

Package: moodlequiz (via r-universe)

May 10, 2026

Title R Markdown format for 'Moodle' XML cloze quizzes

Version 0.2.1.9000

Description Enables the creation of 'Moodle' quiz questions using literate programming with R Markdown. This makes it easy to quickly create a quiz that can be randomly replicated with new datasets, questions, and options for answers.

License MIT + file LICENSE

Encoding UTF-8

Language en

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports rlang, rmarkdown, bookdown, xfun, yaml

URL <https://github.com/numbats/moodlequiz>,
<https://numbats.github.io/moodlequiz/>

BugReports <https://github.com/numbats/moodlequiz/issues>

Suggests knitr, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/pak/sysreqs cmake make libuv1-dev

Repository <https://numbats.r-universe.dev>

Date/Publication 2026-04-10 09:09:42 UTC

RemoteUrl <https://github.com/numbats/moodlequiz>

RemoteRef HEAD

RemoteSha 99ef3b7da8adb9c144b57bc024921557ad2d92c2

Contents

choices	2
cloze_questions	2
moodlequiz	5

Index	7
--------------	----------

choices	<i>Create a set of choices for single or multiple choice questions</i>
---------	--

Description

Create a set of choices for single or multiple choice questions

Usage

```
choices(options, answer)
```

Arguments

options	A character vector of selectable choices
answer	A character vector of the correct answers

Value

A named vector of choices suitable for use with `cloze_singlechoice()` and `cloze_multichoice()`

See Also

[cloze_singlechoice\(\)](#), [cloze_multichoice\(\)](#)

cloze_questions	<i>Generate Cloze-Type Questions for Moodle</i>
-----------------	---

Description

These functions create cloze-type questions for Moodle quizzes, designed for use with inline R code chunks in an R Markdown document formatted with the `moodlequiz::moodlequiz` output format.

Usage

```
cloze_shortanswer(  
  options,  
  weight = max(options),  
  feedback = "",  
  case_sensitive = FALSE  
)
```

```
cloze_multichoice(  
  options,  
  weight = max(options),  
  feedback = "",
```

```

    type = c("vertical", "horizontal"),
    shuffle = FALSE
)

cloze_singlechoice(
  options,
  weight = max(options),
  feedback = "",
  type = c("dropdown", "vertical", "horizontal"),
  shuffle = FALSE
)

cloze_numerical(answer, weight = 1, tolerance = 0, feedback = "")

cloze(x, ...)

```

Arguments

options	A named vector of answer options. For single/multiple choice questions the <code>choices()</code> helper function can help create this vector. Names correspond to answers, and values specify their weights (e.g., 100 for a correct answer or partial weights for partially correct answers). For multiple-choice and single-choice questions, this includes both correct and distractor options.
weight	A numeric value specifying the weight for the question. Defaults to the highest weight in options.
feedback	A character vector providing feedback for answers.
case_sensitive	Logical. For <code>cloze_shortanswer</code> , whether the answer should be case-sensitive. Defaults to FALSE.
type	A character string specifying the presentation style of the options. For <code>cloze_multichoice</code> , valid values are "vertical" or "horizontal". For <code>cloze_singlechoice</code> , valid values are "dropdown", "vertical", or "horizontal".
shuffle	Logical. For <code>cloze_multichoice</code> and <code>cloze_singlechoice</code> , whether the answer options should be shuffled. Defaults to FALSE.
answer	A numeric value specifying the correct numerical answer(s).
tolerance	A numeric value specifying the acceptable range of deviation for <code>cloze_numerical</code> answers. Defaults to 0.
x	For <code>cloze()</code> , the correct answer which also determines the question type (e.g. numeric will use <code>cloze_numerical()</code> and character will use <code>cloze_shortanswer()</code> or <code>cloze_singlechoice()/cloze_multichoice()</code> if selectable options are given as the second argument).
...	Additional arguments passed to other <code>cloze()</code> methods (such as the available options and other <code>cloze_*</code> (<code>)</code> arguments).

Value

A character string containing the Moodle-compatible XML or inline text for the specified cloze question(s).

