# Package 'rprime'

October 14, 2022

**Title** Functions for Working with 'Eprime' Text Files

**Version** 0.1.2

**Description** 'Eprime' is a set of programs for administering
psychological experiments by computer. This package provides functions
for loading, parsing, filtering and exporting data in the text files
produced by 'Eprime' experiments.

**License** GPL-2

**URL** <https://github.com/tjmahr/rprime>

**BugReports** <https://github.com/tjmahr/rprime/issues>

**Depends** R (>= 3.0.1)

**Imports** assertthat, plyr, stringi, stringr (>= 1.0.0), tools, utils

**Suggests** covr, knitr, readr, rmarkdown, testthat

**VignetteBuilder** knitr

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Tristan Mahr [aut, cre] (<<https://orcid.org/0000-0002-8890-5116>>)

**Maintainer** Tristan Mahr <tristan.mahr@wisc.edu>

**Repository** CRAN

**Date/Publication** 2020-09-24 11:20:02 UTC

## R topics documented:

**Index**                                                                                               **10**

---

  as.EprimeFrame                    *Convert a list into an EprimeFrame object*

---

### Description

Convert a list into an EprimeFrame object

### Usage

```
as.EprimeFrame(xs)
```

### Arguments

xs              a list

### Value

the original list as an `EprimeFrame` object (along with dummy Eprime metadata fields)

---

  as.FrameList                    *Convert a list of EprimeFrames into a FrameList object*

---

### Description

Convert a list of EprimeFrames into a FrameList object

### Usage

```
as.FrameList(xs)
```

### Arguments

xs              a list of EprimeFrames

### Value

the original list as a `FrameList` object

---

EprimeFrame                    *Create an EprimeFrame object*

---

### Description

This constructor function converts a character vector into an `EprimeFrame` object, which is just a list with some special metadata values. Strings with the format `"key: value"` are parsed into key = value list items (via `listify`).

### Usage

```
EprimeFrame(keys_values)
```

### Arguments

keys_values        a character vector of containing some `"key: value"` strings.

### Value

a list with the class `EprimeFrame` and with special `Eprime.` metadata, `Running` and `Procedure` values, all set to NA by default.

### Examples

```
# Default metadata values
lines <- c(
  "key: value",
  "question: answer",
  "garbage text")

EprimeFrame(lines)
# List of 8
# $ Eprime.Level      : num 1
# $ Eprime.LevelName  : logi NA
# $ Eprime.Basename   : logi NA
# $ Eprime.FrameNumber: logi NA
# $ Procedure         : logi NA
# $ Running           : logi NA
# $ key               : chr "value"
# $ question          : chr "answer"

# Normalize [Running] related lines
keys_values <- c(
  "Running: Demo",
  "Demo: ExampleCode",
  "Demo.Cycle: 1",
  "Demo.Sample: 1",
  "Key: Value")
```

```
EprimeFrame(keys_values)
# List of 9
# $ Eprime.Level      : num 1
# $ Eprime.LevelName  : chr "Demo_ExampleCode"
# $ Eprime.Basename   : logi NA
# $ Eprime.FrameNumber: logi NA
# $ Procedure         : logi NA
# $ Running           : chr "Demo"
# $ Cycle             : chr "1"
# $ Sample            : chr "1"
# $ Key               : chr "Value"
```

---

extract_chunks                    *Extract log-frames from an Eprime log file*

---

#### Description

Almost all of the information in an Eprime file comes in chunks of text bracketed by the lines ***
LogFrame Start *** and *** LogFrame End ***. The exception is the header information which
is bracketed by *** Header Start *** and *** Header End ***.

#### Usage

```
extract_chunks(eprime_log)
```

#### Arguments

eprime_log        a character vector containing the lines of text from Eprime txt file

#### Details

extract_chunks extracts the bracketed text, storing each log-frame of text in a list. The lists also
include some additional lines of text as metadata: Eprime.FrameNumber and Eprime.Basename
(the name of the source file). The header log-frame also gets dummy lines: Procedure: Header
and Running: Header.

These character vectors of colon-separated lines are converted into proper lists by FrameList(...).

#### Value

a list of character vectors, where each vector contains the lines of a log-frame

---

| filter_in | *Filter levels in or out of a FrameList based on attribute values* |
|---|---|

---

### Description

Filter levels in or out of a FrameList based on attribute values

### Usage

```
filter_in(frame_list, key, values)

filter_out(frame_list, key, values)
```

### Arguments

| | |
|---|---|
| frame_list | a list of `EprimeFrame` objects |
| key | the name of the attribute to filter in or out |
| values | the whitelisted or blacklisted values of the attribute |

### Value

for `filter_in`, only log-frames where key is one of the `values` are kept. for `filter_out`, log-frames where key is one of the `values` are omitted.

---

| FrameList | *Convert lines from an Eprime file into EprimeFrame objects* |
|---|---|

---

### Description

Convert character vectors of implicit key-value pairs (e.g., `c("key1: value1", "key2: value2")`), into lists of explicit key-value pairs, `list(key1 = "value1", key2 = "value2")`.

### Usage

```
FrameList(x)
```

### Arguments

| | |
|---|---|
| x | a character vector with lines of the form `"key: value"`, or a list of vectors of colon-separated text |

### Details

During the conversion, if `Running: x`, then the `x.Sample` and `x.Cycle` lines are simplified into `Sample` and `Cycle` lines. The `x: value` line is recoded as `Eprime.LevelName: x_value`. The purpose of this tidying is to force the same set of key names (eventually, column names) onto frames with different values for "Running".

