

Package ‘SARP.moodle’

April 12, 2019

Type Package

Title XML Output Functions for Easy Creation of Moodle Questions

Version 0.3.10

Date 2019-04-11

Maintainer Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

License Artistic-2.0

Description Provides a set of basic functions for creating Moodle XML output files suited for importing questions in Moodle (a learning management system, see <<https://moodle.org/>> for more information).

Suggests

Depends

Imports stats,utils

Encoding UTF-8

Language fr

NeedsCompilation no

Author Emmanuel Curis [aut, cre]

Repository CRAN

Date/Publication 2019-04-12 12:22:40 UTC

R topics documented:

affichage.moodle	2
categorie.moodle	4
csv.moodle	5
csv_glossaire.moodle	8
debuter_xml.moodle	10
definir_dossier.image.moodle	12
glossaire.moodle	14
libre.moodle	15
numerique.moodle	17

ouverte.moodle	19
qcm.moodle	20
qroc.moodle	22
question.moodle	24
SARP.moodle	26
vf.moodle	27

Index	29
--------------	-----------

affichage.moodle	<i>Aide aux affichages dans les textes des questions</i>
------------------	--

Description

Ces fonctions permettent de créer une chaîne de caractère au format HTML, qui peut ensuite être intégrée au texte d'une question (énoncé, réponse, commentaire...).

Usage

```

afficher_echantillon.moodle( x, tableau = TRUE,
                             n.lignes = 1, trier = FALSE )
afficher_nombre.moodle( x,
                        dec = get( "decimal", enviro = SARP.Moodle.env ),
                        n.chiffres = get( "nombre.chiffres", enviro = SARP.Moodle.env ) )

```

Arguments

x	L'élément à convertir au format HTML. Voyez les détails pour plus de précisions. Options pour <code>afficher_echantillon.moodle</code>
tableau	Si TRUE, un tableau HTML est construit. Sinon, une simple énumération des valeurs est préparée.
n.lignes	Si l'on construit un tableau, combien de lignes doit-il avoir ? <i>Actuellement ignoré</i>
trier	Si TRUE, les valeurs de l'échantillon sont triées par ordre croissant avant d'être converties en code HTML. Options pour <code>afficher_nombre.moodle</code>
dec	Le symbole à utiliser comme séparateur décimal. Par défaut, celui indiqué à la création du questionnaire.
n.chiffres	Le nombre de chiffres à conserver pour l'affichage. Si le nombre est plus petit, en valeur absolue, que $10^{-n.chiffres}$, il ne peut pas être affiché avec ce nombre de décimales : c'est alors le nombre de chiffres significatifs. Sinon, c'est le nombre de décimales.

Details

Pour `afficher_echantillon.moodle`, `x` peut être un vecteur ou une liste; chaque élément est converti en texte avec `afficher_nombre.moodle`. La liste des valeurs est alors créée, soit sous la forme (`x1 ; x2 ; ...`), soit sous la forme d'un tableau d'une ligne, avec un filet au-dessus et en-dessous.

Pour `afficher_nombre.moodle`, `x` doit être un vecteur contenant au plus une seule valeur (dans le cas contraire, seule la première est utilisée, les autres sont ignorées avec un avertissement). S'il est de longueur nulle, le symbole de l'ensemble vide (entité `∅`) est renvoyé, mis en forme avec le style utilisé pour les messages d'erreur. Les valeurs manquantes (NA) ou non-numériques (NaN) sont affichées avec ces caractères, et le style des messages d'erreur. Les valeurs infinies (Inf et -Inf) sont affichées avec le symbole ∞ (entité `∞`). Les chaînes de caractère sont affichées inchangées. Enfin, les nombres sont mis en forme avec le symbole dec comme séparateur décimal, un signe moins correct (entité `−`) et, si nécessaire, une puissance de 10 correcte (soit ` × 10ⁿ`, où n est la puissance, pour avoir $\times 10^n$ avec des espacements corrects).

Value

Ces fonctions renvoient une unique chaîne de caractères, au format HTML, qui peut être utilisée pour construire le texte d'un énoncé, d'une réponse affichée, d'un commentaire...

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

Examples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.affichages" )

# Catégorie : exemples
categorie.moodle( "Exemples SARP/Affichages" )

# Signe moins, décimale
vrai_faux.moodle( "Quelle est l'écriture correcte de l'opposé de cinq huitièmes?",
                  afficher_nombre.moodle( -5/8 ), -5/8 )

# Affichage des puissances
vrai_faux.moodle( paste( "L'inverse de", afficher_nombre.moodle( 10^8 ),
                          "est", afficher_nombre.moodle( 10^-8 ) ) )

# Affichage d'un échantillon
e <- rnorm( 10 ) * 2 + 20
numerique.moodle( paste0( "Donnez la plus petite valeur de l'échantillon ci-dessous.",
                           afficher_echantillon.moodle( e ) ),
                  min( e ) )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.affichages.xml dans Moodle pour voir le résultat...
```

```
print( "Importez le fichier exemple.affichages.xml dans Moodle pour voir le résultat..." )
```

categorie.moodle *Créer une catégorie de questions*

Description

Cette fonction permet de créer une nouvelle catégorie de questions dans le questionnaire XML.

Usage

```
categorie.moodle( nom.categorie, autoriser.dollar = FALSE,  
                 fichier.xml = get( "fichier.xml", envir = SARP.Moodle.env ) )
```

Arguments

`nom.categorie` Le nom de la catégorie. Il est possible de donner une hiérarchie de catégories, en utilisant la barre oblique (/) comme séparateur des différents niveaux — comme dans un chemin incluant des dossiers.

`autoriser.dollar` Le dollar (\$) est un caractère réservé, qui permet d'ancrer les catégories dans les divers cours. Pour éviter des difficultés, les dollars sont donc remplacés par des soulignés (_). Cette conversion peut être désactivée en indiquant TRUE pour cette option.

`fichier.xml` Le fichier correspondant au questionnaire qui contiendra cette question (par défaut, le dernier créé).

Details

La catégorie proposée est toujours intégrée dans le cours actuel, en ajoutant \$course\$ en début de hiérarchie.

Pour éviter des sous-catégories sans nom, les séparateurs multiples sont simplifiés en un seul séparateur et ceux en fin de nom de catégorie sont supprimés. Si vous voulez vraiment des sous-catégories sans nom visible, utilisez des espaces entre les séparateurs ou après le dernier.

Value

Aucune valeur n'est renvoyée par cette fonction.

Attention

Les catégories ne seront prises en compte que si l'option correspondante est cochée lors de l'import du fichier (case « Obtenir la catégorie à partir du fichier »)

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

Exemples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.categorie" )

# Catégorie des questions numériques pour les exemples de cette documentation
categorie.moodle( "Exemples SARP/Numériques" )
numerique.moodle( "Combien vaut 1 + 1&thinsp;?", 2 )
numerique.moodle( "Combien vaut 1 &times; 1&thinsp;?", 1 )
numerique.moodle( "Combien vaut 1 + 1 en binaire&thinsp;?", 10 )

# Catégorie des QROC pour les exemples de cette documentation
categorie.moodle( "Exemples SARP/QROC" )
qroc.moodle( "Comment s'appelle cette biblioth&egrave;que&thinsp;?", "SARP.Moodle" )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.categorie.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.categorie.xml dans Moodle pour voir le résultat..." )
```

 csv.moodle

Convertir un fichier CSV en questions Moodle

Description

Cette fonction permet de convertir un fichier CSV, convenablement construit, en un fichier XML de questions pour Moodle

