

Package ‘finlex’

July 23, 2026

Title Access Open Data from the Finnish 'Finlex' Legislative Database

Version 0.2.0

Description Provides functions to retrieve and structure Finnish legislative data made available through the 'Finlex' Open Data API (<<https://www.finlex.fi/en/open-data>>). Functions cover retrieval of statute catalogues, statute titles, structured statute metadata, and cross-references between amending and amended statutes, returned as tidy tibbles for further analysis.

License MIT + file LICENSE

Encoding UTF-8

Language en-GB

URL <https://github.com/KristianVepsalainen/finlex>,
<http://kristianvepsalainen.com/finlex/>

BugReports <https://github.com/KristianVepsalainen/finlex/issues>

Depends R (>= 4.1.0)

Imports dplyr, httr2, tibble, xml2

Suggests knitr, rmarkdown, spelling, testthat (>= 3.0.0), pkgdown

Config/testthat/edition 3

Config/roxygen2/version 8.0.0

VignetteBuilder knitr

NeedsCompilation no

Author Kristian Vepsäläinen [aut, cre]

Maintainer Kristian Vepsäläinen <kristian.vepsalainen@proton.me>

Repository CRAN

Date/Publication 2026-07-23 17:00:08 UTC

Contents

flx_download_statutes	2
flx_get_affected	3
flx_get_consolidated_text	4
flx_get_doc	5
flx_get_metadata	6
flx_get_text	7
flx_get_title	8

Index	9
--------------	----------

flx_download_statutes *Download the catalogue of Finnish statutes*

Description

Retrieves the list of Finnish statutes (Finnish: *säädökset*) published via the Finlex Open Data REST API (<https://www.finlex.fi/en/open-data>), for a given range of years and statute categories. Only the Finnish-language (fin) version of each statute is returned.

Usage

```
flx_download_statutes(
  start_year = 1987,
  end_year = as.integer(format(Sys.Date(), "%Y")),
  categories = c("new-statute", "amending-statute", "repealing-statute"),
  quiet = FALSE
)
```

Arguments

<code>start_year</code>	Integer. First year to include. The Finlex Open Data API currently starts from 1987.
<code>end_year</code>	Integer. Last year to include. Defaults to the current year.
<code>categories</code>	Character vector. One or more of "new-statute", "amending-statute", "repealing-statute". Defaults to all three.
<code>quiet</code>	Logical. If FALSE (the default), prints one progress message per year processed.

Value

A `tibble::tibble()` with one row per statute and the columns:

akn_uri The Akoma Ntoso URI identifying the statute.

year Year of the statute, as an integer.

number Statute number within that year, as an integer.

statute_type One of "NewStatute", "Amendment", "Repeal".

Returns a zero-row tibble with the same columns if nothing matches.

Examples

```
flx_download_statutes(
  start_year = 2023,
  end_year   = 2023,
  categories = "new-statute"
)
```

flx_get_affected	<i>Get statutes affected by an amending or repealing statute</i>
------------------	--

Description

For a given amending or repealing statute, retrieves the statute(s) it affects (the affectedDocument references in the Akoma Ntoso document). A single amending statute can affect more than one target statute (for example, an omnibus act amending several other acts), in which case one row is returned per affected target.

Usage

```
flx_get_affected(year, number)
```

Arguments

year	Integer vector. Year(s) of the amending/repealing statute(s).
number	Integer vector. Number(s) of the statute(s) within that year. Recycled against year if shorter.

Details

This function does not check whether the supplied statute is actually an amendment or repeal; it simply looks for affectedDocument references in whatever statute is requested. New statutes will typically return status = "no_affected".

Value

A `tibble::tibble()` with the columns:

source_year Year of the amending/repealing statute, as supplied.

source_number Number of the amending/repealing statute, as supplied.

status One of "ok" (at least one affected statute found), "no_affected" (statute found, but no affected-document references present), "not_found", or "error".

target_href URI of the affected statute, or NA.

target_year Year of the affected statute, or NA.

target_number Number of the affected statute, or NA.

