

# Package ‘exams2forms’

May 2, 2025

**Version** 0.2-0

**Date** 2025-05-02

**Title** Embedding 'exams' Exercises as Forms in 'rmarkdown' or 'quarto'  
Documents

**Description** Automatic generation of quizzes or individual questions as (interac-  
tive) forms within 'rmarkdown' or 'quarto' documents based on 'R/exams' exercises.

**Depends** R (>= 4.0.0), exams (>= 2.4-2)

**Imports** grDevices, tools, utils, base64enc, digest, knitr, rmarkdown

**Suggests** xfun

**License** GPL-3

**URL** <https://www.R-exams.org/tutorials/exams2forms/>

**BugReports** <https://www.R-exams.org/contact/>

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Achim Zeileis [aut, cre] (ORCID:

<<https://orcid.org/0000-0003-0918-3766>>),

Reto Stauffer [aut] (ORCID: <<https://orcid.org/0000-0002-3798-5507>>),

Dale Barr [ctb] (ORCID: <<https://orcid.org/0000-0002-1121-4608>>),

Lisa DeBruine [ctb] (ORCID: <<https://orcid.org/0000-0002-7523-5539>>),

Florian Stampfer [ctb] (ORCID: <<https://orcid.org/0000-0003-4980-4272>>),

Jonas Tscholl [ctb]

**Maintainer** Achim Zeileis <Achim.Zeileis@R-project.org>

**Repository** CRAN

**Date/Publication** 2025-05-02 08:10:02 UTC

## Contents

exams2forms . . . . .	2
exams2webquiz . . . . .	4
forms_schoice . . . . .	6

<b>Index</b>	<b>8</b>
--------------	----------

## Description

The exams2forms interface can be used within R/Markdown or Quarto documents to dynamically insert individual questions or entire quizzes based on R/exams exercise templates.

## Usage

```
exams2forms(file,
  write = TRUE, check = TRUE, box = TRUE, solution = TRUE, nchar = c(20, 40),
  schoice_display = "buttons", mchoice_display = "buttons",
  cloze_schoice_display = "dropdown", cloze_mchoice_display = mchoice_display,
  usecase = TRUE, usespace = TRUE, auto = FALSE,
  show_filename = !isFALSE(auto), show_tolerance = !isFALSE(auto),
  n = 1L, nsamp = NULL, dir = ".", edir = NULL, tdir = NULL, sdir = NULL, verbose = FALSE,
  quiet = TRUE, resolution = 100, width = 4, height = 4, svg = FALSE,
  converter = "pandoc-mathjax", base64 = NULL, obfuscate = TRUE, ...)
```

## Arguments

file	character. A specification of (a list of) exercise files.
write	logical. Should <code>writeln</code> be called on the resulting Markdown/HTML text?
check	logical. Should a check for showing the answer be added to the exercise?
box	logical. Should a box be added around the check environment?
solution	logical or character. Should the solution environment of the R/exams exercises (if any) be displayed or not? Optionally, <code>solution</code> can also be a character with the title for the corresponding solution box.
nchar	numeric. Number of characters for numeric or string fields ( <code>forms_num</code> , <code>forms_string</code> ). If a vector of length 2 is provided it is used as the minimum and maximum number of characters, respectively.
schoice_display, cloze_mchoice_display	character. Should the interaction for single-choice or multiple-choice forms be displayed with buttons (radio buttons or checkboxes, respectively) or with drop-down menus? The default is to use buttons for single-choice and multiple-choice items and for multiple-choice interactions within cloze items. Only for single-choice interactions within cloze items, the default is to use drop-down menus.
usespace	logical. Defaults to TRUE, only used for string questions. If FALSE spaces are ignored when evaluating correct answers.
usecase	logical. Defaults to TRUE, only used for string questions. If FALSE answers are evaluated ignoring uppercase/lowercase.

