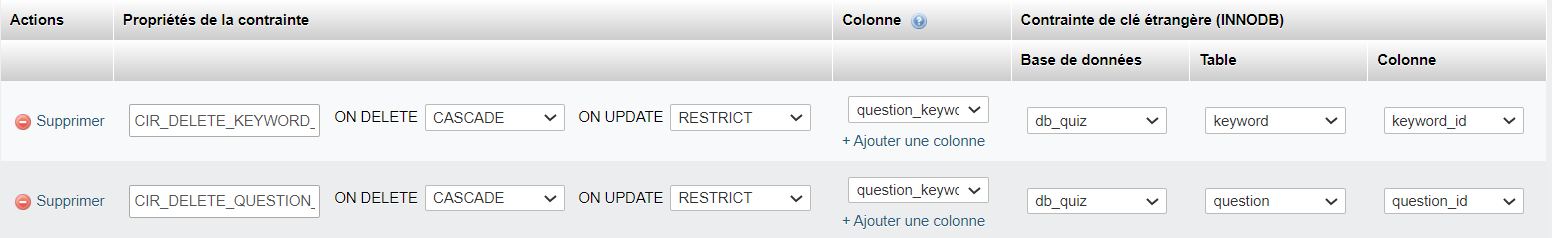
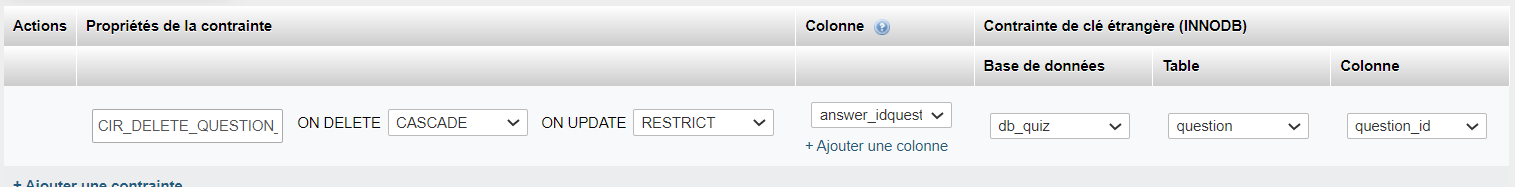
**question\_keyword :**

ALTER TABLE `question\_keyword` ADD CONSTRAINT `CIR\_DELETE\_KEYWORD\_QUESTION-KEYWORD` FOREIGN KEY (`question\_keyword\_idkeyword`) REFERENCES `keyword`(`keyword\_id`) ON DELETE CASCADE ON UPDATE RESTRICT; ALTER TABLE `question\_keyword` ADD CONSTRAINT `CIR\_DELETE\_QUESTION\_QUESTION-KEYWORD` FOREIGN KEY (`question\_keyword\_idquestion`) REFERENCES `question`(`question\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

****

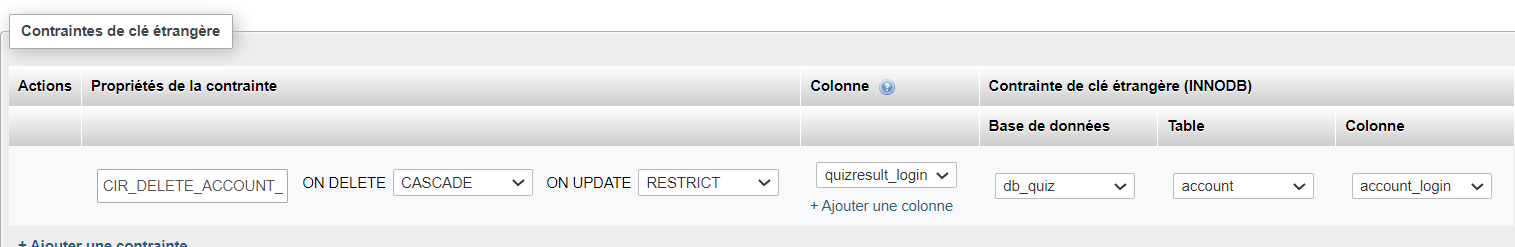
**answer :**

ALTER TABLE `answer` ADD CONSTRAINT `CIR\_DELETE\_QUESTION\_ANSWER` FOREIGN KEY (`answer\_idquestion`) REFERENCES `question`(`question\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