Usage

```
csv.moodle(fichier.csv,
  colonne.texte = 'Question',
  colonne.reponse = 'R\u00e9ponse',
  colonne.note = NA, colonne.titre = NA,
  colonne.code = NA, colonne.type = NA,
  colonne.retour = NA, colonne.global = NA,
  fichier.xml = if ( TRUE == nv.fichier ) gsub(
    "\\.[Cc][Ss][Vv]$",
    ".xml", fichier.csv )
    else get( "fichier.xml",
      envir = SARP.Moodle.env ),
  nv.fichier = TRUE,
  creer.titre = TRUE, lg.titre = 30,
  embellir = TRUE, deja.HTML = FALSE,
  forcer.multiple = TRUE, melanger.reponses = TRUE,
  categorie.base = "",
  sep = ";", header = TRUE, quote = '\'',
  ... )
```

Arguments

<code>fichier.csv</code>	Un vecteur de chaînes de caractères contenant les noms des fichiers à convertir. Chacun des fichiers est traité indépendamment, mais avec les mêmes valeurs des options qui suivent.
<code>colonne.texte</code>	Le nom ou le numéro de la colonne qui contient l'énoncé des questions.
<code>colonne.reponse</code>	Le nom ou le numéro de la colonne qui contient la ou les réponses aux questions.
<code>colonne.note</code>	Le nom ou le numéro de la colonne qui contient la note associée à chaque réponse. Si elle n'est pas indiquée, une colonne intitulée « Note » (insensible à la casse) est cherchée dans le fichier. Cette colonne est obligatoire en cas de question à réponses multiples. Elle peut alors contenir soit le pourcentage de la note (voir les détails), soit « Vrai » ou « Faux » (insensible à la casse) et les pourcentages sont calculés en conséquence (avec toujours 0 pour « Faux »: indiquez explicitement un pourcentage négatif si nécessaire).
<code>colonne.titre</code>	Le nom ou le numéro de la colonne qui contient le titre à donner à la question dans Moodle. Si elle est manquante, le titre est construit, sous la forme <code>xx yy</code> où <code>yy</code> correspond aux <code>lg.titre</code> premiers caractères de la question, et <code>xx</code> au code de la question.
<code>colonne.code</code>	Le nom ou le numéro de la colonne qui contient le code de la question. Cette colonne n'est obligatoire que pour créer des questions « cloze ». Si elle est manquante, un code interne de la forme <code>[Qnn]</code> , où <code>nn</code> est le numéro de la question, est construit, chaque question correspondant à une ligne avec les deux colonnes <code>colonne.texte</code> et <code>colonne.reponse</code> renseignées. Voyez les détails pour davantage de précisions.
<code>colonne.type</code>	Le nom ou le numéro de la colonne qui contient le type de la question. Cette colonne peut être utilisée pour indiquer qu'une question avec plusieurs réponses proposées n'a qu'une seule réponse possible, en indiquant QCU pour au moins une des réponses (question à choix unique); QCM permet d'indiquer que l'on peut cocher plusieurs réponses (question à choix multiples). Dans ce cas, <code>forcer.multiple</code> est ignoré.
<code>colonne.retour</code>	Le nom ou le numéro de la colonne qui contient le commentaire à afficher pour chaque réponse. Cette colonne est facultative.
<code>colonne.global</code>	Le nom ou le numéro de la colonne qui contient le commentaire à afficher globalement pour la question. Cette colonne est facultative.
<code>fichier.xml</code>	Le nom du fichier XML à créer ou un fichier XML déjà créé avec debuter_xml.moodle . Par défaut, la fonction crée un fichier de même nom que le premier fichier fourni, en remplaçant l'extension <code>.csv</code> par <code>.xml</code> .
<code>nv.fichier</code>	Une valeur logique indiquant si l'on veut créer un nouveau fichier (TRUE) ou utiliser un fichier XML déjà ouvert avec debuter_xml.moodle (FALSE)
<code>creer.titre</code>	Une valeur logique indiquant si l'on veut créer un titre pour chaque question, à partir de son code et du début de son énoncé (TRUE) ou non. Ignoré si une colonne de titres a été fournie.
<code>lg.titre</code>	Le nombre de caractères de l'énoncé d'une question à conserver pour construire son titre. Ignoré si une colonne de titres a été fournie ou si <code>creer.titre=FALSE</code> .

<code>embellir,deja.HTML</code>	Actuellement inutilisés, prévus pour une version future
<code>forcer.multiple</code>	Si TRUE, les questions avec plusieurs réponses sont toujours considérées comme des réponses à choix multiples (l'étudiant pourra cocher plusieurs réponses). Sinon, si une seule bonne réponse est proposée (c'est-à-dire s'il existe une réponse donnant 100 % des points), la question est à choix unique (l'étudiant ne pourra choisir qu'une seule réponse). Cette option est ignoré pour les questions qui auraient une valeur indiquée dans la colonne précisée dans <code>colonne.type</code> .
<code>melanger.reponses</code>	Si TRUE, autorise Moodle a permuter aléatoirement l'ordre des réponses lorsqu'il pose la question. Sinon, l'ordre du fichier est tout le temps utilisé.
<code>categorie.base</code>	La catégorie de base à utiliser pour créer les questions. Elle contiendra toute les sous-catégories du fichier, s'il y en a.
<code>sep, header, quote</code>	Options pour <code>read.table</code> , correspondant au format d'un fichier CSV par défaut lorsqu'il est créé par Libre office ou Excel.
<code>...</code>	Autres options pour <code>read.table</code> , si nécessaire.

Details

Ces fonctions réalisent la conversion d'un ou plusieurs fichiers CSV de questions pour Moodle en un fichier XML. La nature des questions est déduite de la structure du fichier.

Ce fichier doit comporter au moins deux colonnes : l'une avec les énoncés des questions et l'autre avec les réponses. Chaque ligne correspond à une réponse possible.

Pour les questions simples, la question tient sur une ligne. La nature de la question est déduite de la réponse. Si la réponse peut être convertie en nombre, la question est supposée être une réponse numérique; elle sera créée avec un appel à `numerique.moodle`. Si la réponse est identifiée comme une valeur logique (un des textes « V », « F », « T », « Vrai », « Faux », « True » ou « False », quelle que soit la casse), la question est supposée être une question binaire avec les deux réponses « Vrai » et « Faux »; elle sera créée avec un appel à `vrai_faux.moodle`. Dans tous les autres cas, la question est supposée être à réponse ouverte et courte et sera créée par un appel à `qroc.moodle`.

