

# Package ‘moodef’

March 9, 2024

**Type** Package

**Title** Defining 'Moodle' Elements from R

**Version** 1.1.0

**Description** The main objective of this package is to support the definition of 'Moodle' elements taking advantage of the power that R offers. In this first version, it allows the definition of questions to be included in the question bank.

**License** MIT + file LICENSE

**URL** <https://josesamos.github.io/moodef/>,  
<https://github.com/josesamos/moodef>

**BugReports** <https://github.com/josesamos/moodef/issues>

**Imports** blastula, glue, magick, purrr, readr, readxl, snakecase,  
tibble, tidyr, tools, xlsx, xml2

**Suggests** knitr, pander, rmarkdown, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Encoding** UTF-8

**Language** en-GB

**RoxygenNote** 7.3.1

**NeedsCompilation** no

**Author** Jose Samos [aut, cre] (<<https://orcid.org/0000-0002-4457-3439>>),  
Universidad de Granada [cph]

**Maintainer** Jose Samos <[jsamos@ugr.es](mailto:jsamos@ugr.es)>

**Repository** CRAN

**Date/Publication** 2024-03-09 13:30:05 UTC

## R topics documented:

create_question_csv . . . . .	2
create_question_data_frame . . . . .	3
create_question_excel . . . . .	3
define_question . . . . .	4
define_questions_from_csv . . . . .	5
define_questions_from_data_frame . . . . .	6
define_questions_from_excel . . . . .	7
generate_xml . . . . .	8
generate_xml_file . . . . .	9
question_category . . . . .	10
read_question_csv . . . . .	11
read_question_excel . . . . .	12
vector_to_string . . . . .	13

<b>Index</b>	<b>14</b>
--------------	-----------

---

create_question_csv	<i>Create a question csv file</i>
---------------------	-----------------------------------

---

### Description

Creates an empty question csv file.

### Usage

```
create_question_csv(file, sep = ",")
```

### Arguments

file	A string, name of a text file.
sep	Column separator character.

### Value

A string.

### See Also

Other support functions: [create\\_question\\_data\\_frame\(\)](#), [create\\_question\\_excel\(\)](#), [read\\_question\\_csv\(\)](#), [read\\_question\\_excel\(\)](#), [vector\\_to\\_string\(\)](#)

### Examples

```
file <- create_question_csv(file = tempfile(fileext = '.csv'))
```

---

create\_question\_data\_frame  
*Create a question data frame*

---

**Description**

Creates an empty question data frame.

**Usage**

```
create_question_data_frame()
```

**Value**

A data frame.

**See Also**

Other support functions: [create\\_question\\_csv\(\)](#), [create\\_question\\_excel\(\)](#), [read\\_question\\_csv\(\)](#), [read\\_question\\_excel\(\)](#), [vector\\_to\\_string\(\)](#)

**Examples**

```
df <- create_question_data_frame()
```

---

create\_question\_excel *Create a question Excel file*

---

**Description**

Creates an empty question Excel file.

**Usage**

```
create_question_excel(file)
```

**Arguments**

file            A string, name of a text file.

**Value**

A string.

**See Also**

Other support functions: [create\\_question\\_csv\(\)](#), [create\\_question\\_data\\_frame\(\)](#), [read\\_question\\_csv\(\)](#), [read\\_question\\_excel\(\)](#), [vector\\_to\\_string\(\)](#)

**Examples**

```
file <- create_question_excel(file = tempfile(fileext = '.xlsx'))
```

---

define_question	<i>Define question</i>
-----------------	------------------------

---

**Description**

Define a question and the possible answers. The type of question is deduced.

**Usage**

```
define_question(qc, type, question, image, image_alt, answer, ...)
```

```
## S3 method for class 'question_category'
define_question(
  qc,
  type = "",
  question = "",
  image = "",
  image_alt = "",
  answer = "",
  ...
)
```

**Arguments**

qc	A question_category object.
type	A string, question type (if needed).
question	A string, statement of the question.
image	A string, optional, image file to include in the question.
image_alt	A string, description of the image to include in the question.
answer	A string, correct answer to the question.
...	A string, rest of the answers to the question.