****

**quizresult :**

ALTER TABLE `quizresult` ADD CONSTRAINT `CIR\_DELETE\_ACCOUNT\_QUIZRESULT` FOREIGN KEY (`quizresult\_loginuser`) REFERENCES `account`(`account\_login`) ON DELETE CASCADE ON UPDATE RESTRICT;



**session\_user :**

ALTER TABLE `session\_user` ADD CONSTRAINT `CIR\_DELETE\_ACCOUNT\_SESSION-USER` FOREIGN KEY (`session\_user\_loginuser`) REFERENCES `account`(`account\_login`) ON DELETE CASCADE ON UPDATE RESTRICT;

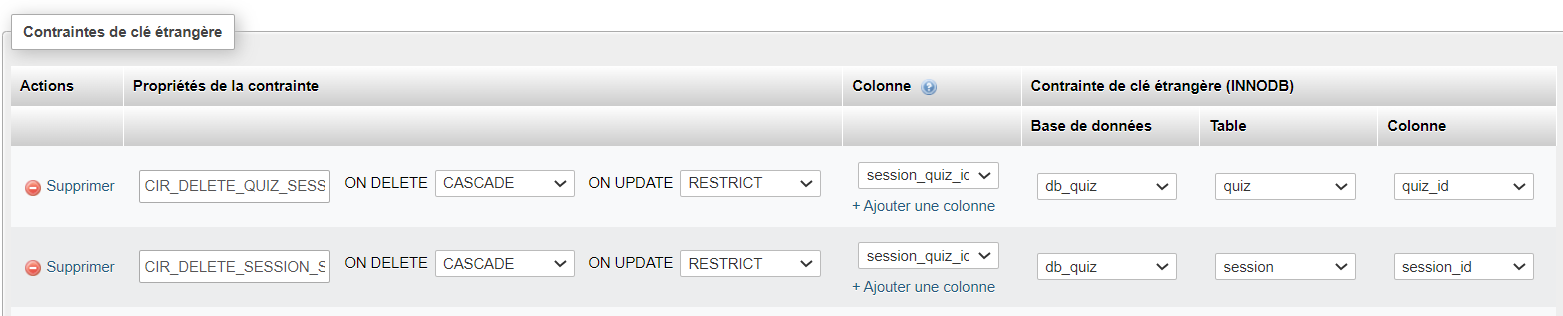
ALTER TABLE `session\_user` ADD CONSTRAINT `CIR\_DELETE\_SESSION\_SESSION-USER` FOREIGN KEY (`session\_user\_idsession`) REFERENCES `session`(`session\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;



**session\_quiz :**

ALTER TABLE `session\_quiz` ADD CONSTRAINT `CIR\_DELETE\_SESSION\_SESSION-QUIZ` FOREIGN KEY (`session\_quiz\_idsession`) REFERENCES `session`(`session\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

ALTER TABLE `session\_quiz` ADD CONSTRAINT `CIR\_DELETE\_QUIZ\_SESSION-QUIZ-QUESTION` FOREIGN KEY (`session\_quiz\_idquiz`) REFERENCES `quiz`(`quiz\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;



**quizresult\_ questionko :**

ALTER TABLE `quizresult\_questionko` ADD CONSTRAINT `CIR\_DELETE\_QUESTION\_QUIZRESULT-QUESTIONKO` FOREIGN KEY (`quizresult\_questionko\_idquestion`) REFERENCES `question`(`question\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

ALTER TABLE `quizresult\_questionko` ADD CONSTRAINT `CIR\_DELETE\_QUIZRESULT\_QUIZRESULT-QUESTIONKO` FOREIGN KEY (`quizresult\_questionko\_idquizresult`) REFERENCES `quizresult`(`quizresult\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