S'il y a un énoncé, mais pas de réponse, cet énoncé est supposé être une indication de catégorie (sauf si la question a le même code qu'une question « cloze », voir plus loin) qui sera créée comme une sous-catégorie de la catégorie de base, par un appel à `categorie.moodle`.

S'il y a une réponse, mais pas d'énoncé, cette réponse est supposée être une autre réponse possible à une question à choix multiple, dont l'énoncé est dans la première ligne précédente qui en contient un. Dans ce cas, le fichier doit contenir une colonne qui contient les notes associées à chaque réponse. Cette note peut être précise, exprimée en pourcentage suivant la convention Moodle, ou indicative, avec la simple mention « Vrai » pour les bonnes réponses et « Faux » pour les mauvaises. Dans le deuxième cas, les mauvaises réponses sont associées à la note 0 et les bonnes à la fraction correspondante de bonne réponse, de sorte que cocher toutes les bonnes réponses donne la note totale.

Construire des questions “cloze”

Pour pouvoir construire des questions “cloze”, le fichier doit obligatoirement contenir une colonne qui indique le code de chaque question. Ce code doit être unique pour une même question; toutes les lignes qui auront le même code seront donc considérées comme faisant partie de la même question.

De ce fait, si plusieurs lignes d'énoncé renseignées ont le même code, elles sont considérées faisant partie de la même question et sont utilisées pour construire une question "cloze", selon le même principe que la fonction `question_libre.moodle`: après chaque partie d'énoncé, un champ de réponse est créé, dont la bonne réponse est celle indiquée dans la colonne `colonne.reponse` et sert à définir le type de champ de réponse (selon la logique des questions en une ligne). Pour avoir un texte après le dernier champ de réponse, laissez la colonne `colonne.reponse` tout en gardant le même code de question.

Value

De façon invisible, une liste des data.frames des questions après traitement de chaque fichier. Elle n'a guère d'autre intérêt que pour détecter l'origine d'un problème de conversion.

Attention!

Moodle est assez peu permissif pour les notes exprimées en pourcentage; en particulier, lorsque ce pourcentage ne tombe pas juste (33% pour trois bonnes réponses, par exemple), à cause de l'arrondi il peut rejeter le fichier créé. Dans ce cas, désactivez l'arrêt en cas d'erreur et demandez l'arrondi à la note la plus proche pour pouvoir importer les questions.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

Les fonctions de base de création de questions pour plus de souplesse, et en particulier pour créer des questions non gérées par le format de fichier ci-dessus.

Examples

```
# Conversion du fichier d'exemple fourni
# (à placer dans le répertoire de travail)
## Not run:
  csv.moodle( "exemple_Moodle.csv", colonne.code = "Code" )

## End(Not run)
```

csv_glossaire.moodle *Convertir un fichier CSV en glossaire Moodle*

Description

Cette fonction permet de convertir un fichier CSV, convenablement construit, en un fichier XML de glossaire pour Moodle

Usage

```

csv_glossaire.moodle( fichier.csv,
                      colonne.termes = "Mot",
                      colonne.definition = "D\u00e9finition",
                      fichier.xml = if ( TRUE == nv.fichier ) gsub(
                        "\\.[Cc][Ss][Vv]$",
                        ".xml", fichier.csv )
                        else get( "fichier.xml",
                                envir = SARP.Moodle.env ),
                      nv.fichier = TRUE,
                      embellir = TRUE, deja.HTML = FALSE,
                      sep = ";", header = TRUE, quote = "\"",
                      ... )

```

Arguments

fichier.csv	Un vecteur de chaînes de caractères contenant les noms des fichiers à convertir. Chacun des fichiers est traité indépendamment, mais avec les mêmes valeurs des options qui suivent.
colonne.termes	Le nom ou le numéro de la colonne qui contient les termes à définir dans le glossaire.
colonne.definition	Le nom ou le numéro de la colonne qui contient les définitions des termes du glossaire.
fichier.xml	Le nom du fichier XML à créer ou un fichier XML déjà créé avec creer_glossaire.moodle . Par défaut, la fonction crée un fichier de même nom que le premier fichier fourni, en remplaçant l'extension .csv par .xml.
nv.fichier	Une valeur logique indiquant si l'on veut créer un nouveau fichier (TRUE) ou utiliser un fichier XML déjà ouvert avec creer_glossaire.moodle (FALSE)
embellir,deja.HTML	Actuellement inutilisés, prévus pour une version future
sep, header, quote	Options pour read.table , correspondant au format d'un fichier CSV par défaut lorsqu'il est créé par Libre office ou Excel.
...	Autres options pour read.table , si nécessaire.

Details

Cette fonction réalise la conversion d'un ou plusieurs fichiers CSV de définitions d'un glossaire pour Moodle en un fichier XML.

Ce fichier doit comporter au moins deux colonnes : l'une avec les termes à définir et l'autre avec les définitions. Chaque ligne correspond à une définition.

Value

De façon invisible, une liste des data.frames des définitions après traitement de chaque fichier. Elle n'a guère d'autre intérêt que pour détecter l'origine d'un problème de conversion.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

Les fonctions de base de création de glossaire pour plus de souplesse.

Examples

```
# Conversion du fichier d'exemple fourni
# (à placer dans le répertoire de travail)
## Not run:
  csv_questionnaire.moodle( "exemple_glossaire.csv" )

## End(Not run)
```

debuter_xml.moodle *Commencer et terminer un questionnaire Moodle*

Description

Ces deux fonctions permettent de commencer et de terminer la génération d'un questionnaire Moodle au format XML.

Usage

```
debuter_xml.moodle( fichier.xml,
                    n.chiffres = 2, dec = ",",
                    racine = 2004197487,
                    glossaire = FALSE )

finir_xml.moodle(fichier.xml = get( "fichier.xml", envir = SARP.Moodle.env ))
```

Arguments

fichier.xml	Pour <code>debuter_xml.moodle</code> , le nom du fichier qui contiendra le questionnaire moodle au format XML. L'extension <code>.xml</code> est automatiquement rajoutée, sauf si elle est déjà présente dans ce nom. Pour <code>finir_xml.moodle</code> , le fichier correspondant au questionnaire à terminer (par défaut, le dernier créé).
dec	Le séparateur décimal à utiliser dans les textes. Par défaut, la virgule (comme utilisé en français). Il est modifié par options.
n.chiffres	Le nombre de décimales à utiliser pour les sorties au format XML. Attention, cela n'affecte que les valeurs numériques explicitement converties en texte par un appel à la fonction <code>afficher_nombre.moodle</code> .

racine	La racine (graine) à utiliser pour le générateur de nombres aléatoires, afin d'avoir une génération de questionnaires reproductible. Utilisez NA pour ne pas définir la graine.
glossaire	Si TRUE, le fichier XML créé est au format permettant de construire un glossaire. Sinon, c'est le format XML moodle « classique » qui est utilisé.