auto	logical or named list/vector. If set to TRUE, the form fields are prefilled, the check button is clicked, the solution button (if any) is clicked, and the order of the exercises (if $n > 1$ ) is not shuffled. The options can also be controlled separately if auto is a named list/vector with (a subset of the) names prefill, check, solution, noshuffle.
show_filename	logical. Should the tolerance for numeric fields be shown before the field? Enabled by default only if the form is auto-filled/clicked.
show_tolerance	logical. Should the filename be displayed in the form? Enabled by default only if the form is auto-filled/clicked.
n	integer. The number of copies to be taken from file.
nsamp	integer. The number(s) of exercise files sampled from each list element of file. Sampling without replacement is used if possible. (Only if some element of nsamp is larger than the length of the corresponding element in file, sampling with replacement is used.)
dir	character. Standard argument for exams interfaces and hence included in the argument - but actually not used because exams2forms does not write output files.
edir	character specifying the path of the directory (along with its sub-directories) in which the files in file are stored.
tdir	character specifying a temporary directory, by default this is chosen via <a href="#">tempfile</a> . Note that this is cleaned up (i.e., existing files are deleted) and only certain temporary files are preserved.
sdir	character specifying a directory for storing supplements, by default this is chosen via <a href="#">tempfile</a> .
verbose	logical. Should information on progress of exam generation be reported?
quiet	logical. Should output be suppressed when calling <a href="#">xweave</a> ?
resolution, width, height	numeric. Options for rendering PNG (or SVG) graphics passed to <a href="#">xweave</a> .
svg	logical. Should graphics be rendered in SVG or PNG (default)?
converter	passed on to <a href="#">make_exercise_transform_html</a> (if markup = "html").
base64	logical. Should supplementary files be embedded using Base 64 coding? Passed on to <a href="#">make_exercise_transform_pandoc</a> . The default is to enable Base 64 when an HTML-type document is produced.
...	currently not used.
obfuscate	logical. Should the correct answer be obfuscated when embedded in the HTML source code? If FALSE, it is included as plain text.

## Details

exams2forms provides an interface for including exercises from R/exams (<https://www.R-exams.org/>) in R/Markdown or Quarto documents. This is done by rendering the exercises (either in .Rnw or .Rmd format) into Markdown which can then be included in .Rmd or .qmd documents. For a short introduction see <https://www.R-exams.org/tutorials/exams2forms/>.

The function `exams2webquiz` can set up a suitable .Rmd file and directly render it. This is intended mostly for quick interactive testing when authoring R/exams exercises.

The idea and original code for inserting interactions into R/Markdown documents along with CSS and Javascript is adapted from the **webexercises** package, authored by Dale Barr and Lisa DeBruine.

## Value

A list of exercises, each of which is a character vector with the Markdown code of the individual exercises.

## See Also

`exams2webquiz`, `xexams`, `xweave`

## Examples

```
## example R/exams exercises: string, num, schoice, mchoice
exams2forms("function.Rmd")
exams2forms("deriv.Rmd")
exams2forms("swisscapital.Rmd")
exams2forms("switzerland.Rmd")

## usually exams2forms() is used within R/Markdown or Quarto tutorials,
## two example files are shipped within the package.
## - questions.Rmd: all questions set up individually
## - quiz.Rmd: an entire quiz set up in one go
lr <- system.file("forms", package = "exams2forms")
dir(lr)

## quiz.Rmd is rather short and straightforward
rmd <- file.path(lr, "quiz.Rmd")
writeLines(readLines(rmd))

## Not run:
## either .Rmd file can be loaded in RStudio and rendered from there or
## using the command line, e.g.,
rmarkdown::render(rmd)

## End(Not run)
```

## Description

The `exams2webquiz` function is a convenience tool that sets up an R/Markdown document embedding R/exams exercises in a (temporary) directory and directly rendering it to an HTML page.

**Usage**

```
exams2webquiz(file, n = 1L, nsamp = NULL, dir = NULL,
  name = "webquiz", title = "R/exams quiz", browse = TRUE, edir = NULL,
  ..., clean = TRUE, quiet = TRUE, envir = parent.frame())

webquiz(...)
```

**Arguments**

<code>file</code>	character. A specification of (a list of) exercise files.
<code>n</code>	integer. The number of copies to be taken from <code>file</code> .
<code>nsamp</code>	integer. The number(s) of exercise files sampled from each list element of <code>file</code> . Sampling without replacement is used if possible. (Only if some element of <code>nsamp</code> is larger than the length of the corresponding element in <code>file</code> , sampling with replacement is used.)
<code>dir</code>	character. A path in which the tutorial file is created, by default chosen as a <a href="#">tempfile</a> .
<code>name</code>	character. A name prefix for the resulting <code>.Rmd</code> document.
<code>title</code>	character. Title of the quiz.
<code>browse</code>	logical. Should the resulting tutorial be displayed using <a href="#">browseURL</a> ?
<code>edir</code>	character. The path of the directory (along with its sub-directories) in which the files in <code>file</code> are stored. If both <code>dir</code> and <code>edir</code> are <code>NULL</code> (the default), the <code>edir</code> is set to the current working directory.
<code>...</code>	arguments passed to <a href="#">exams2forms</a> (in <code>exams2webquiz</code> ) or <a href="#">html_document</a> (in <code>webquiz</code> ), respectively.
<code>clean, quiet, envir</code>	arguments passed to <a href="#">render</a> .