## Value

When passed a list of character vectors of "key: value" lines, a FrameList object (a list of Eprime-Frames) is returned. when passed a single vector vector of "key: value" lines, a single Eprime-Frame object is returned inside of a FrameList object.

## Examples

```
lines <- c("\t*** LogFrame Start ***",
           "\tProcedure: FamTask",
           "\titem1: bear",
           "\titem2: chair",
           "\tCorrectResponse: bear",
           "\tImageSide: Left",
           "\tDuration: 885",
           "\tFamiliarization: 1",
           "\tFamInforcer: 1",
           "\tReinforcerImage: Bicycle1",
           "\tFamiliarization.Cycle: 1",
           "\tFamiliarization.Sample: 1",
           "\tRunning: Familiarization",
           "\tFamTarget.RESP: ",
           "\tCorrect: True",
           "\t*** LogFrame End ***")
# List of 1
# $ :List of 17
# ..$ Eprime.Level      : num 2
# ..$ Eprime.LevelName  : chr "Familiarization_1"
# ..$ Eprime.Basename   : chr "NA"
# ..$ Eprime.FrameNumber: chr "1"
# ..$ Procedure         : chr "FamTask"
# ..$ Running           : chr "Familiarization"
# ..$ item1             : chr "bear"
# ..$ item2             : chr "chair"
# ..$ CorrectResponse   : chr "bear"
# ..$ ImageSide         : chr "Left"
# ..$ Duration          : chr "885"
# ..$ FamInforcer       : chr "1"
# ..$ ReinforcerImage   : chr "Bicycle1"
# ..$ Cycle             : chr "1"
# ..$ Sample            : chr "1"
# ..$ FamTarget.RESP    : chr ""
# ..$ Correct           : chr "True"
# ..- attr(*, "class")= chr [1:2] "EprimeFrame" "list"
# - attr(*, "class")= chr [1:2] "list" "FrameList"
```

---

keep_levels                     *Filter levels in or out of a FrameList based on Eprime.Level values*

---

### Description

These functions are shortcuts for calls to `filter_in` or `filter_out`.

### Usage

```
keep_levels(frame_list, level_numbers)

drop_levels(frame_list, level_numbers)
```

### Arguments

`frame_list`     a list of `EprimeFrame` objects

`level_numbers`   the whitelisted or blacklisted values for Eprime.Level

### Details

Note that the meaning of Eprime.Level value in a log-frame ultimately is equal to one plus the number of tabs before each line in the log-frame.

### Value

for `keep_levels`, only log-frames where the level matches one of the `level_numbers` are kept. for `keep_levels`, log-frames where the level matches one of the `level_numbers` are omitted.

---

| listify | *Convert a vector of colon-separated text lines into a list of named elements* |
|---|---|

---

### Description

Convert a vector of colon-separated text lines into a list of named elements

### Usage

```
listify(colon_sep_xs)
```

### Arguments

`colon_sep_xs`    a character vector with lines of the form `"key: value"`

### Details

Some minor cleaning of the input is performed:

- Lines without a colon-space separator `": "` are filtered out.
- Once the strings are split at the separator, white-space on the left and right sides of each half-string is omitted.

**Value**

a named list of the values in the colon-separated lines. `"key: value"` yields `list(key = "value")`

---

preview_eprime                   *Preview the levels in a parsed Eprime file*

---

**Description**

Preview the levels in a parsed Eprime file

**Usage**

```
preview_eprime(frame_list)

preview_levels(frame_list)

preview_frames(frame_list)
```

**Arguments**

frame_list       a FrameList (a list of EprimeFrames)

**Details**

`preview_levels` prints out the unique combinations of Eprime.Level number, Procedure, and Running in the frame list. `preview_frames` prints out example frame from each of the unique levels. `preview_eprime` does both.

**Value**

Nothing. Preview text is printed to the console.

---

read_eprime                      *Read in a text file generated by Eprime*

---

**Description**

Read in a text file generated by Eprime

**Usage**

```
read_eprime(filename, remove_clock = TRUE)
```

### Arguments

| | |
|---|---|
| `filename` | Either the full or relative path to an Eprime .txt file |
| `remove_clock` | Whether to exclude the Clock.Information XML entries. Enabled by default. |

### Details

The encoding on an Eprime txt file should be UCS-2 Little Endian, but sometimes this is not the case. We delegate the fussy encoding details to the `stringi::str_read_lines` function.

If the file is not an Eprime txt–that is, if it is missing the lines `*** Header Start ***` and `*** Header End ***`–a warning is raised and the lines of text are replaced by a dummy header.

### Value

Each line of the file is stored and returned in a character vector.

---

| to_data_frame | *Convert Eprime Frames into data-frames* |
|---|---|

---

### Description

Convert Eprime Frames into data-frames

### Usage

```
to_data_frame(x)
```

### Arguments

| | |
|---|---|
| `x` | an EprimeFrame object, or a FrameList object (a list of EprimeFrames) |

### Details

Individual EprimeFrames are converted to a data-frame using `as.data.frame`. (Strings are not converted to factors.)

Each of the individual data-frames are then `rbinded` together, with missing columns being filled with NA.

### Value

all of the EprimeFrames combined into a single data frame.

### See Also

[rbind.fill](rbind.fill)

# Index