Details

Lorsque l'on débute un questionnaire avec `debuter_xml.moodle`, le fichier XML est créé avec son en-tête. Le descripteur de fichier est mémorisé dans l'environnement de la bibliothèque : toutes les fonctions ultérieures utiliseront ce fichier par défaut pour écrire les questions créées. Attention, en cas d'appels multiples pour travailler sur plusieurs questionnaires en parallèle, seul le dernier fichier créé est mémorisé.

Value

La fonction `debuter_xml.moodle` renvoie un descripteur de fichier correspondant au questionnaire nouvellement créé. Ce descripteur peut être utilisé par toutes les fonctions de la bibliothèque, mais aussi par la fonction `cat` de R directement, si nécessaire.

La fonction `finir_xml.moodle` ne renvoie rien.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

`options(OutDec)` pour le choix du séparateur décimal à l'affichage.

Examples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple" )

# Une question Moodle basique
vrai_faux.moodle( "Cette question a deux réponses" )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.xml dans Moodle pour voir le résultat..." )
```

 definir_dossier.image.moodle

Insérer une image dans une question

Description

Ces fonctions permettent de créer des liens vers des images dans les questions Moodle (texte, mais aussi réponses).

Usage

```
definir_dossier.image.moodle( URL )
```

```
lier_image.moodle( nom.image, largeur = -1, hauteur = -1,
                  description = NULL )
```

Arguments

URL	L'URL du dossier ou de l'activité Moodle contenant les images à lier. Voyez les détails pour plus d'information. Ce doit être un vecteur de type character, ne contenant qu'un seul élément.
nom.image	Le nom du fichier contenant l'image, tel qu'il apparaîtra dans le dossier moodle. Attention aux conversions de caractères que peut faire Moodle lors de l'import d'un fichier... Ce doit être un vecteur de type character contenant un seul élément.
largeur, hauteur	Les dimensions d'affichage de l'image dans la question (équivalent des champs height et width de la balise). Seules les dimensions numériques, finies, positives sont prises en compte. Toutes les autres valeurs sont ignorées et le champ correspondant n'est pas créé dans le code HTML. Ces deux valeurs doivent être des vecteurs numériques contenant un seul élément.
description	Un vecteur de chaîne de caractère, ne contenant qu'un seul élément. S'il existe, ce texte sera utilisé comme description textuelle de l'image (champ alt, fortement conseillé).

Details

La fonction `lier_image.moodle` crée une balise HTML `` avec une URL permettant de trouver l'image. Cette URL est obtenue en concaténant une racine et le nom du fichier image indiqué.

La racine peut être n'importe quel schéma d'URL valide. Le plus souvent, ce sera soit une URL vers un site externe qui contient les images soit, encore plus fréquent, une URL référant l'image dans Moodle.

La méthode la plus simple pour obtenir cette racine de l'URL est la suivante :

1. Créez une activité « Dossier » dans Moodle, cachée aux utilisateurs.
2. Dans ce dossier, importez la ou les images, puis enregistrez.

3. Affichez le contenu de ce dossier, comme un utilisateur.
4. Affichez le code source de la page HTML correspondance (Ctrl-U avec Firefox)
5. Dans ce code source, repérez le nom d'une image, dans un élément ``. Ce nom devrait ressembler à `https://votre.site.moodle/pluginfile.php/00000/mod_folder/content/0/votre_image.jpg?forcedownload` où 0000 est un nombre quelconque qui identifie ce dossier dans la base Moodle.
6. Mémorisez cette URL et utilisez-la comme base:

```
lier_image.moodle("https://votre.site.moodle/pluginfile.php/0000/mod_folder/content/0/" )
```

Toutes les images situées dans ce dossier (au moment de l'import du fichier XML ou ajoutée ultérieurement) sont alors directement utilisables dans la fonction `lier_image.moodle`, en indiquant juste leur nom.

Value

La fonction `lier_image.moodle` renvoie une chaîne de caractère qui contient le code HTML créé. Cette chaîne peut ensuite être utilisée, directement ou après concaténation avec d'autres éléments, dans la constitution des textes ou des réponses de n'importe quelle question moodle.

Attention

Vous devez appeler au moins une fois la fonction `definir_dossier.image.moodle` pour pouvoir ensuite lier des images. Au moment de la création du fichier XML, il n'y a aucun moyen de vérifier que l'URL indiquée est correcte...

Astuce

Pour importer plusieurs images en une fois, vous avez tout intérêt à les regrouper dans une archive au format zip, que vous décompresserez après l'avoir importée dans le dossier moodle ciblé.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

Examples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.image" )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.image.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.image.xml dans Moodle pour voir le résultat..." )
```

glossaire.moodle *Commencer et terminer la génération de glossaires Moodle*

Description

Ces fonctions permettent de préparer et de remplir un glossaire Moodle au format XML.

Usage

```
creer_glossaire.moodle( nom.fichier, nom.glossaire, texte.intro,  
                        doublons = TRUE )
```

```
entree_glossaire.moodle( terme, definition,  
                        fichier.xml = get( "fichier.xml", enviro = SARP.Moodle.env ) )
```

Arguments

nom.fichier	Une chaîne de caractères indiquant le nom du fichier XML, contenant le glossaire, à créer.
nom.glossaire	Une chaîne de caractères indiquant le nom du glossaire.
texte.intro	Une chaîne de caractères indiquant le texte décrivant le glossaire.
doublons	Une valeur logique indiquant si les doublons sont autorisés.
terme	Une chaîne de caractères indiquant le terme à insérer dans le glossaire.
definition	Une chaîne de caractères indiquant la définition du terme à insérer dans le glossaire.
fichier.xml	Le fichier correspondant au glossaire qui contiendra cette entrée (par défaut, le dernier créé).

Details

Ces fonctions permettent de préparer un glossaire Moodle au format XML.

Value

Aucune valeur n'est renvoyée par cette fonction.

Attention

Le support des glossaires au format XML est expérimental, en l'absence de documentation officielle.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

libre.moodle

Créer une question libre (« cloze », texte à compléter)

Description

Cette fonction permet de créer une question dont le format est libre, qui se présente sous forme d'un texte à trous — format dit « cloze ».

