

Package ‘edgarfundamentals’

March 26, 2026

Title Retrieve Fundamental Financial Data from SEC 'EDGAR'

Version 0.1.1

Description Provides a simple, ticker-based interface for retrieving fundamental financial data from the United States Securities and Exchange Commission's 'EDGAR' 'XBRL' API <<https://www.sec.gov/edgar/sec-api-documentation>>. Functions return key financial ratios including earnings per share, return on equity, return on assets, debt-to-equity, current ratio, gross margin, operating margin, net margin, price-to-earnings, price-to-book, and dividend yield for any publicly traded U.S. company. Data is sourced directly from company 10-K annual filings, requiring no API key or paid subscription. Designed for use in quantitative finance courses and research workflows.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Imports dplyr (>= 1.1.0), httr (>= 1.4.0), jsonlite (>= 1.8.0), rlang, tidyquant (>= 1.0.0)

Suggests knitr, rmarkdown

VignetteBuilder knitr

URL <https://github.com/robschumaker/edgarfundamentals>

BugReports <https://github.com/robschumaker/edgarfundamentals/issues>

NeedsCompilation no

Author Robert Schumaker [aut, cre] (ORCID:
<<https://orcid.org/0000-0003-2977-1856>>)

Maintainer Robert Schumaker <rob.schumaker@gmail.com>

Depends R (>= 4.1.0)

Repository CRAN

Date/Publication 2026-03-26 10:30:09 UTC

Contents

get_cik	2
get_filing_history	3
get_fundamentals	4
get_fundamentals_batch	6
Index	8

get_cik	<i>Look up a company's SEC Central Index Key from its ticker symbol</i>
---------	---

Description

Translates a stock ticker symbol into the corresponding SEC EDGAR Central Index Key (CIK). The CIK is a unique numerical identifier assigned by the SEC to every company that files with EDGAR. It is required for all subsequent EDGAR API calls.

Usage

```
get_cik(symbol)
```

Arguments

symbol	A character string containing the stock ticker symbol (e.g. "AAPL"). Case-insensitive.
--------	--

Details

Data is retrieved from https://www.sec.gov/files/company_tickers.json, a publicly maintained mapping file updated by the SEC. No API key is required.

The SEC requests that automated tools identify themselves via a User-Agent header. Set your identifier once per session with: `options(edgarfundamentals.user_agent = "Your Name your@email.com")`

Value

A character string containing the CIK number (without zero-padding).

Examples

```
## Not run:
options(edgarfundamentals.user_agent = "Jane Smith jane@example.com")
get_cik("AAPL")
get_cik("LLY")

## End(Not run)
```

get_filing_history *Retrieve recent SEC filing history for a company*

Description

Returns a data frame listing the most recent EDGAR filings of a specified form type for a given company. Useful for verifying that a company has filed the expected number of 10-K or 10-Q reports, or for retrieving accession numbers needed to access specific filings.

Usage

```
get_filing_history(symbol, form_type = "10-K", n = 5)
```

Arguments

symbol	A character string containing the stock ticker symbol (e.g. "AAPL"). Case-insensitive.
form_type	A character string specifying the SEC form type to retrieve. Common values are "10-K" (annual report), "10-Q" (quarterly report), and "8-K" (current report). Defaults to "10-K".
n	An integer specifying the maximum number of filings to return. Defaults to 5.

Details

Data is retrieved from the SEC EDGAR submissions API (<https://data.sec.gov/submissions/>). The API returns up to the 1,000 most recent filings across all form types; older filings may not appear.

Set your User-Agent once per session: `options(edgarfundamentals.user_agent = "Your Name your@email.com")`

Value

A data frame with one row per filing and the following columns:

symbol The ticker symbol passed to the function.

accession_number The SEC accession number uniquely identifying the filing (e.g. "0000320193-24-000123").

filing_date The date the filing was submitted to EDGAR.

report_date The period-end date covered by the filing.

form The form type as recorded in EDGAR.

primary_document The filename of the primary HTML document within the filing.

Examples

```
## Not run:
options(edgarfundamentals.user_agent = "Jane Smith jane@example.com")

# Five most recent annual reports for Lockheed Martin
get_filing_history("LMT", form_type = "10-K", n = 5)

# Most recent quarterly reports for Eli Lilly
get_filing_history("LLY", form_type = "10-Q", n = 4)

## End(Not run)
```

get_fundamentals	<i>Retrieve key fundamental financial ratios for a single stock</i>
------------------	---

Description

Pulls fundamental financial data directly from a company's most recent annual 10-K filing in SEC EDGAR and computes key financial ratios. No API key or paid subscription is required.

Usage

```
get_fundamentals(symbol, to_date = as.character(Sys.Date()))
```

Arguments

symbol	A character string containing the stock ticker symbol (e.g. "LLY"). Case-insensitive.
to_date	A character string in "YYYY-MM-DD" format. The function returns ratios from the most recent 10-K with a period end date on or before this date. Defaults to today's date.

