4	5	NA						
Х	у	Z	Provide header	A B C	Read a subset of lines read_csv("file.csv", n_max = 1)			
Α	В	С	read_csv("file.csv",					
1	2	3	col_names = c("x", "y", "z"))	A B C	Read values as missing			
4	5	NA	- · · · · · · · · · · · · · · · · · · ·	NA 2 3	read_csv("file.csv", na = c("1"))			
			Read multiple files into a single table	4 5 NA				
Ц	→		read_csv(c("f1.csv", "f2.csv", "f3.csv"), id = "origin_file")	A;B;C	Specify decimal marks read_delim("file2.csv", locale =			

Save Data with readr

write_*(x, file, na = "NA", append, col_names, quote, escape, eol, num_threads, progress)



write_csv(x, file) Write a comma delimited file.

write_csv2(x, file) Write a semi-colon delimited file.

1,5;2;3,0

locale(decimal_mark = ","))

write_delim(x, file, delim = " ") Write files with any delimiter.

write_tsv(x, file) Write a tab delimited file.

Often one of the first steps of a project is to import outside data into R. Data is often stored in tabular formats, like csv files or spreadsheets.



The front page of this sheet shows how to import and save text files into R using **readr**.



The back page shows how to import spreadsheet data from Excel files using **readxl** or Google Sheets using **googlesheets4**.

OTHER TYPES OF DATA

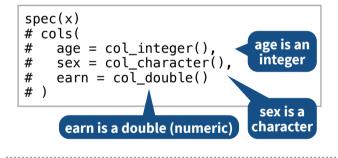
Try one of the following packages to import other types of files:

- haven SPSS, Stata, and SAS files
- DBI databases
- **jsonlite** json
- xml2 XML
- httr Web APIs
- rvest HTML (Web Scraping)
- readr::read_lines() text data

Column Specification with readr

Column specifications define what data type each column of a file will be imported as. By default readr will generate a column spec when a file is read and output a summary.

spec(x**)** Extract the full column specification for the given imported data frame.



COLUMN TYPES

Each column type has a function and corresponding string abbreviation.

- col_logical() "l"
- col_integer() "i"
- col_double() "d"
- col_number() "n"
- col_character() "c"
- col_factor(levels, ordered = FALSE) "f"
- col_datetime(format = "") "T"
- col_date(format = "") "D"
- col_time(format = "") "t"
- col_skip() "-", "_"
- col_guess() "?"

USEFUL COLUMN ARGUMENTS

Hide col spec message

read_*(file, show_col_types = FALSE)

Select columns to import

Use names, position, or selection helpers. read_*(file, col_select = c(age, earn))

Guess column types

To guess a column type, read_*() looks at the first 1000 rows of data. Increase with **guess_max**. read_*(file, guess_max = Inf)

DEFINE COLUMN SPECIFICATION

Set a default type

```
read_csv(
  file,
  col_type = list(.default = col_double())
)
```

Use column type or string abbreviation

```
read_csv(
  file,
    col_type = list(x = col_double(), y = "l", z = "_")
)
```

Use a single string of abbreviations

```
# col types: skip, guess, integer, logical, character
read_csv(
    file,
    col_type = "_?ilc"
)
```



Import Spreadsheets

with readxl

READ EXCEL FILES



read_excel(path, sheet = NULL, range = NULL) Read a .xls or .xlsx file based on the file extension. See front page for more read arguments. Also read_xls() and read_xlsx().

read_excel("excel_file.xlsx")

READ SHEETS



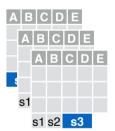
read_excel(path, sheet = **NULL)** Specify which sheet to read by position or name.

read_excel(path, sheet = 1) read_excel(path, sheet = "s1")



excel sheets(path) Get a vector of sheet names.

excel sheets("excel file.xlsx")



To read multiple sheets:

- 1. From a file path, create a vector of sheet names.
- 2. Set the vector names to be the sheet names.
- 3. Use purrr::map_dfr() to read multiple files into one data frame

path <- "your_file_path.xlsx" path %>% excel_sheets() %>% set_names() %>% map_dfr(read_excel, path = path)

For functions to write data to Excel files, see:

writexl - docs.ropensci.org/writexl/

For working with non-tabular Excel data, see:

tidyxl - nacnudus.github.io/tidyxl/

openxlsx - ycphs.github.io/openxlsx/

OTHER USEFUL EXCEL PACKAGES

READXL COLUMN SPECIFICATION

Column specifications define what data type each column of a file will be imported as.