Usage

```
question_libre.moodle( texte.intro, textes.avant, texte.final,
                      reponses, notes = NULL, types = NULL,
                      commentaires = NULL,
                      titre = "Question libre",
                      commentaire.global = NA,
                      fichier.xml = get( "fichier.xml",
                                         envir = SARP.Moodle.env ) )
```

Arguments

<code>texte.intro</code>	L'introduction de la question (une chaîne de caractères). Cet énoncé peut contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML. Il se trouvera au tout début de la question.
<code>textes.avant</code>	Un vecteur de chaîne de caractères. À chaque élément de ce vecteur sera associé un champ de réponse, qui sera <i>précédé</i> du texte contenu dans cette chaîne de caractères. Ces textes peuvent contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML. Le nombre d'éléments du vecteur doit être égal au nombre d'éléments de reponses.
<code>texte.final</code>	Le texte qui doit se placer après le dernier champ de réponse de la question. Il peut contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML.
<code>reponses</code>	Une liste dont chaque élément décrit la ou les réponses attendues pour chaque « trou » du texte. Le nombre d'éléments de cette liste définit le nombre de champs réponses dans la question et doit être égal à la longueur du vecteur <code>textes.avant</code> . Consultez les détails pour plus d'information.
<code>notes</code>	Le vecteur des notes associées à chaque réponse. S'il est fourni, il doit être de même longueur que <code>reponses</code> .
<code>types</code>	Le vecteur des types de champ réponse. S'il est fourni, il doit être de même longueur que <code>reponses</code> . Les types possibles sont les textes NUMERICAL (par défaut), MULTICHOICE, MULTICHOICE_V, MULTICHOICE_H, SHORTANSWER et SHORTANSWER_C. Consultez les détails pour plus d'information.
<code>commentaires</code>	Une liste de commentaires associés aux réponses de chaque champ réponse. Si elle est fournie, elle doit être de même longueur que <code>reponses</code> . Consultez les détails pour plus d'information.
<code>titre</code>	L'intitulé de la question, en HTML.

<code>commentaire.global</code>	Le commentaire global affiché après avoir répondu à la question. Il n'y en aura pas s'il vaut NA, NULL ou est manquant.
<code>fichier.xml</code>	Le fichier correspondant au questionnaire qui contiendra cette question (par défaut, le dernier créé).

Details

Cette fonction construit, à partir des éléments fournis, une question de type « cloze » : un texte à trous, les réponses étant à entrer dans les trous ou à choisir dans une liste. C'est donc une suite de « champs réponses », séparés par des textes.

Chaque champ réponse est défini par la ou les réponses possibles, son type, sa note et éventuellement les commentaires renvoyés à l'étudiant en fonction de ses réponses. Ces informations sont obtenues à partir des paramètres ci-dessus, le *i*-ème élément étant utilisé pour le *i*-ème champ réponse.

Le type NUMERICAL correspond à un champ attendant une réponse numérique — c'est l'équivalent d'une question numérique. Une seule réponse est possible, qui sera la bonne réponse. Il ne peut donc y avoir aussi qu'un seul commentaire.

Les types MULTICHOICE, MULTICHOICE_V, MULTICHOICE_H, SHORTANSWER et SHORTANSWER_C correspondent à un champ permettant de choisir (cocher) une ou plusieurs réponses dans une liste prédéfinie — c'est l'équivalent d'un QCM. Ces cinq types fonctionnent de façon similaire, seule la façon d'afficher la question varie (elles donnent respectivement un menu déroulant, une série de boutons verticaux, une série de boutons horizontaux et un champ de texte libre insensible ou sensible à la casse). Dans les cinq cas, réponses doit contenir pour ces champs une liste de deux éléments, l'un contenant le vecteur des textes des réponses (élément nommé Textes), le second indiquant pour chaque texte s'il correspond à une réponse correcte (TRUE) ou non (FALSE). S'il y a plusieurs réponses correctes, chacune donne une fraction égale des points. Les autres cas de réponses donnant une partie des points ne sont pas encore possibles avec cette fonction. Les commentaires de chaque réponse sont à donner dans un vecteur de même longueur que `reponses$Textes`; utilisez NA pour une réponse sans commentaire.

Une fois le texte complet de la question créé, au format « cloze », à partir de ces éléments, la question est créée grâce à la fonction [question.moodle](#).

Value

Aucune valeur n'est renvoyée par cette fonction.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

[numerique.moodle](#), [qcm.moodle](#), [groc.moodle](#), [question_ouverte.moodle](#), [vrai_faux.moodle](#)

Examples

```

# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.libre" )

# Catégorie : exemples
categorie.moodle( "Exemples SARP/Question libre" )

# Une question Moodle libre avec champs numériques
question_libre.moodle( "Indiquez les chiffres du nombre 1974.",
  list( "Unit&eacute; : ", " &mdash; dizaine : ",
        " &mdash; centaine : ", " &mdash; milliers : " ), ".",
  reponses = list( 4, 7, 9, 1 ) )

# Avec un champ numérique et des champs à choix multiples
question_libre.moodle( "<cite>Le Corbeau et le Renard</cite>.",
  list( "<i>Ma&iacirc;tre ",
        " sur son arbre perch&eacute;<br />Tenait en son bec un ",
        ".<br />Ma&iacirc;tre ",
        "</i>par l'odeur all&eacute;ch&eacute;<br />...",
        " est un po&egrave;me compos&eacute; en " ),
  " par le po&egrave;te Jean de la Fontaine.",
  types = c( "MULTICHOICE", "MULTICHOICE_H",
             "SHORTANSWER", "NUMERICAL" ),
  reponses = list( list( "Textes" = c( "Renard", "Goupil",
                                       "Corbeau", "Corneille" ),
                    "Correct" = c( FALSE, FALSE, TRUE, FALSE ) ),
                  list( "Textes" = c( "Fromage", "Camembert",
                                       "Bleu", "Livre" ),
                    "Correct" = c( TRUE, FALSE, FALSE, FALSE ) ),
                  list( "Textes" = c( "Renard", "Goupil",
                                       "Corbeau", "Corneille" ),
                    "Correct" = c( TRUE, FALSE, FALSE, FALSE ) ),
                  1668 ) )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.libre.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.libre.xml dans Moodle pour voir le r\u00e9sultat..." )

```

numerique.moodle

Créer une question à réponse numérique simple

Description

Cette fonction permet de créer une question dont la réponse est une valeur numérique unique.

Usage

```
numerique.moodle( texte, bonne.reponse, etendue = NULL,
                  n.decimales = get( "nombre.chiffres", envir = SARP.Moodle.env ),
                  titre = "Question num&eacute;rique...",
                  fichier.xml = get( "fichier.xml", envir = SARP.Moodle.env ),
                  commentaire.global = NA )
```

Arguments

<code>texte</code>	L'énoncé de la question (une chaîne de caractères). Cet énoncé peut contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML.
<code>bonne.reponse</code>	La valeur numérique correspondant à la bonne réponse.
<code>etendue</code>	Actuellement inutilisé
<code>n.decimales</code>	Le nombre de décimales à conserver pour la bonne réponse (utiliser NA pour ne pas arrondir); par défaut, la valeur indiquée à la création du questionnaire est utilisée.
<code>titre</code>	L'intitulé de la question, en HTML.
<code>fichier.xml</code>	Le fichier correspondant au questionnaire qui contiendra cette question (par défaut, le dernier créé).
<code>commentaire.global</code>	Le commentaire global affiché après avoir répondu à la question. Il n'y en aura pas s'il vaut NA, NULL ou est manquant.

