

# Package: GetDFPData2 (via r-universe)

October 27, 2026

**Title** Reading Annual and Quarterly Financial Reports from B3

**Version** 0.6.5

**Description** Reads annual and quarterly financial reports from companies traded at B3, the Brazilian exchange  
<<https://www.b3.com.br/>>. All data is downloaded and imported from CVM's public ftp site  
<[https://dados.cvm.gov.br/dados/CIA\\_ABERTA/](https://dados.cvm.gov.br/dados/CIA_ABERTA/)>.

**Depends** R (>= 3.5.0)

**Imports** stringr, xml2, dplyr, readr, lubridate, magrittr, purrr, memoise, rvest, tidyr, RCurl, shiny, writexl, stats, cli, glue, janitor

**ByteCompile** true

**License** GPL-2

**BugReports** <https://github.com/msperlin/GetDFPData2/issues/>

**URL** <https://github.com/msperlin/GetDFPData2/>

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 8.0.0

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), ggplot2, covr, fs

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Config/pak/sysreqs** cmake make libicu-dev libuv1-dev libxml2-dev libssl-dev libx11-dev zlib1g-dev

**Repository** <https://msperlin.r-universe.dev>

**Date/Publication** 2026-06-04 13:20:14 UTC

**RemoteUrl** <https://github.com/msperlin/getdfpdata2>

**RemoteRef** HEAD

**RemoteSha** 58633789ea9a9a68594991858715d313e09eaa75

Contents

|                              |          |
|------------------------------|----------|
| export_xlsx . . . . .        | 2        |
| get_dfp_data . . . . .       | 3        |
| get_info_companies . . . . . | 4        |
| get_tickers . . . . .        | 5        |
| search_company . . . . .     | 5        |
| <b>Index</b>                 | <b>7</b> |

---

|             |                                           |
|-------------|-------------------------------------------|
| export_xlsx | <i>Export DFP/ITR data to a xlsx file</i> |
|-------------|-------------------------------------------|

---

Description

Export DFP/ITR data to a xlsx file

Usage

```
export_xlsx(l_dfp, f_xlsx = "GetDFPData-XLSX.xlsx")
```

Arguments

- l\_dfp            A list from get\_dfp\_data or get\_itr\_data
- f\_xlsx          Path to xlsx file

Value

A dataframe with several information about B3 companies

Examples

```
## Not run: # keep cran check fast
df_info <- get_info_companies()
str(df_info)

## End(Not run)
```

get\_dfp\_data

*Downloads and reads DFP datasets***Description**

The DFP (demonstrativos financeiros padronizados) is the annual reporting system of companies traded at B3. This function will access the CVM ftp and parse all available files according to user choices

**Usage**

```
get_dfp_data(
  companies_cvm_codes = NULL,
  first_year = 2010,
  last_year = lubridate::year(Sys.Date()),
  type_docs = c("BPA", "BPP", "DRE"),
  type_format = c("con", "ind"),
  clean_data = TRUE,
  use_memoise = FALSE,
  cache_folder = "gdfpd2_cache",
  do_shiny_progress = FALSE
)
```

**Arguments**

|                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| companies_cvm_codes | Numeric CVM code of companies. IF set to NULL (default), will return data for all available companies.                                                                                                                                                                                                                                                                                                                                                                                                  |
| first_year          | First year of selected data                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| last_year           | Last year of selected data                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| type_docs           | Type of financial documents. E.g. c('DRE', 'BPA'). Definitions: '*' = fetch all docs, 'BPA' = Assets (ativos), 'BPP' = Liabilities (passivo), 'DRE' = income statement (demonstrativo de resultados), 'DFC_MD' = cash flow by direct method (fluxo de caixa pelo metodo direto), 'DFC_MI' = cash flow by indirect method (fluxo de caixa pelo metodo indireto), 'DMPL' = statement of changes in equity (mutacoes do patrimonio liquido), 'DVA' = value added report (desmonstrativo de valor agregado) |
| type_format         | Type of format of document (con = consolidated, ind = individual). Default = c('con', 'ind')                                                                                                                                                                                                                                                                                                                                                                                                            |
| clean_data          | Clean data or return raw data? See read_dfplitr_csv() for details                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| use_memoise         | Use memoise caching? If no (default), the function will read all .csv files. If yes, will use package memoise for caching results (execution speed increases significantly)                                                                                                                                                                                                                                                                                                                             |
| cache_folder        | Path of cache folder to keep memoise and zip files                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| do_shiny_progress   | Whether to use shiny progress indicator (default = FALSE)                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Value**

A list of tibbles containing all requested financial data. Each element of the list is a table from DFP.

**Examples**

```
## Not run:  
df_dfp <- get_dfp_data(companies_cvm_codes = NULL)  
  
## End(Not run)
```

---

|                    |                                                                       |
|--------------------|-----------------------------------------------------------------------|
| get_info_companies | <i>Returns a up to date table with information about B3 companies</i> |
|--------------------|-----------------------------------------------------------------------|

---

**Description**

Imports a dataset from cvm <<https://dados.cvm.gov.br/>>, containing up to date information about companies, active or not.

**Usage**

```
get_info_companies(cache_folder = "gdfpd2_cache")
```

**Arguments**

cache\_folder      Path of cache folder to keep memoise and zip files

**Value**

A dataframe with information about available companies in the database

**Examples**

```
## Not run: # keep cran check fast  
df_info <- get_info_companies()  
str(df_info)  
  
## End(Not run)
```

---

|             |                                                              |
|-------------|--------------------------------------------------------------|
| get_tickers | <i>Retrieves a ticker list from cvm open data repository</i> |
|-------------|--------------------------------------------------------------|

---

**Description**

Retrieves ticker information from <[https://dados.cvm.gov.br/dados/CIA\\_ABERTA/DOC/FCA/DADOS/](https://dados.cvm.gov.br/dados/CIA_ABERTA/DOC/FCA/DADOS/)>.

**Usage**

```
get_tickers(year = lubridate::year(Sys.Date()))
```

**Arguments**

year                      year of the data (2018 onwards)

**Value**

a dataframe with information about companies and tickers

**Examples**

```
## Not run:  
tickers <- get_tickers()  
  
## End(Not run)
```

---

|                |                                         |
|----------------|-----------------------------------------|
| search_company | <i>Helps users search for a company</i> |
|----------------|-----------------------------------------|

---

**Description**

Helps users search for a company

**Usage**

```
search_company(char_to_search, cache_folder = "gdfpd2_cache")
```

**Arguments**

char\_to\_search    Character for partial matching  
cache\_folder      Path of cache folder to keep memoise and zip files

**Value**

A dataframe with information about available companies in the CVM database

**Examples**

```
## Not run: # dontrun: keep cran check fast  
df <- search_company('petrobras')  
  
## End(Not run)
```

# Index

`export_xlsx`, [2](#)

`get_dfp_data`, [3](#)

`get_info_companies`, [4](#)

`get_tickers`, [5](#)

`search_company`, [5](#)