**Une image contenant texte, capture d’écran, différent

Description générée automatiquement**

**questionstat :**

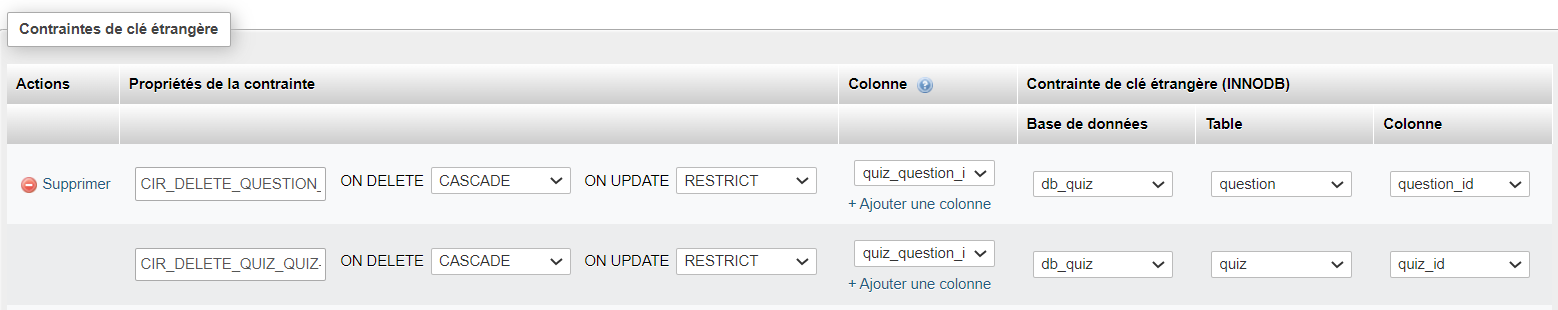
ALTER TABLE `questionstat` ADD CONSTRAINT `CIR\_DELETE\_QUESTION\_QUESTIONSTATSWER` FOREIGN KEY (`questionstat\_idquestion`) REFERENCES `question`(`question\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

****

**quiz\_question :**

ALTER TABLE `quiz\_question` ADD CONSTRAINT `CIR\_DELETE\_QUESTION\_QUIZ-QUESTION` FOREIGN KEY (`quiz\_question\_idquestion`) REFERENCES `question`(`question\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

ALTER TABLE `quiz\_question` ADD CONSTRAINT `CIR\_DELETE\_QUIZ\_QUIZ-QUESTION` FOREIGN KEY (`quiz\_question\_idquiz`) REFERENCES `quiz`(`quiz\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

****

ALTER TABLE `quizlock` ADD CONSTRAINT `CIR\_DELETE\_QUIZ\_QUIZLOCK` FOREIGN KEY (`quizlock\_quizid`) REFERENCES `quiz`(`quiz\_id`) ON DELETE CASCADE ON UPDATE RESTRICT;

ALTER TABLE `quizlock` ADD CONSTRAINT `CIR\_DELETE\_ACCOUNT\_QUIZLOCK` FOREIGN KEY (`quizlock\_user`) REFERENCES `account`(`account\_login`) ON DELETE CASCADE ON UPDATE RESTRICT;

Une image contenant texte

Description générée automatiquement

**Cannot drop table 'account' referenced by a foreign key constraint 'CIR\_DELETE\_ACCOUNT\_QUIZRESULT' on table 'quizresult'.**

ALTER TABLE quizlock ADD PRIMARY KEY (quizlock\_user, quizlock\_quizid);

ALTER TABLE session\_quiz ADD PRIMARY KEY (`session\_quiz\_idsession`, `session\_quiz\_idquiz`);

ALTER TABLE question\_keyword ADD PRIMARY KEY (`question\_keyword\_idquestion`, `question\_keyword\_idkeyword`);

ALTER TABLE quizresult\_questionko ADD PRIMARY KEY (`quizresult\_questionko\_idquizresult`, `quizresult\_questionko\_idquestion`);

ALTER TABLE quiz\_question ADD PRIMARY KEY (`quiz\_question\_idquiz`, `quiz\_question\_idquestion`);