## Details

If we include an image in the question, we must also include text in the alt field associated with it. After the correct answer, we can indicate as many answers as we want, if we do not indicate all the parameters, we have to give each answer a parameter name different from the rest of the parameter names.

## Value

A question\_category.

## See Also

Other question definition: [define\\_questions\\_from\\_csv\(\)](#), [define\\_questions\\_from\\_data\\_frame\(\)](#), [define\\_questions\\_from\\_excel\(\)](#), [generate\\_xml\(\)](#), [generate\\_xml\\_file\(\)](#), [question\\_category\(\)](#)

## Examples

```
qc <- question_category(category = 'Initial test') |>
  define_question(
    question = 'What are the basic arithmetic operations?',
    answer = 'Addition, subtraction, multiplication and division.',
    a_1 = 'Addition and subtraction.',
    a_2 = 'Addition, subtraction, multiplication, division and square root.'
  )
```

---

define\_questions\_from\_csv

*Define questions from a csv file*

---

## Description

Each row in the text file is interpreted as a question. We only have to define the columns that we are going to use, the rest of the columns are taken by default.

## Usage

```
define_questions_from_csv(qc, file, sep)

## S3 method for class 'question_category'
define_questions_from_csv(qc, file, sep = ",")
```

## Arguments

qc	A question_category object.
file	A string, name of a text file.
sep	Column separator character.

**Details**

For answers where a vector is required, "<|>" is used as a separator of the vector elements.

**Value**

A question\_category.

**See Also**

Other question definition: [define\\_question\(\)](#), [define\\_questions\\_from\\_data\\_frame\(\)](#), [define\\_questions\\_from\\_excel\\_file\(\)](#), [generate\\_xml\(\)](#), [generate\\_xml\\_file\(\)](#), [question\\_category\(\)](#)

**Examples**

```
file <- system.file("extdata", "questions.csv", package = "moodef")
qc <-
  question_category(category = 'Initial test', adapt_images = TRUE) |>
  define_questions_from_csv(file = file)
```

---

define\_questions\_from\_data\_frame

*Define questions from a data frame*

---

**Description**

Each row in the text data frame is interpreted as a question. We only have to define the columns that we are going to use, the rest of the columns are taken by default.

**Usage**

```
define_questions_from_data_frame(qc, df)

## S3 method for class 'question_category'
define_questions_from_data_frame(qc, df)
```

**Arguments**

qc	A question_category object.
df	A data frame.

**Details**

For answers where a vector is required, "<|>" is used as a separator of the vector elements.

**Value**

A question\_category.

**See Also**

Other question definition: [define\\_question\(\)](#), [define\\_questions\\_from\\_csv\(\)](#), [define\\_questions\\_from\\_excel\(\)](#), [generate\\_xml\(\)](#), [generate\\_xml\\_file\(\)](#), [question\\_category\(\)](#)

**Examples**

```
file <- system.file("extdata", "questions.csv", package = "moodef")
df <- read_question_csv(file = file)

qc <-
  question_category(category = 'Initial test', adapt_images = TRUE) |>
  define_questions_from_data_frame(df)
```

---

define\_questions\_from\_excel

*Define questions from a Excel file*

---

**Description**

Each row in the Excel file is interpreted as a question. We only have to define the columns that we are going to use, the rest of the columns are taken by default.

**Usage**

```
define_questions_from_excel(qc, file, sheet_index, sheet_name)

## S3 method for class 'question_category'
define_questions_from_excel(qc, file, sheet_index = NULL, sheet_name = NULL)
```

**Arguments**

qc	A question_category object.
file	A string, name of an Excel file.
sheet_index	A number, sheet index in the workbook.
sheet_name	A string, sheet name.

**Details**

In addition to the file, we can indicate the sheet by its name or index. If we do not indicate anything, it considers the first sheet.

For answers where a vector is required, "<>" is used as a separator of the vector elements.

**Value**

A question\_category.