**Details**

`exams2webquiz` is a convenience function that sets up a `.Rmd` document (in a temporary directory by default) and then calls [render](#) from **rmarkdown** to quickly try out the quiz interactively.

`webquiz` is a small wrapper function that creates an [html\\_document](#) but includes some custom CSS and Javascript for the quiz display and user interactions. The idea and original code is adapted from the **webexercises** package, authored by Dale Barr and Lisa DeBruine.

For full customization it is recommended to set up a dedicated `.Rmd` file within which [exams2forms](#) can be used to include R/exams exercises.

**Value**

Invisible character vector containing the path of the `Rmd` document and rendered `HTML` file, respectively.

**See Also**

[exams2forms](#), [render](#)

## Examples

```
## Not run:
## quickly render a quiz based on a set of R/exams exercises
ex <- c("swisscapital.Rmd", "capitals.Rmd", "fruit.Rmd", "function.Rmd", "lm2.Rmd")
exams2webquiz(ex)

## End(Not run)
```

---

forms\_schoice

---

*Create Interaction Forms for Different Exercise Types*


---

## Description

Helper functions for `exams2forms` to create the interaction forms for different R/exams exercise types: single-choice (schoice), multiple-choice (mchoice), text (string), or numeric (num).

## Usage

```
forms_schoice(answerlist, solution,
  display = c("buttons", "dropdown"), obfuscate = TRUE)

forms_mchoice(answerlist, solution,
  display = c("buttons", "dropdown"), obfuscate = TRUE)

forms_string(answer, width = NULL,
  usecase = TRUE, usespace = FALSE, regex = FALSE, obfuscate = TRUE)

forms_num(answer, tol = 0, width = NULL,
  usespace = FALSE, regex = FALSE, obfuscate = TRUE)
```

## Arguments

answerlist	character vector of answer alternatives for a single-choice or multiple-choice answer.
solution	logical vector indicating which alternatives in answerlist are correct. Must have the same length as answerlist. For forms_schoice exactly one element of solution must be TRUE.
display	character. Should the interaction for single-choice or multiple-choice forms be displayed with buttons (radio buttons or checkboxes, respectively) or with drop-down menus?
answer	character (for forms_string) or numeric (for forms_num). Correct answer(s).
width	numeric. Width of the input box in characters. Defaults to the length of answer.
usecase	logical. Should the evaluation of the input be case-sensitive?
usespace	logical. Should the evaluation of the input assess white spaces?
regex	logical. Should the evaluation of the input employ regular expressions?

tol	numeric. Tolerance for the evaluation of numeric input.
obfuscate	logical. Should the correct answer be obfuscated when embedded in the HTML source code? If FALSE, it is included as plain text.

**Value**

A character string with the HTML or Markdown code for the interaction forms to be embedded in R/Markdown or Quarto documents. For a short introduction see <https://www.R-exams.org/tutorials/exams2forms/>.

**Examples**

```
## single-choice: What is the third letter of the latin alphabet?
forms_schoice(c("A", "B", "C", "D"), c(FALSE, FALSE, TRUE, FALSE),
  obfuscate = FALSE)

## multiple-choice: Which of these numbers are prime?
forms_mchoice(c("1", "2", "3", "4"), c(FALSE, TRUE, TRUE, FALSE),
  obfuscate = FALSE)

## string: Which base R function estimates linear regression models?
forms_string("lm", width = 10, obfuscate = FALSE)

## numeric: A population of 1000 grows with a nominal rate of 10% per period.
## How big is the population after three periods?
forms_num(1000 * exp(0.1 * 3), tol = 0.1, width = 10, obfuscate = FALSE)
```

# Index

## \* **utilities**

forms\_schoice, [6](#)

browseURL, [5](#)

exams2forms, [2](#), [5](#), [6](#)

exams2webquiz, [4](#), [4](#)

forms\_mchoice(forms\_schoice), [6](#)

forms\_num, [2](#)

forms\_num(forms\_schoice), [6](#)

forms\_schoice, [6](#)

forms\_string, [2](#)

forms\_string(forms\_schoice), [6](#)

html\_document, [5](#)

make\_exercise\_transform\_html, [3](#)

make\_exercise\_transform\_pandoc, [3](#)

render, [5](#)

tempfile, [3](#), [5](#)

webquiz(exams2webquiz), [4](#)

writelnLines, [2](#)

xexams, [4](#)

xweave, [3](#), [4](#)