Functions

- `cloze_shortanswer()`: Creates a short-answer question where the student provides a text response.
- `cloze_singlechoice()`: Generates a single-choice question where students select one correct answer from a list.
- `cloze_multichoice()`: Creates a multiple-choice question where students can select one or more correct answers.
- `cloze_numerical()`: Generates a numerical question where students input a numeric response with optional tolerance.
- `cloze()`: Automatic question types based on the class of the answers.

Examples

```
# Short-answer question: Where is the best coffee?
cloze_shortanswer(
  options = c("Melbourne" = 1),
  case_sensitive = FALSE
)

# Multiple-choice question: Select all lower-case answers
cloze_multichoice(
  options = c("a" = 1, "F" = 0, "g" = 1, "V" = 0, "K" = 0),
  type = "vertical"
)

# Where is Melbourne?
cloze_singlechoice(
  choices(
    c("New South Wales", "Victoria", "Queensland", "Western Australia",
      "South Australia", "Tasmania", "Australian Capital Territory",
      "Northern Territory"),
    "Victoria"
  ),
  type = "dropdown"
)

# Numerical question: Pick a number between 1 and 10
cloze_numerical(
  answer = 5.5,
  tolerance = 4.5
)

# Automatic cloze questions
cloze(42) # Numerical
cloze("Australia") # Short answer
cloze("rep_len", c("rep", "rep.int", "rep_len", "replicate")) # Single choice
cloze(c("A", "B", "C"), LETTERS) # Multiple choice
```

moodlequiz

R Markdown format for Moodle XML quizzes

Description

Provides an alternative interface to working with the exams package for producing Moodle questions any type.

Usage

```
moodlequiz(
  replicates = 1L,
  self_contained = TRUE,
  extra_dependencies = NULL,
  theme = NULL,
  includes = NULL,
  lib_dir = NULL,
  md_extensions = NULL,
  pandoc_args = NULL,
  ...
)
```

Arguments

- | | |
|--------------------|---|
| replicates | The number of times the questions are rendered, useful for producing multiple versions of the same quiz with different random samples. To keep identify replicates of questions for random importation into Moodle we recommend organising the materials into categories using top level headers. |
| self_contained | Produce a standalone HTML file with no external dependencies, using data: URIs to incorporate the contents of linked scripts, stylesheets, images, and videos. Note that even for self contained documents MathJax is still loaded externally (this is necessary because of its size). |
| extra_dependencies | Extra dependencies as a list of the <code>html_dependency</code> class objects typically generated by <code>htmltools::htmlDependency()</code> . |
| theme | One of the following: <ul style="list-style-type: none"> • A <code>bslib::bs_theme()</code> object (or a list of <code>bslib::bs_theme()</code> argument values) <ul style="list-style-type: none"> – Use this option for custom themes using Bootstrap 4 or 3. – In this case, any <code>.scss/.sass</code> files provided to the <code>css</code> parameter may utilize the theme's underlying Sass utilities (e.g., variables, mixins, etc). • NULL for no theme (i.e., no <code>html_dependency_bootstrap()</code>). • A character string specifying a Bootstrap 3 theme name (for backwards-compatibility). |

<code>includes</code>	Named list of additional content to include within the document (typically created using the includes function).
<code>lib_dir</code>	Directory to copy dependent HTML libraries (e.g. jquery, bootstrap, etc.) into. By default this will be the name of the document with <code>_files</code> appended to it.
<code>md_extensions</code>	Markdown extensions to be added or removed from the default definition of R Markdown. See the rmarkdown_format for additional details.
<code>pandoc_args</code>	Additional command line options to pass to pandoc
<code>...</code>	Additional function arguments to pass to the base R Markdown HTML output formatter html_document_base

Value

R Markdown output format to pass to [rmarkdown::render\(\)](#)

Index

`bslib::bs_theme()`, 5

choices, 2

`choices()`, 3

`cloze (cloze_questions)`, 2

`cloze_multichoice (cloze_questions)`, 2

`cloze_multichoice()`, 2

`cloze_numerical (cloze_questions)`, 2

`cloze_questions`, 2

`cloze_shortanswer (cloze_questions)`, 2

`cloze_singlechoice (cloze_questions)`, 2

`cloze_singlechoice()`, 2

`html_dependency_bootstrap()`, 5

`html_document_base`, 6

`htmltools::htmlDependency()`, 5

includes, 6

moodlequiz, 5

`rmarkdown::render()`, 6

`rmarkdown_format`, 6