Examples

```
# Look up affected statutes for a known amending statute
flx_download_statutes(
  start_year = 2023, end_year = 2023,
  categories = "amending-statute"
) |>
  head(1) |>
  with(flx_get_affected(year, number))
```

```
flx_get_consolidated_text
```

Get the consolidated (up-to-date) text of one or more Finnish statutes

Description

Retrieves the *consolidated* version of a statute – the text as currently in force, with all amendments merged in – split into one row per section (Finnish: *pykala*). This complements `flx_get_text()`, which returns the statute’s *original*, as-enacted text.

Usage

```
flx_get_consolidated_text(year, number)
```

Arguments

year	Integer vector. Year(s) of the statute(s) as originally enacted. The consolidated version shares the same year/number as the original statute – it is a different rendering of the same legal work, not a separate document.
number	Integer vector. Number(s) of the statute(s) within that year. Recycled against year if shorter.

Details

Not every statute has a consolidated version available – for example, very recent statutes that have never been amended, or statutes outside the API’s consolidation coverage, will return `status = "not_found"`.

Value

A tibble with one row per section (or one row with NA section fields if the statute has no sections, has no consolidated version, was not found, or an error occurred) and the columns: `year`, `number`, `status` (one of "ok", "no_sections", "not_found", or "error"), `section_index`, `section_id`, `heading`, and `text`.

Examples

```
flx_get_consolidated_text(year = 1992, number = 1535) # Tuloverolaki
```

flx_get_doc

*Get a Finlex document by type, year, and number***Description**

Retrieves a document from the generic document endpoint of the Finlex Open Data API (`/akn/fi/doc/...`), which covers document types that are not part of the Statute Book of Finland: treaties, government proposals, authority regulations, and certain collective agreements.

Usage

```
flx_get_doc(doc_type, year, number)
```

Arguments

<code>doc_type</code>	Character, length 1. One of "authority-regulation", "treaty", "tax-treaty-consolidated", "government-proposal", "collective-agreement-general-applicability-decision", or "trade-union-center-agreement". Call this function separately for different document types.
<code>year</code>	Integer vector. Year(s) of the document(s).
<code>number</code>	Character or integer vector. Number(s) of the document(s) within that year. Recycled against year if shorter. Accepts character because document numbers in these collections are not always simple integers.

Details

Unlike `flx_get_text()`, this function does not split the document into sections. The document types covered here (treaties, government proposals, agreements) do not share a consistent internal structure the way statutes do, so the full body text is returned as a single block.

Some documents, particularly older ones, have never been digitised as machine-readable text and exist only as a scanned or original PDF. For these, `status` is "pdf_only", `text` is NA, and `pdf_url` points to the PDF so the content is still reachable.

Two document types available in the underlying API are intentionally *not* covered by this function: "treaty-metadata" (a lighter reference record rather than a full document) and "legal-literature-references" (an index of literature discussing case law, not the case law itself). Both likely need bespoke handling and may be added in a future version.

Value

A `tibble::tibble()` with one row per document and the columns:

doc_type As supplied.

year As supplied.

number As supplied.

status One of "ok", "pdf_only" (the document exists but has no machine-readable text, only a scanned/original PDF), "not_found", or "error".

title The document's official title.

date_issued Date the document was issued, as a [Date](#).

text The full body text of the document, or NA if only a PDF is available.

pdf_url URL of the document's PDF version. Populated for both "ok" and "pdf_only" statuses.

Examples

```
flx_get_doc(
  doc_type = "authority-regulation",
  year = 1996, number = 32082
)
```

flx_get_metadata	<i>Get structured metadata for one or more Finnish statutes</i>
------------------	---

Description

Retrieves core bibliographic metadata for one or more Finnish statutes from the Finlex Open Data API: the date of issue, the official title, and the number of sections (Finnish: *pykälät*) in the statute text.