Details

Dans la version actuelle, la tolérance n'est pas prise en compte. La bonne réponse doit donc être donnée exactement: n'oubliez pas d'arrondir la valeur au bon nombre de décimales, soit en utilisant l'option `n.decimales`, soit avant d'appeler la fonction.

L'arrondi est fait avec la fonction `round` de R. Dans ce cas, un message est automatiquement ajouté à la fin du texte de la question, précisant le nombre de décimales attendu. Ce texte est sur sa propre ligne (précédé d'une balise HTML `
`) et en italique (balises HTML `<i>` et `</i>`).

Cette fonction est une interface simplifiée, et enrichie, pour la fonction [question.moodle](#).

Value

Aucune valeur n'est renvoyée par cette fonction.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

[qcm.moodle](#), [qroc.moodle](#), [question_ouverte.moodle](#), [vrai_faux.moodle](#)

Exemples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.numerique" )

# Catégorie : exemples
categorie.moodle( "Exemples SARP/Numérique" )

# Une question Moodle numérique, avec 3 décimales
numerique.moodle( "Donnez la valeur de &pi;", pi, n.decimales = 3 )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.numerique.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.numerique.xml dans Moodle pour voir le résultat..." )
```

ouverte.moodle	<i>Créer une question à réponse ouverte (rédactionnelle)</i>
----------------	--

Description

Cette fonction permet de créer une question dont la réponse est libre (rédactionnelle).

Usage

```
question_ouverte.moodle( texte,
                        titre = "Question rédactionnelle ouverte",
                        fichier.xml = get( "fichier.xml", enviro = SARP.Moodle.env ) )
```

Arguments

texte	L'énoncé de la question (une chaîne de caractères). Cet énoncé peut contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML.
titre	L'intitulé de la question, en HTML.
fichier.xml	Le fichier correspondant au questionnaire qui contiendra cette question (par défaut, le dernier créé).

Details

Cette fonction est une interface simplifiée pour la fonction [question.moodle](#).

Value

Aucune valeur n'est renvoyée par cette fonction.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

[numerique.moodle](#), [qcm.moodle](#), [qroc.moodle](#), [vrai_faux.moodle](#)

Examples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.ouverte" )

# Catégorie : exemples
categorie.moodle( "Exemples SARP/Question ouverte" )

# Une question Moodle ouverte
question_ouverte.moodle( "Racontez votre journ&eacute;e." )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.ouverte.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.ouverte.xml dans Moodle pour voir le résultat..." )
```

qcm.moodle

Créer une question à choix (réponse unique ou réponses multiples)

Description

Cette fonction permet de créer une question de type QCM ou QCU, dans laquelle la ou les bonnes réponses sont à cocher dans une liste de réponses prédéfinie.

Usage

```
qcm.moodle( texte, bonnes.reponses, mauvaises.reponses,
            commentaires = NULL, fractions = list( "Bonnes" = NULL, "Fausses" = NULL ),
            unique = ( length( bonnes.reponses ) == 1 ), melanger = TRUE,
            titre = "QCM...",
            numerotation = c( "none", "abc", "ABCD", "123" ),
            fichier.xml = get( "fichier.xml", envir = SARP.Moodle.env ) )
```

Arguments

texte	L'énoncé de la question (une chaîne de caractères). Cet énoncé peut contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML.
bonnes.reponses	Un vecteur de chaînes de caractères correspondant aux intitulés des bonnes réponses. Il doit y avoir au moins une bonne réponse.
mauvaises.reponses	Un vecteur de chaînes de caractères correspondant aux intitulés des mauvaises réponses.

commentaires	Un vecteur de chaînes de caractère, correspondant au retour à indiquer pour chacune des réponses, dans l'ordre dans lequel elles sont passées, en commençant par les bonnes réponses. Utilisez NA pour laisser un commentaire vide.
fractions	<p>Une liste de deux éléments, contenant les pourcentage de note associés à chaque réponse. Ces deux éléments doivent être intitulés Bonnes (pour les bonnes réponses) et Fausses (pour les mauvaises réponses) ; s'ils sont indiqués, ce doivent être des vecteurs numériques ayant le même nombre d'éléments que la liste des réponses. Par défaut (si l'élément vaut NULL), chaque bonne réponse est affectée de la même fraction de note, de sorte que l'on atteigne 100 % en cochant toutes les bonnes réponses. De la même façon, les mauvaises réponses sont affectées d'une fraction négative de sorte que cocher toutes les mauvaises réponses annule le fait de cocher toutes les bonnes réponses, sans donner une note négative.</p> <p>Les fractions doivent être comprises entre 0 (exclus) et 100 pour les bonnes réponses ; elles doivent être négatives pour les mauvaises réponses. Si ce n'est pas le cas, ou si la somme des fractions des bonnes réponse ne vaut pas 100, un avertissement est donné, mais la question est tout de même créée. Toutefois, si toutes les fractions des mauvaises réponses sont positives, elles sont considérées négatives sans avertissement.</p>
unique	Si TRUE, indique que l'on ne peut cocher qu'une seule bonne réponse.
melanger	Si TRUE, indique que l'ordre des réponses doit être choisi aléatoirement à chaque affichage de la question. Sinon, l'ordre est celui défini dans le fichier. Actuellement, cela correspond à l'ensemble des bonnes réponses dans l'ordre indiqué, suivi de l'ensemble des mauvaises réponses dans l'ordre indiqué.
titre	L'intitulé de la question, en HTML.
numerotation	Le type de numérotation à utiliser pour les différentes réponses. Ces numéros sont ajoutés par Moodle lors de l'affichage de la question. Seules les quatre valeurs indiquées sont reconnues par Moodle. Vous pouvez abrégé en n'utilisant que la première lettre.
fichier.xml	Le fichier correspondant au questionnaire qui contiendra cette question (par défaut, le dernier créé).

Details

La question générée avec cette fonction est une question de type multichoice.

Cette fonction est une interface simplifiée pour la fonction [question.moodle](#).

Value

Aucune valeur n'est renvoyée par cette fonction.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

[numerique.moodle](#), [qroc.moodle](#), [question_ouverte.moodle](#), [vrai_faux.moodle](#)

Exemples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.QCM" )

# Catégorie : exemples
categorie.moodle( "Exemples SARP/QCM" )

# Le format par défaut, avec deux bonnes réponses et trois mauvaises,
qcm.moodle( "Quels sont les fruits de la liste ci-dessous&thinsp;?",
            c( "Orange", "Tomate" ), c( "Pomme de terre", "Carotte", "Navet" ) )

# Deux bonnes réponses, deux mauvaises réponses, avec commentaires
# et fractions inégales
qcm.moodle( "Indiquez les romanciers de la liste ci-dessous.",
            c( "Eugène&nbsp;S<small>UE</small>", "Alexandre&nbsp;D<small>UMAS</small>" ),
            c( "Labiche", "Ronsard" ),
            commentaires = c( "Bravo&thinsp;!", NA,
                              "C'est un auteur de pièces de théâtre", "C'est un poète" ),
            fractions = list( "Bonnes" = c( 75, 25 ),
                              "Fausses" = c( -50, -100 ) ) )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.QCM.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.QCM.xml dans Moodle pour voir le résultat..." )
```

qroc.moodle

Créer une question à réponse ouverte, courte

Description

Cette fonction permet de créer une question dont la réponse est un court texte (« QROC »)

Usage