Details

Financial statement values are extracted from XBRL-tagged 10-K filings via the SEC EDGAR companyfacts API (<https://data.sec.gov/api/xbrl/companyfacts/>). Because data comes from annual filings, ratios reflect the most recently completed fiscal year ending on or before to_date, not real-time values.

Fallback XBRL tags are attempted automatically when a company uses a non-standard tag name for a concept. A courtesy pause of 0.5 seconds is inserted after the companyfacts API call to respect the SEC's rate limit of 10 requests per second.

Set your User-Agent once per session: options(edgarfundamentals.user_agent = "Your Name your@email.com")

Value

A named numeric vector with the following elements:

CIK SEC Central Index Key – the company’s unique EDGAR identifier.

EPS Diluted Earnings Per Share (USD per share) from the most recent qualifying 10-K. Falls back to basic EPS if diluted is unavailable.

NetIncome Net income attributable to the company (USD).

Revenue Total revenue (USD).

ROE Return on Equity as a percentage ($\text{NetIncome} / \text{StockholdersEquity} * 100$). A measure of how efficiently the company generates profit from shareholder capital.

ROA Return on Assets as a percentage ($\text{NetIncome} / \text{TotalAssets} * 100$). A measure of how efficiently the company uses its assets.

DE Debt-to-Equity ratio ($\text{LongTermDebt} / \text{StockholdersEquity}$). A measure of financial leverage.

CurrentRatio Current Assets divided by Current Liabilities. A measure of short-term liquidity; values above 1 indicate the company can cover near-term obligations.

GrossMargin Gross Profit as a percentage of Revenue. Measures pricing power and production efficiency.

OperatingMargin Operating Income as a percentage of Revenue. Measures operational efficiency before interest and taxes.

NetMargin Net Income as a percentage of Revenue. Bottom-line profitability after all expenses.

PE Price-to-Earnings ratio, computed by dividing the most recent adjusted closing price (from Yahoo Finance via tidyquant) by EPS. Returns NA if EPS is zero or negative.

PB Price-to-Book ratio, computed by dividing the most recent adjusted closing price by Book Value per Share ($\text{StockholdersEquity} / \text{SharesOutstanding}$). Returns NA if shares outstanding is zero or unavailable.

DIV Dividend Yield as a percentage ($\text{DividendsPerShareDeclared} / \text{Price} * 100$). Returns NA if no dividend data is available.

Examples

```
## Not run:
options(edgarfundamentals.user_agent = "Jane Smith jane@example.com")

# Fundamentals for Eli Lilly as of end of 2024
get_fundamentals("LLY", "2024-12-31")

# Fundamentals as of today
get_fundamentals("JNJ")

## End(Not run)
```

`get_fundamentals_batch`*Retrieve fundamental financial ratios for a portfolio of stocks*

Description

A vectorized wrapper around [get_fundamentals](#) that accepts a vector of ticker symbols and returns a tidy data frame with one row per stock. Failed lookups are recorded as NA rather than stopping execution, so a single problematic ticker does not interrupt the batch.

Usage

```
get_fundamentals_batch(symbols, to_date = as.character(Sys.Date()))
```

Arguments

<code>symbols</code>	A character vector of stock ticker symbols (e.g. <code>c("LLY", "PFE", "UNH")</code>). Case-insensitive.
<code>to_date</code>	A character string in "YYYY-MM-DD" format. Passed to get_fundamentals for each symbol. Defaults to today's date.

Details

Each symbol requires two SEC EDGAR API calls (one for the CIK lookup and one for the companyfacts data) plus one Yahoo Finance call for the current price. A 0.5-second pause is inserted after each companyfacts call to respect the SEC rate limit of 10 requests per second. For a portfolio of 14 stocks, expect a total retrieval time of approximately 20–30 seconds.

Set your User-Agent once per session: `options(edgarfundamentals.user_agent = "Your Name your@email.com")`

Value

A data frame with one row per symbol and the following columns: `symbol`, `CIK`, `EPS`, `NetIncome`, `Revenue`, `ROE`, `ROA`, `DE`, `CurrentRatio`, `GrossMargin`, `OperatingMargin`, `NetMargin`, `PE`, `PB`, `DIV`. See [get_fundamentals](#) for definitions. Rows where data retrieval failed contain NA for all ratio columns.

See Also

[get_fundamentals](#) for single-stock retrieval.

Examples

```
## Not run:
options(edgarfundamentals.user_agent = "Jane Smith jane@example.com")

healthcare <- c("UNH", "PFE", "MRK", "ABT", "LLY", "CVS", "AMGN")
```

get_fundamentals_batch

7

```
get_fundamentals_batch(healthcare, "2024-12-31")

defense <- c("LMT", "RTX", "NOC", "GD", "HII", "LHX", "LDOS")
get_fundamentals_batch(defense, "2024-12-31")

## End(Not run)
```

Index

`get_cik`, [2](#)
`get_filing_history`, [3](#)
`get_fundamentals`, [4](#), [6](#)
`get_fundamentals_batch`, [6](#)