

Package: html2excel (via r-universe)

July 21, 2026

Type Package

Title Convert 'HTML' Tables to 'Excel' Files

Version 1.0.1

Date 2026-07-12

Description Reads tables from 'HTML' web pages or local documents. The tables are returned as a list of 'tibbles' and may be written to 'Excel' files.

Imports openxlsx, rvest, tools, utils

License GPL (>= 3)

Encoding UTF-8

Depends R (>= 4.1.0)

Suggests testthat (>= 3.0.0)

URL <https://paulnorthrop.github.io/html2excel/>,
<https://github.com/paulnorthrop/html2excel>

BugReports <https://github.com/paulnorthrop/html2excel/issues>

Config/testthat/edition 3

RoxygenNote 7.3.3

NeedsCompilation no

Author Paul Northrop [aut, cre, cph]

Maintainer Paul Northrop <p.northrop@ucl.ac.uk>

Repository <https://cran.r-universe.dev>

Date/Publication 2026-07-21 10:50:12 UTC

RemoteUrl <https://github.com/cran/html2excel>

RemoteRef HEAD

RemoteSha 0e0349e0ea1c93735e6e1cba815e22921475a22c

Contents

html2excel-package	2
html2excel	2
summary.html2excel	5
Index	7

html2excel-package	<i>html2excel: Convert HTML Tables to Excel Files</i>
--------------------	---

Description

Reads tables from HTML web pages or local documents. The tables are returned as a list of tibbles and may be written to Excel files.

Details

The main function is [html2excel](#). See also [summary.html2excel](#).

Author(s)

Maintainer: Paul Northrop <p.northrop@ucl.ac.uk> [copyright holder]

See Also

- Useful links:
- <https://paulnorthrop.github.io/html2excel/>
 - <https://github.com/paulnorthrop/html2excel>
 - Report bugs at <https://github.com/paulnorthrop/html2excel/issues>

html2excel	<i>Convert HTML Tables to Excel Files</i>
------------	---

Description

Reads tables from HTML documents using [rvest::read_html](#) and [rvest::html_table](#). The tables are returned as tibbles and may be written to Excel .xlsx files using [openxlsx::write.xlsx](#).

Usage

```
html2excel(
  html,
  ext = "*.html",
  write = TRUE,
  dir,
  sheets,
  read_args = list(),
  html_args = list(),
  write_args = list()
)
```

Arguments

html	A character vector containing one of the following: • paths to HTML files, • a path to a directory containing HTML files, • URLs, that is, strings beginning with <code>www.</code>, <code>http:</code> or <code>https:</code>.
ext	A character scalar. If <code>html</code> is a path to a directory, then <code>ext</code> determines the extensions of files that should be read from <code>dir</code>. The default is only to read files with an <code>.html</code> extension. <code>utils::glob2rx(ext)</code> is used to create the pattern argument to <code>list.files</code>. For example, if <code>ext = "*.htm*"</code> then all files with extensions that start with <code>htm</code> are read.
write	A logical scalar. Should the tibbles be written to Excel <code>.xlsx</code> files using <code>openxlsx::write.xlsx?</code>
dir	A path to a directory. If <code>write = TRUE</code> then <code>dir</code> determines where the Excel files are written. • If <code>html</code> contains URLs, then <code>dir</code> is an absolute path and must be supplied. To write Excel files to the current working directory use <code>dir = "."</code>. • If <code>html</code> contains paths to HTML files or a directory containing HTML files, then <code>dir</code> is relative to the directory in which an HTML file is located. If <code>dir</code> is not supplied then Excel files are written to the respective directories supplied in <code>html</code>. If <code>dir</code> does not exist then it is created.
sheets	A numeric vector or list of numeric vectors. Component <code>i</code> of the vector or list gives the number(s) of the tables to be included as sheets in the <code>i</code>th output Excel file. This argument is recycled to the number of output Excel files. If <code>sheets</code> is not supplied then all sheets are included. <code>summary.html2excel</code> provides dimensions of the tibbles read.
read_args	A list of arguments for <code>rvest::read_html</code>.
html_args	A list of arguments for <code>rvest::html_table</code>.
write_args	A list of arguments for <code>openxlsx::write.xlsx</code>, but not <code>file</code>, as this is determined from <code>html</code> and <code>dir</code>.