```
qroc.moodle( texte, reponses, notes = rep( 100, length( reponses ) ),
            commentaires = NULL, casse = TRUE,
            titre = "QROC...",
            fichier.xml = get( "fichier.xml", envir = SARP.Moodle.env ) )
```

Arguments

texte	L'énoncé de la question (une chaîne de caractères). Cet énoncé peut contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML.
reponses	Un vecteur de chaînes de caractères, dont chaque élément correspond à une réponse correcte possible.

notes	Un vecteur de nombres, correspondant aux pourcentages de la note totale associés à chaque réponse. Par défaut, toutes les réponses sont supposées parfaitement correctes.
commentaires	Un vecteur de chaînes de caractères, correspondant au retour à afficher pour chacune des réponses entrées (dans le même ordre que les réponses). Le vecteur doit avoir la même longueur ; si certaines réponses n'appellent pas de commentaire, utilisez NA.
casse	Si TRUE, on distingue entre majuscules et minuscules pour savoir si la réponse est correcte. Sinon, cette distinction n'est pas faite.
titre	L'intitulé de la question, en HTML.
fichier.xml	Le fichier correspondant au questionnaire qui contiendra cette question (par défaut, le dernier créé).

Details

Cette fonction est une interface simplifiée, et enrichie, pour la fonction [question.moodle](#).

Value

Aucune valeur n'est renvoyée par cette fonction.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

[numerique.moodle](#), [qcm.moodle](#), [question_ouverte.moodle](#), [vrai_faux.moodle](#)

Examples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.qroc" )

# Catégorie : exemples
categorie.moodle( "Exemples SARP/QROC" )

# Une question Moodle en QROC, avec une seule bonne réponse
qroc.moodle( "Quel est le prénom de Victor&nbsp;H<small>UGO</small>&thinsp;?",
             "Victor" )

# La même, en indiquant que l'initiale doit être en majuscule
qroc.moodle( "Quel est le prénom de Victor&nbsp;H<small>UGO</small>&thinsp;?",
             c( "Victor", "victor" ),
             notes = c( 100, 50 ),
             commentaires = c( NA, "N'oubliez pas la majuscule aux noms propres!" ) )

# On termine le questionnaire
finir_xml.moodle( )
```

```
# Importez le fichier exemple.numerique.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.numerique.xml dans Moodle pour voir le résultat..." )
```

question.moodle

Créer une question Moodle au format XML

Description

Cette fonction permet de créer une question pour Moodle au format XML.

Usage

```
question.moodle( type = "cloze",
                 titre = "Question...", texte, reponses = NULL,
                 penalite = NA, note = NA, commentaire.global = NA,
                 autres.codes = NULL,
                 fichier.xml = get( "fichier.xml", envir = SARP.Moodle.env ) )
```

Arguments

texte	Le texte de la question (une chaîne de caractères). Cet énoncé peut contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML. Il correspond à l'élément entre les balises <questiontext> et </questiontext> de la question. C'est le seul argument obligatoire.
type	Le type de question à créer (élément type de la balise <question>). Vecteur contenant une seule chaîne de caractères.
titre	L'intitulé de la question, en HTML. Il correspond à l'élément entre les balises <name> et </name> de la question. Indiquez NULL pour ne pas donner de titre.
reponses	Un vecteur. Chaque élément correspond à une réponse possible, c'est-à-dire à un champ délimité par les balises <answer> et </answer>. <i>Attention</i> , les valeurs de ce vecteur sont indiquées telles qu'elles dans le fichier XML, sans être protégées : n'utilisez des formats particuliers qu'avec précaution, ou en les protégeant manuellement par une balise <![CDATA[. . .]]> — ce que font les fonctions de création de question prédéfinies, si nécessaire. Pour indiquer des commentaires (champs <feedback>), créez un vecteur de chaînes de caractères de même longueur que celui des réponses et associez-le aux réponses comme attribut, de nom commentaire. Utilisez NA pour laisser un commentaire vide. Par défaut, la première réponse est considérée comme parfaitement correcte (100% de la note) et toutes les autres comme fausses (0%). Pour indiquer des fractions pour chaque note, créez un vecteur numérique de même longueur que le nombre de réponses et associez-le aux réponses comme attribut, de nom fractions.
commentaire.global	Un vecteur de chaînes de caractère, de longueur 1, correspondant au retour à indiquer globalement (champ délimité par les balises <generalfeedback> et </generalfeedback>). Utilisez NA pour ne pas indiquer de commentaire.

note	La note associée à cette question (champ délimité par les balises <defaultgrade> et </defaultgrade>). Doit être un vecteur numérique avec un seul élément. Utilisez NA pour ne pas indiquer de note.
penalite	La pénalité à utiliser en cas de tentatives multiples, si le premier essai est incorrect (champ délimité par les balises <penalty> et </penalty>). Doit être un vecteur numérique avec un seul élément. Utilisez NA pour ne pas indiquer de pénalité.
autres.codes	Un vecteur de chaînes de caractère, correspondant aux valeurs à donner à des champs optionnels de la question. Ils sont écrits dans l'ordre. Le nom des champs est déterminé à partir du nom des éléments du vecteur.
fichier.xml	Le fichier correspondant au questionnaire qui contiendra cette question (par défaut, le dernier créé).

Details

Cette fonction est la fonction principale, appelée par toutes les autres fonctions de création d'une question Moodle. Les autres fonctions étant d'usage plus simple, n'utilisez cette fonction que pour des cas non-prévus par les fonctions spécialisées.

Value

Aucune valeur n'est renvoyée par cette fonction.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

[numerique.moodle](#), [qcm.moodle](#) [qroc.moodle](#), [question_libre.moodle](#), [question_ouverte.moodle](#), [vrai_faux.moodle](#)

Examples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.question" )

# Catégorie : exemples
categorie.moodle( "Exemples SARP/Question" )

# Une question vrai-faux (en utilisant le type binaire)
reponses <- c( "Vrai", "Faux" )
attr( reponses, "commentaire" ) <- c( "Bravo !", "Elle n'est pas plate" )
attr( reponses, "fractions" ) <- c( 100, 0 )
question.moodle( type = "truefalse", titre = "Question binaire",
                 texte = "La Terre est ronde", reponses = reponses )