Use the col types argument of read excel() to set the column specification.

Guess all columns

To guess, read_excel() looks at the first 1000 rows of data to guess what type a column is. Increase with the **guess** max argument. read_excel(path, col_types = NULL, $guess_max = Inf)$

Set all columns to same type, e.g. character read excel(path, col types = "text")

Set each column individually

read excel(path, col_types = c("text", "guess", "guess", "numeric")

COLUMN TYPES

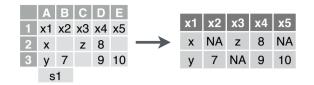
logical	numeric	text	date	list
TRUE	2	hello	1947-01-08	hello
FALSE	3.45	world	1956-10-21	1

- skip
- logical
 - date
- guess
- numeric
 list
- text

Use **list** for columns that include multiple data types. See **tidyr** and **purrr** for list-column data.

with googlesheets4

READ SHEETS



read_sheet(ss, sheet = NULL, range = NULL) Read a sheet from a URL, a Sheet ID, or a dribble from the googledrive package. See front page for more read arguments. Same as range_read().

SHEETS METADATA

URLs are in the form:

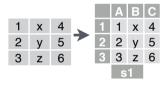
https://docs.google.com/spreadsheets/d/ SPREADSHEET ID/edit#gid=SHEET ID

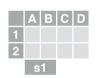
gs4 get(ss) Get spreadsheet meta data.

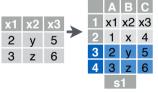
gs4 find(...) Get data on all spreadsheet files.

sheet_properties(ss**)** Get a tibble of properties for each worksheet. Also **sheet_names()**.

WRITE SHEETS







write sheet(data, ss = NULL, sheet = NULL) Write a data frame into a new or existing Sheet.

gs4_create(name, ..., sheets = NULL) Create a new Sheet with a vector of names, a data frame, or a (named) list of data frames.

sheet_append(ss, data, sheet = 1) Add rows to the end of a worksheet.

googlesheets

Column specifications define what data type each column of a file will be imported as.

GOOGLESHEETS4 COLUMN SPECIFICATION

Use the col_types argument of read_sheet()/ range read() to set the column specification.

Guess all columns

To guess, read_sheet()/range_read() looks at the first 1000 rows of data to guess what type a column is. Change with guess max.

read_sheet(path, col_types = NULL, $guess_max = Inf)$

Set all columns to same type, e.g. character read sheet(path, col types = "c")

Set each column individually

I I n I c

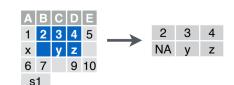
col types: skip, guess, integer, logical, character read sheets(ss, col types = "?ilc")

COLUMN TYPES

TRUE	2	hello	1947-01-08	hello	
FALSE	3.45	world	1956-10-21	1	
skip -guess			date - "D"datetime - "T"		
 logica 	ıl - "l"		character - "c"		
 integer 	er - "i"		list-column - "L"		
 doubl 	e - "d"		cell - "C" Returns		
 nume 	ric - "n"		list of raw cell data.		

Use list for columns that include multiple data types. See **tidyr** and **purrr** for list-column data.

CELL SPECIFICATION FOR READXL AND GOOGLESHEETS4



Use the range argument of readxl::read_excel() or googlesheets4::read_sheet() to read a subset of cells from a sheet.

read_excel(path, range = "Sheet1!D12:F15") read_sheet(ss, range = "D12:F15")

Also use the cell specification functions **cell_limits**, **cell_rows**, cell_cols, and anchored.

FILE LEVEL OPERATIONS

googlesheets4 also offers ways to modify other aspects of Sheets (e.g. freeze rows, set column width, manage (work) sheets. Go to googlesheets4.tidyverse.org to read more.

For whole-file operations (e.g. renaming, sharing, placing within a folder), see the tidyverse package **googledrive** at googledrive.tidyverse.org.