Details

HTML files are

- read using `rvest::read_html`,
- converted to tables (tibbles) using `rvest::html_table`,
- written to Excel documents using `openxlsx::write.xlsx`.

If `rvest::read_html` fails to connect to a URL in `html` then a message is printed and `html` is returned.

The output filename for an input HTML file `file.html` is usually `file.xlsx`. Exceptions are when a combination of `html` and `ext` means that duplicate filenames would be produced, for example, if files `file.html` and `file.mhtml` are read. Then the output filenames are distinguished by `file_a.xlsx` and `file_b.xlsx`, and similarly if there are more than two identical filenames.

The initial motivation for creating the `html2excel` package was to convert to Excel format HTML files that has mistakenly been given a `.xlsx` file extension. If such files are supplied by `html` then each output Excel file will be written to a directory output created in the directory of the respective input file.

Value

An object of class `c("html2excel", "list")`. A list of (lists of) tibbles created from objects returned from `rvest::html_table`. The names of the list are the input HTML filenames. Each list component is named `sheet1`, `sheet2` etc. with each sheet containing a separate table found in the input HTML file. The numbering of these sheets is the same as the output `.xlsx` files, not the order of the tables in the input HTML files. If `write = TRUE` then a character vector containing the file paths to the `.xlsx` files created is added as an attributed named `files`.

See Also

[summary.html2excel](#)

Examples

```
### HTML files

## .html and .mhtml versions of the URL example below
html <- system.file("extdata", "tables.html", package = "html2excel")
mhtml <- system.file("extdata", "tables.mhtml", package = "html2excel")

## .html
# Table 5 only
t1 <- html2excel(html, sheets = 5, write = FALSE)
# The table
t1[[1]]
# The dimensions of the table
summary(t1)

## .mhtml
# Pass UTF-8 encoding to avoid error "Unsupported encoding: 3DUTF-8"
```

```

t2 <- html2excel(mhtml, read_args = list(encoding = "UTF-8"), write = FALSE)
# The same table as above
t2[[1]][5]
# The dimensions of all tables
summary(t2)

### A directory

## Contains the .html and .mhtml files above
directory <- system.file("extdata", package = "html2excel")
# Change the ext argument to include the .mhtml file
# Extract tables 3 and 5 from .html and table 5 from .mhtml
x <- html2excel(directory, ext = ".*.html", sheets = list(c(3, 5), 5),
  read_args = list(encoding = "UTF-8"), write = FALSE)

### A URL
url <- "https://afd.calpoly.edu/web/sample-tables"
x <- html2excel(url, write = FALSE)

```

summary.html2excel	<i>Summarising html2excel objects</i>
--------------------	---------------------------------------

Description

summary method for class html2excel.

Usage

```

## S3 method for class 'html2excel'
summary(object, ...)

## S3 method for class 'summary.html2excel'
print(x, ...)

```

Arguments

object	An object inheriting from class "html2excel" returned from html2excel .
...	Further arguments to be passed to print.default .
x	An object returned by summary.html2excel.

Details

The default print method for a list of tibbles prints the dimensions of the tibbles and up to 10 rows and 6 columns of each tibble. `summary.html2excel` prints a shorter summary that contains only the dimensions of the tibbles. These print and/or summary methods may be helpful in deciding which sheets are required for which excel file, that is, how to set the argument sheets to [html2excel](#).

Value

A list containing the dimensions (number of rows and number of columns) of each tibble.

Examples

See [html2excel](#).

See Also

[html2excel](#).

Index

html2excel, [2](#), [2](#), [5](#), [6](#)

html2excel-package, [2](#)

list.files, [3](#)

openxlsx::write.xlsx, [2-4](#)

print.default, [5](#)

print.summary.html2excel
 (summary.html2excel), [5](#)

rvest::html_table, [2-4](#)

rvest::read_html, [2-4](#)

summary.html2excel, [2-4](#), [5](#)