# On termine le questionnaire
finir_xml.moodle( )
```

```
# Importez le fichier exemple.question.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.question.xml dans Moodle pour voir le résultat..." )
```

SARP.moodle

Création de questions Moodle au format XML avec R

Description

Cette bibliothèque permet de générer un fichier de questions pour Moodle, au format XML, sans avoir à connaître ce format. Les questions peuvent être construites à partir d'un fichier CSV ou directement à partir de fonctions R.

Details

Cette bibliothèque permet de faciliter la création de questions pour les tests Moodle : elle génère un fichier XML qui peut être importé dans la base de questions d'un cours Moodle. Il est ensuite possible de piocher dans ces questions pour créer un test. Afin d'éviter toute confusion avec d'autres fonctions, toutes les fonctions de cette bibliothèque se terminent par le suffixe `.moodle`.

L'utilisation la plus simple est la conversion d'un fichier CSV (ou tout autre format texte lisible par `read.table`) avec une ligne par question au format XML (`csv.moodle`). Mais il est aussi possible de créer les questions directement dans R, en créant le fichier XML (`debuter_xml.moodle`) puis en appelant les fonctions appropriées. Dans ce cas, il ne faut pas oublier de clore le fichier avant de l'importer (`finir_xml.moodle`).

La plupart des types de questions de base de Moodle sont connus : vrai-faux (`vrai_faux.moodle`), questions à choix parmi des réponses prédéfinies (`qcm.moodle`), questions à réponse numérique (`numerique.moodle`), questions à réponse libre (`qroc.moodle`, `question_ouverte.moodle`) et le format libre permettant de poser plusieurs questions en une (`question_libre.moodle`). Il est possible de générer un type de question non explicitement supporté grâce à la fonction générique `question.moodle`, à condition que vous connaissiez les noms des balises XML nécessaires.

Il est possible de trier les questions en catégories afin de les repérer plus facilement dans la base de question (`categorie.moodle` ; c'est indispensable si vous prévoyez dans votre test le tirage au sort d'une question parmi une série de questions similaires)

Afin de faciliter la construction des énoncés des questions, quelques fonctions sont proposées pour convertir des objets fréquemment utilisés en code HTML tout en respectant les conventions typographiques françaises : nombres (`afficher_nombre.moodle`), échantillons de valeurs (vecteur de nombres, `afficher_echantillon.moodle`, construisant un tableau).

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

Description

Cette fonction permet de créer une question de type vrai ou faux.

Usage

```
vrai_faux.moodle( texte, texte.vrai = "Vrai", texte.faux = "Faux",
                 commentaires = NULL,
                 titre = "Question vrai-faux...",
                 ordre = c( "aleatoire", NA, "random", "vrai premier", "faux premier" ),
                 melanger = FALSE,
                 fichier.xml = get( "fichier.xml", envir = SARP.Moodle.env ) )
```

Arguments

texte	L'énoncé de la question (une chaîne de caractères). Cet énoncé peut contenir n'importe quel caractère (codage en UTF-8) et n'importe quelles balises HTML.
texte.vrai	Le texte à utiliser pour la bonne réponse.
texte.faux	Le texte à utiliser pour la mauvaise réponse.
commentaires	Un vecteur de deux chaînes de caractère, correspondant au retour à indiquer pour la bonne réponse (premier élément) et pour la mauvaise réponse (second élément). Utilisez NA pour laisser un commentaire vide.
titre	L'intitulé de la question, en HTML.
ordre	L'ordre dans lequel les deux réponses possibles doivent apparaître. Les trois premières possibilités décident d'un ordre aléatoire (fixe). Cette option n'a d'intérêt que si melanger=FALSE...
melanger	Si TRUE, indique que Moodle devra tirer au sort l'ordre d'affichage des réponses (qui changera donc à chaque fois que la question sera posée). Sinon, l'ordre sera toujours celui indiqué lors de la création de la question.
fichier.xml	Le fichier correspondant au questionnaire qui contiendra cette question (par défaut, le dernier créé).

Details

Bien qu'un type particulier de question existe dans Moodle pour les questions « vrai ou faux » (type truefalse), la question générée avec cette fonction est une question de type multichoice, afin d'avoir la liberté des intitulés des bonnes et mauvaises réponses.

Cette fonction est une interface simplifiée pour la fonction [question.moodle](#).

Value

Aucune valeur n'est renvoyée par cette fonction.

Author(s)

Emmanuel Curis <emmanuel.curis@parisdescartes.fr>

See Also

[numerique.moodle](#), [qcm.moodle](#), [qroc.moodle](#), [question_ouverte.moodle](#).

Examples

```
# Commencer un nouveau questionnaire
q <- debuter_xml.moodle( "exemple.VF" )

# Catégorie : exemples
categorie.moodle( "Exemples SARP/Vrai ou Faux" )

# Le format par défaut
vrai_faux.moodle( "Une question « vrai ou faux » offre le choix entre deux réponses" )

# et si la bonne réponse est « faux »
vrai_faux.moodle( "Le Soleil tourne autour de la Terre", "Faux", "Vrai" )

# avec des intitulés libres
vrai_faux.moodle( "La bataille de Marignan a eu lieu en...", "1515", "1914" )

# On termine le questionnaire
finir_xml.moodle( )

# Importez le fichier exemple.VF.xml dans Moodle pour voir le résultat...
print( "Importez le fichier exemple.VF.xml dans Moodle pour voir le résultat..." )
```

Index

*Topic **package**

- SARP.moodle, 26
- affichage.moodle, 2
- afficher_echantillon.moodle, 26
- afficher_echantillon.moodle (affichage.moodle), 2
- afficher_nombre.moodle, 26
- afficher_nombre.moodle (affichage.moodle), 2
- cat, 11
- categorie.moodle, 4, 7, 26
- conversion_csv.moodle (csv.moodle), 5
- creer_glossaire.moodle, 9
- creer_glossaire.moodle (glossaire.moodle), 14
- csv.moodle, 5, 26
- csv_glossaire.moodle, 8
- debuter_xml.moodle, 6, 10, 26
- definir_dossier.image.moodle, 12
- entree_glossaire.moodle (glossaire.moodle), 14
- finir_xml.moodle, 26
- finir_xml.moodle (debuter_xml.moodle), 10
- glossaire.moodle, 14
- libre.moodle, 15
- lier_image.moodle (definir_dossier.image.moodle), 12
- numerique.moodle, 7, 16, 17, 20, 21, 23, 25, 26, 28
- options, 11
- ouverte.moodle, 19
- qcm (qcm.moodle), 20
- qcm.moodle, 16, 18, 20, 20, 23, 25, 26, 28
- qroc (qroc.moodle), 22
- qroc.moodle, 7, 16, 18, 20, 21, 22, 25, 26, 28
- question.moodle, 16, 18, 19, 21, 23, 24, 26, 27
- question_libre.moodle, 8, 25, 26
- question_libre.moodle (libre.moodle), 15
- question_ouverte.moodle, 16, 18, 21, 23, 25, 26, 28
- question_ouverte.moodle (ouverte.moodle), 19
- read.table, 7, 9, 26
- round, 18
- SARP.moodle, 26
- SARP.moodle-package (SARP.moodle), 26
- vf.moodle, 27
- vrai_faux.moodle, 7, 16, 18, 20, 21, 23, 25, 26
- vrai_faux.moodle (vf.moodle), 27