Usage

```
flx_get_metadata(year, number)
```

Arguments

year	Integer vector. Year(s) of the statute(s).
number	Integer vector. Number(s) of the statute(s) within that year. Recycled against year if shorter.

Details

This function intentionally returns only raw, directly observed fields. Any derived classification (for example, grouping statutes into categories such as "Act", "Decree", or "Decision" based on title patterns) is left to a separate function, since that involves methodological choices rather than data retrieval.

Value

A `tibble::tibble()` with one row per (year, number) pair and the columns:

year As supplied.

number As supplied.

status One of "ok", "not_found", or "error".

date_issued Date the statute was issued, as a [Date](#).

title The statute's official title.

n_sections Integer. Number of section elements found in the Akoma Ntoso document — a rough, purely structural proxy for statute length.

Examples

```
flx_get_metadata(year = 1992, number = 1535) # Tuloverolaki
```

flx_get_text

Get the full text of one or more Finnish statutes, by section

Description

Retrieves the body text of a Finnish statute from the Finlex Open Data API, split into one row per section (*pykälä*). This complements [flx_get_metadata\(\)](#), which only reports the *number* of sections; this function returns their actual content.

Usage

```
flx_get_text(year, number)
```

Arguments

year	Integer vector. Year(s) of the statute(s).
number	Integer vector. Number(s) of the statute(s) within that year. Recycled against year if shorter.

Details

Statutes can be long. Calling this function for many statutes at once will be slow and will retrieve a large amount of text — consider narrowing down to the statutes you actually need (for example with [flx_download_statutes\(\)](#) or [flx_get_metadata\(\)](#)) before calling this function.

Value

A [tibble::tibble\(\)](#) with one row per section (or one row with NA section fields if the statute has no sections, was not found, or an error occurred) and the columns:

year Year of the statute, as supplied.

number Number of the statute, as supplied.

status One of "ok", "no_sections" (statute found, but contains no section elements — for example a very short decision), "not_found", or "error".

section_index Integer. The section's position within the statute (1-based), in document order.

section_id Character. The section's Akoma Ntoso eId attribute, if present (for example "sec_1").

heading Character. The section heading, if present.

text Character. The full text of the section, including its heading.

Examples

```
flx_get_text(year = 1992, number = 1535) # Tuloverolaki, all sections
```

```
flx_get_title
```

Get the title of one or more Finnish statutes

Description

Retrieves the official title of one or more Finnish statutes from the Finlex Open Data API, given each statute's year and number.

Usage

```
flx_get_title(year, number)
```

Arguments

year	Integer vector. Year(s) of the statute(s).
number	Integer vector. Number(s) of the statute(s) within that year. Recycled against year if shorter.

Details

Not every statute in Finnish legal history is available through the Finlex Open Data API (very old statutes in particular are not covered). When a statute cannot be found, **status** is set to "not_found" rather than raising an error, so that vectorized calls complete even when some statutes are missing.

Value

A `tibble::tibble()` with one row per (year, number) pair and the columns:

year As supplied.

number As supplied.

status One of "ok", "not_found", or "error".

title The statute's official title, or NA if unavailable.

Examples

```
flx_get_title(year = 1992, number = 1535) # Tuloverolaki
flx_get_title(year = c(1992, 1993), number = c(1535, 1501))
```


Index

Date, [6](#), [7](#)

flx_download_statutes, [2](#)

flx_download_statutes(), [7](#)

flx_get_affected, [3](#)

flx_get_consolidated_text, [4](#)

flx_get_doc, [5](#)

flx_get_metadata, [6](#)

flx_get_metadata(), [7](#)

flx_get_text, [7](#)

flx_get_text(), [4](#), [5](#)

flx_get_title, [8](#)

tibble::tibble(), [2](#), [3](#), [5–8](#)