

# Package ‘srt’

May 9, 2026

**Title** Read Subtitle Files as Tabular Data

**Version** 1.0.4

**Description** Read 'SubRip' <<https://sourceforge.net/projects/subrip/>> subtitle files as data frames for easy text analysis or manipulation. Easily shift numeric timings and export subtitles back into valid 'SubRip' timestamp format to sync subtitles and audio.

**License** GPL-3

**URL** <https://k5cents.github.io/srt/>, <https://github.com/k5cents/srt>

**BugReports** <https://github.com/k5cents/srt/issues>

**Suggests** covr (>= 3.5.1), spelling (>= 2.2), testthat (>= 3.0.0),  
tibble (>= 3.0.4)

**Encoding** UTF-8

**Language** en-US

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Kiernan Nicholls [aut, cre, cph] (ORCID:  
<<https://orcid.org/0000-0002-9229-7897>>)

**Maintainer** Kiernan Nicholls <k5cents@gmail.com>

**Repository** CRAN

**Date/Publication** 2024-03-11 15:40:05 UTC

## Contents

read_srt . . . . .	2
srt_example . . . . .	2
srt_seconds . . . . .	3
srt_shift . . . . .	4
write_srt . . . . .	4

<b>Index</b>	<b>6</b>
--------------	----------

---

read_srt	<i>Read a subtitle file as data frame</i>
----------	---

---

### Description

Convert the SubRip file format to a tabular data frame of times and text.

### Usage

```
read_srt(path, collapse = "\n")
```

### Arguments

path	A path or connection to an .srt file.
collapse	The character with which to separate subtitle lines.

### Details

The SubRip format is a newline-separated, non-tabular text file with groups of subtitle text separated by a newline character and preceded by an index and a timestamp string containing the length of the spoken subtitle text. These components (index, time, text) can be parsed individually and combined into a data frame of subtitle groups.

### Value

A data frame of subtitles.

### Examples

```
# read linear text to tabular data
read_srt(srt_example(), collapse = " ")
```

---

srt_example	<i>Get path to srt example</i>
-------------	--------------------------------

---

### Description

srt comes bundled with a number of sample files in its inst/extdata directory. This function make them easy to access.

### Usage

```
srt_example()
```

### Details

*It's a Wonderful Life* (1946) entered the public domain in 1974.

**Value**

The path or name to a example .srt file.

**Examples**

```
srt_example()
```

---

srt_seconds	<i>Parse components of a subtitle file</i>
-------------	--

---

**Description**

Parse components of a subtitle file

**Usage**

```
srt_seconds(x)
srt_index(x)
srt_text(x, collapse = "\n")
```

**Arguments**

**x** A character vector with the lines of an .srt file.  
**collapse** The character with which to separate subtitle lines.

**Value**

The parsed individual components of a subtitle: integer indexes, numeric times, and collapsed string subtitles.

**Examples**

```
# return individual components of each subtitle
x <- readLines(srt_example())
head(srt_seconds(x)[[1]])
head(srt_index(x))
head(srt_text(x))
```

---

srt_shift	<i>Uniformly shift subtitle times</i>
-----------	---------------------------------------

---

**Description**

Uniformly shift subtitle times

**Usage**

```
srt_shift(x, seconds)
```

**Arguments**

x	A subtitle data frame from <code>read_srt()</code> .
seconds	The number of seconds to shift the start and end time.

**Details**

Here is a workflow of how a linear srt file is shifted in R.

```
read_srt(file) %>%  
  srt_shift(2.1) %>%  
  write_srt(file)
```

**Value**

The numeric start times uniformly shifted by some amount.

**Examples**

```
# shift all start and stop by a some time  
x <- read_srt(srt_example(), collapse = " ")  
srt_shift(x, 1.234)
```

---

write_srt	<i>Write subtitle data frame as SubRip text file</i>
-----------	--

---

**Description**

Write subtitle data frame as SubRip text file

**Usage**

```
write_srt(x, path = NULL, wrap = TRUE, width = 40)
```

**Arguments**

x	A subtitle data frame from <code>read_srt()</code> .
path	File or connection to write to.
wrap	If TRUE (default), subtitle lines will be wrapped.
width	If wrap is TRUE, the width of each wrapped subtitle.

**Details**

The SubRip text files format subtitles with four components separated by a blank line:

1. A numeric counter identifying each sequential subtitle
2. The time that the subtitle should appear on the screen, followed by --> and the time it should disappear
3. Subtitle text itself on one or more lines
4. A blank line containing no text, indicating the end of this subtitle

**Value**

The path to the written file, invisibly.

**Examples**

```
# read and write without line breaks
x <- read_srt(srt_example(), collapse = " ")
write_srt(x, tempfile(fileext = ".srt"), wrap = FALSE)
```

# Index

`read_srt`, 2  
`read_srt()`, 4, 5

`srt_example`, 2  
`srt_index (srt_seconds)`, 3  
`srt_seconds`, 3  
`srt_shift`, 4  
`srt_text (srt_seconds)`, 3

`write_srt`, 4