ALTER TABLE session\_user ADD PRIMARY KEY (`session\_user\_idsession`, `session\_user\_loginuser`);

<button class="button" type="button" id="up\_question-keywords-restor" onclick="up\_questionKeywordsRestor(<?php echo($strKeywords) ?>)">Restaurer les mots clés d'origine</button>

echo '<button class="button" type="button" id="up\_question-keywords-restor" onclick="up\_questionKeywordsRestor("'.$strKeywords.'")">Restaurer les mots clés d\'origine</button>';

**-----**

**class\_questions\_list / \_\_construct :**

SELECT question\_id, question\_question, question\_status, question\_lastmodificationdate, question\_idwidget,

question\_keyword\_idquestion, question\_keyword\_idkeyword

FROM question LEFT JOIN question\_keyword ON question\_keyword\_idquestion = question\_id

WHERE question\_loginadmin = 'cma'

ORDER BY question\_lastmodificationdate DESC

SELECT COUNT(y.question\_id) FROM (SELECT question\_id, question\_keyword\_idquestion

FROM question LEFT JOIN question\_keyword ON question\_keyword\_idquestion = question\_id

WHERE question\_loginadmin = 'cma') AS y >>>> 5

SELECT COUNT(y.question\_id) FROM (SELECT DISTINCT question\_id

FROM question LEFT JOIN question\_keyword ON question\_keyword\_idquestion = question\_id

WHERE question\_loginadmin = 'cma') AS y >>>> 3

SELECT COUNT(y. question\_keyword\_idquestion) FROM (SELECT question\_id, question\_keyword\_idquestion

FROM question LEFT JOIN question\_keyword ON question\_keyword\_idquestion = question\_id

WHERE question\_loginadmin = 'cma') AS y >>>> 4

* 5 – 4 = 1 NULL

SELECT COUNT(y. question\_keyword\_idquestion) FROM (SELECT DISTINCT question\_keyword\_idquestion

FROM question LEFT JOIN question\_keyword ON question\_keyword\_idquestion = question\_id

WHERE question\_loginadmin = 'cma') AS y >>>> 2

* 3 – 2 = 1 NULL

SELECT DISTINCT keyword\_id , keyword\_word

FROM question\_keyword INNER JOIN keyword ON question\_keyword\_idkeyword = keyword\_id

WHERE keyword\_loginadmin = 'cma'

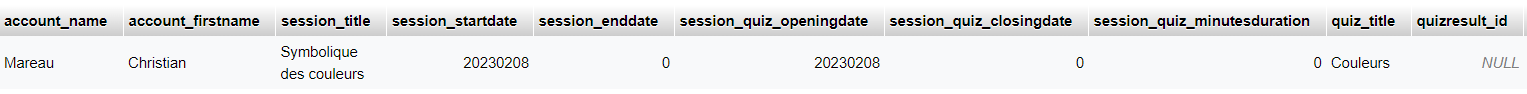
ORDER BY keyword\_word ASC

**class\_accountresult / getOne() :**

**SELECT quizresult\_sessiontitle,quizresult\_sessionsubtitle, quizresult\_quiztitle, quizresult\_quizsubtitle, quizresult\_firstnameadmin, quizresult\_loginadmin, quizresult\_nameadmin, quizresult\_startdate, quizresult\_enddate, quizresult\_maxduration, quizresult\_nbquestasked, quizresult\_maxnbquest, quizresult\_questaskedscore, quizresult\_maxquestaskedscore, quizresult\_maxscore, quizresult\_loginuser, quizresult\_idsession, quizresult\_idquiz FROM quizresult WHERE quizresult\_id = 7**

**class\_quiz\_userlist / public function \_\_construct($login) :**

SELECT account\_name, account\_firstname, session\_title, session\_startdate, session\_enddate, session\_quiz\_openingdate, session\_quiz\_closingdate, session\_quiz\_minutesduration, quiz\_title, quizresult\_id, quizresult\_startdate, quizresult\_maxscore FROM session\_user LEFT JOIN session ON session\_id = session\_user\_idsession LEFT JOIN account ON account\_login = session\_loginadmin LEFT JOIN session\_quiz ON session\_quiz\_idsession = session\_id LEFT JOIN quiz ON quiz\_id = session\_quiz\_idquiz LEFT JOIN quizresult ON quizresult\_idquiz = quiz\_id WHERE session\_user\_loginuser = 'chr' ORDER BY session\_startdate DESC, session\_quiz\_openingdate DESC, quizresult\_startdate DESC



