

Package ‘exams.mylearn’

May 25, 2020

Title Question Generation in the 'MyLearn' XML Format

Version 1.1

Description Randomized multiple-select and single-select question generation for the 'MyLearn' teaching and learning platform. Question templates in the form of the R/exams package (see <<http://www.r-exams.org/>>) are transformed into XML format required by 'MyLearn'.

License GPL-3

Depends R (>= 3.3.0)

Imports exams (>= 2.3-4), glue (>= 1.4.0), stringr (>= 1.4.0), stringi (>= 1.4.6), xml2 (>= 1.2.5)

Suggests knitr

URL <https://github.com/hdarjus/WU-MyLearn-QGen>

BugReports <https://github.com/hdarjus/WU-MyLearn-QGen/issues>

Encoding UTF-8

LazyData true

VignetteBuilder knitr

RoxygenNote 7.1.0

NeedsCompilation no

Author Darjus Hosszejni [aut, cre] (<<https://orcid.org/0000-0002-3803-691X>>)

Maintainer Darjus Hosszejni <darjus.hosszejni@wu.ac.at>

Repository CRAN

Date/Publication 2020-05-25 09:50:02 UTC

R topics documented:

exams.mylearn-package	2
example_paths	2
exams2mylearn	2

Index	5
--------------	----------

exams.mylearn-package *Question Generation in the 'MyLearn' XML Format*

Description

Randomized multiple-select and single-select question generation for the 'MyLearn' platform. Question templates in the form of the R/exams package are transformed into XML format required by 'MyLearn'.

example_paths *Paths to Examples*

Description

Paths to Examples

Usage

```
example_paths()
```

Value

Named sequence of characters. Each character string points at an example file in the package.

Examples

```
ex_files <- example_paths()
ex_files
```

exams2mylearn *Exam Generation for the 'MyLearn' Platform*

Description

The 'MyLearn' distance learning and teaching platform has a special XML format. exams2mylearn transforms input files in the R/exams format to XML files and zips them. The resulting zip file can be directly uploaded to the 'MyLearn' platform.

Usage

```
exams2mylearn(
  filename,
  n,
  dir,
  tdir = NULL,
  name = NULL,
  outfile = NULL,
  dontask = !base::interactive(),
  distort.shortname = FALSE,
  ...
)
```

Arguments

filename	(character) absolute or relative path to the exercise template. Usually simply a filename pointing at a .Rmd file in the working directory
n	(integer) number of random variants to create
dir	(character) output directory, will be created if non-existent
tdir	(character, <i>optional</i>) temporary directory; will use tempdir() if unspecified
name	(character, <i>optional</i>) unique name prefix of temporary and output files, defaults to filename without the non-alphabetic characters
outfile	(character, <i>optional</i>) output filename (not a path), defaults to name.zip
dontask	(logical, <i>optional</i>) if TRUE and the output zip file exists then
distort.shortname	(logical, <i>optional</i>) should the shortname include a random ending? Defaults to FALSE
...	forwarded to exams2html

Value

If dir is invalid or unspecified, the function returns with an error. Otherwise the function produces a zip file in directory dir. The exact path to the zip file is returned invisibly.

Note

The development team has to turn on the upload functionality on a per course basis.

Examples

```
# Get the examples provided with the package
ex_files <- example_paths()
if (interactive()) {
  # Produce 40 exam questions in the current directory
  # using the example that contains a plot; the
  # output is "final_exam.zip", and we want to
```

```
# distort the shortname used by 'MyLearn'
exams2mylearn(ex_files["plot"], 40, dir = ".",
              outfile = "final_exam_question_1.zip",
              distort.shortname = TRUE)
}

# Takes some time:
## Not run:
# Produce 500 exam questions in the current
# directory using a different example with more
# verbose output from exams::exams2html
exams2mylearn(ex_files["single_choice"], 500,
              dir = ".", name = "final_exam_question_1",
              verbose = TRUE)

## End(Not run)
```

Index

`example_paths`, [2](#)
`exams.mylearn` (`exams.mylearn-package`), [2](#)
`exams.mylearn-package`, [2](#)
`exams2mylearn`, [2](#)