**See Also**

Other question definition: [define\\_question\(\)](#), [define\\_questions\\_from\\_csv\(\)](#), [define\\_questions\\_from\\_data\\_frame\(\)](#), [generate\\_xml\(\)](#), [generate\\_xml\\_file\(\)](#), [question\\_category\(\)](#)

**Examples**

```
file <- system.file("extdata", "questions.xlsx", package = "moodef")
qc <-
  question_category(category = 'Initial test', adapt_images = TRUE) |>
  define_questions_from_excel(file = file)
```

---

generate\_xml

*Generate questions xml string*

---

**Description**

Generate questions xml string

**Usage**

```
generate_xml(qc)

## S3 method for class 'question_category'
generate_xml(qc)
```

**Arguments**

qc                    A question\_category object.

**Value**

A string.

**See Also**

Other question definition: [define\\_question\(\)](#), [define\\_questions\\_from\\_csv\(\)](#), [define\\_questions\\_from\\_data\\_frame\(\)](#), [define\\_questions\\_from\\_excel\(\)](#), [generate\\_xml\\_file\(\)](#), [question\\_category\(\)](#)

## Examples

```
qc <- question_category(category = 'Initial test') |>
  define_question(
    question = 'What are the basic arithmetic operations?',
    answer = 'Addition, subtraction, multiplication and division.',
    a_1 = 'Addition and subtraction.',
    a_2 = 'Addition, subtraction, multiplication, division and square root.'
  )

xml <- qc |>
  generate_xml()
```

---

generate_xml_file	<i>Generate questions xml file</i>
-------------------	------------------------------------

---

## Description

Generate questions xml file

## Usage

```
generate_xml_file(qc, file)

## S3 method for class 'question_category'
generate_xml_file(qc, file = NULL)
```

## Arguments

qc	A question_category object.
file	A string, file name.

## Value

A question\_category.

## See Also

Other question definition: [define\\_question\(\)](#), [define\\_questions\\_from\\_csv\(\)](#), [define\\_questions\\_from\\_data\\_frame\(\)](#), [define\\_questions\\_from\\_excel\(\)](#), [generate\\_xml\(\)](#), [question\\_category\(\)](#)

**Examples**

```
qc <- question_category(category = 'Initial test') |>
  define_question(
    question = 'What are the basic arithmetic operations?',
    answer = 'Addition, subtraction, multiplication and division.',
    a_1 = 'Addition and subtraction.',
    a_2 = 'Addition, subtraction, multiplication, division and square root.'
  ) |>
  generate_xml_file(file = tempfile(fileext = '.xml'))
```

---

question_category	question_category <i>S3 class</i>
-------------------	-----------------------------------

---

**Description**

Creates a question\_category object.

**Usage**

```
question_category(
  category = "Default category",
  first_question_number = 1,
  copyright = "",
  license = "",
  correct_feedback = "Correct.",
  partially_correct_feedback = "Partially correct.",
  incorrect_feedback = "Incorrect.",
  adapt_images = FALSE,
  width = 800,
  height = 600
)
```

**Arguments**

category	A string, category name.
first_question_number	An integer, first number to compose the question names.
copyright	A string, copyright text to be included in each question that is defined.
license	A string, license text to be included in each question that is defined.
correct_feedback	A string, feedback on correct answers to each question.
partially_correct_feedback	A string, feedback on partially correct answers to each question.

incorrect_feedback	A string, feedback on incorrect answers to each question.
adapt_images	A boolean, adapt the images so that they are a similar size.
width	A integer, width of each image.
height	A integer, height of each image.

### Details

Defines a category of questions to be included in the *Moodle* question bank.

It allows us to define the name of the category, the copyright and license literals that will be added to each question, and the feedback literals for correct, partially correct and incorrect questions.

Each question can include an image after the text. We can also configure if we want to automatically transform the images so that they have a standard size that we can also indicate.

### Value

A question\_category object.

### See Also

Other question definition: [define\\_question\(\)](#), [define\\_questions\\_from\\_csv\(\)](#), [define\\_questions\\_from\\_data\\_frame\(\)](#), [define\\_questions\\_from\\_excel\(\)](#), [generate\\_xml\(\)](#), [generate\\_xml\\_file\(\)](#)

### Examples

```
qc <- question_category(category = 'Initial test')
```

---

read_question_csv	<i>Read a question csv file</i>
-------------------	---------------------------------

---

### Description

Reads a csv file of questions and returns a data frame.

### Usage

```
read_question_csv(file, sep = ",")
```

### Arguments

file	A string, name of a text file.
sep	Column separator character.

**Value**

A data frame.

**See Also**

Other support functions: [create\\_question\\_csv\(\)](#), [create\\_question\\_data\\_frame\(\)](#), [create\\_question\\_excel\(\)](#), [read\\_question\\_excel\(\)](#), [vector\\_to\\_string\(\)](#)

**Examples**

```
file <- system.file("extdata", "questions.csv", package = "moodef")
df <- read_question_csv(file = file)
```

---

read\_question\_excel    *Read a question Excel file*

---

**Description**

Reads an Excel file of questions and returns a data frame.

**Usage**

```
read_question_excel(file, sheet_index = NULL, sheet_name = NULL)
```

**Arguments**

file	A string, name of a text file.
sheet_index	A number, sheet index in the workbook.
sheet_name	A string, sheet name.

**Details**

In addition to the file, we can indicate the sheet by its name or index. If we do not indicate anything, it considers the first sheet.

**Value**

A data frame.

**See Also**

Other support functions: [create\\_question\\_csv\(\)](#), [create\\_question\\_data\\_frame\(\)](#), [create\\_question\\_excel\(\)](#), [read\\_question\\_csv\(\)](#), [vector\\_to\\_string\(\)](#)

**Examples**

```
file <- system.file("extdata", "questions.xlsx", package = "moodef")
df <- read_question_excel(file = file)
```

---

vector_to_string	<i>Transforms a vector of strings into a string</i>
------------------	---

---

**Description**

Insert the separator that we consider to later perform the reverse operation.

**Usage**

```
vector_to_string(vector)
```

**Arguments**

vector            A vector of strings.

**Value**

A string.

**See Also**

Other support functions: [create\\_question\\_csv\(\)](#), [create\\_question\\_data\\_frame\(\)](#), [create\\_question\\_excel\(\)](#), [read\\_question\\_csv\(\)](#), [read\\_question\\_excel\(\)](#)

**Examples**

```
s <- vector_to_string(c('Addition', '+'))
```

# Index

## \* question definition

- define\_question, [4](#)
- define\_questions\_from\_csv, [5](#)
- define\_questions\_from\_data\_frame,  
[6](#)
- define\_questions\_from\_excel, [7](#)
- generate\_xml, [8](#)
- generate\_xml\_file, [9](#)
- question\_category, [10](#)

## \* support functions

- create\_question\_csv, [2](#)
- create\_question\_data\_frame, [3](#)
- create\_question\_excel, [3](#)
- read\_question\_csv, [11](#)
- read\_question\_excel, [12](#)
- vector\_to\_string, [13](#)

[create\\_question\\_csv](#), [2](#), [3](#), [4](#), [12](#), [13](#)  
[create\\_question\\_data\\_frame](#), [2](#), [3](#), [4](#), [12](#),  
[13](#)  
[create\\_question\\_excel](#), [2](#), [3](#), [3](#), [12](#), [13](#)

[define\\_question](#), [4](#), [6–9](#), [11](#)  
[define\\_questions\\_from\\_csv](#), [5](#), [5](#), [7–9](#), [11](#)  
[define\\_questions\\_from\\_data\\_frame](#), [5](#), [6](#),  
[6](#), [8](#), [9](#), [11](#)  
[define\\_questions\\_from\\_excel](#), [5–7](#), [7](#), [8](#), [9](#),  
[11](#)

[generate\\_xml](#), [5–8](#), [8](#), [9](#), [11](#)  
[generate\\_xml\\_file](#), [5–8](#), [9](#), [11](#)

[question\\_category](#), [5–9](#), [10](#)

[read\\_question\\_csv](#), [2–4](#), [11](#), [12](#), [13](#)  
[read\\_question\\_excel](#), [2–4](#), [12](#), [12](#), [13](#)

[vector\\_to\\_string](#), [2–4](#), [12](#), [13](#)