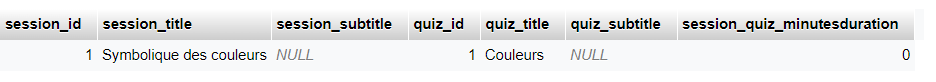
Une image contenant texte

Description générée automatiquement

**\_\_\_\_\_\_\_\_\_\_\_\_**

**class\_takenquiz / public function \_\_construct($sessionId, $quizId) //Getting general information about session and quiz :**

SELECT session\_id, session\_title, session\_subtitle, quiz\_id, quiz\_title, quiz\_subtitle, session\_quiz\_minutesduration FROM session LEFT JOIN session\_quiz ON session\_id = 1 AND session\_quiz\_idsession = session\_id AND session\_quiz\_idquiz = 1 LEFT JOIN quiz ON quiz\_id = session\_quiz\_idquiz WHERE quiz\_status = 'inline'



**class\_takenquiz / public function \_\_construct($sessionId, $quizId) //Getting questions and answers :**

SELECT question\_id, question\_question, question\_guideline, widget\_name, quiz\_question\_weight, quiz\_question\_numorder, answer\_idquestion, answer\_answer, answer\_ok FROM quiz\_question LEFT JOIN question ON quiz\_question\_idquiz = 1 AND question\_id = quiz\_question\_idquestion LEFT JOIN answer ON answer\_idquestion = question\_id LEFT JOIN widget ON widget\_id = question\_idwidget WHERE question\_status = 'inline' AND answer\_status = 'inline'

Une image contenant table

Description générée automatiquement

V1 ko

Array (

[0] => Array (

[0] => 1 [1] => De quelle couleur est le ciel ? [2] => [3] => radio buton [4] => 1 [5] => 0   
[6] => Array ( [0] => 1 [1] => Bleu [2] => 1 )   
[7] => Array ( [0] => 1 [1] => Blanc [2] => 0 )

)

[1] => Array (

[0] => 1 [1] => De quelle couleur est le ciel ? [2] => [3] => radio buton [4] => 1 [5] => 0  
[6] => Array ( [0] => 1 [1] => Rouge [2] => 0 )

)

[2] => Array (

[0] => 2 [1] => Quelle est la couleur de la mer ? [2] => [3] => radio buton [4] => 1 [5] => 0  
[6] => Array ( [0] => 2 [1] => Bleu [2] => 0 )

)

[3] => Array (

[0] => 2 [1] => Quelle est la couleur de la mer ? [2] => [3] => radio buton [4] => 1 [5] => 0  
[6] => Array ( [0] => 2 [1] => Jaune [2] => 0 )

)

[4] => Array (

[0] => 2 [1] => Quelle est la couleur de la mer ? [2] => [3] => radio buton [4] => 1 [5] => 0   
[6] => Array ( [0] => 2 [1] => Grise [2] => 1 )

)

[5] => Array (

[0] => 3 [1] => Quel est la couleur de tes yeux ? [2] => [3] => radio buton [4] => 1 [5] => 0   
[6] => Array ( [0] => 3 [1] => Rouge [2] => 0 )

)

[6] => Array (

[0] => 3 [1] => Quel est la couleur de tes yeux ? [2] => [3] => radio buton [4] => 1 [5] => 0  
[6] => Array ( [0] => 3 [1] => Noir [2] => 0 )

)

[7] => Array (

[0] => 3 [1] => Quel est la couleur de tes yeux ? [2] => [3] => radio buton [4] => 1 [5] => 0   
[6] => Array ( [0] => 3 [1] => Gris [2] => 1 )

)

)

**V2 ok**

Array (

[0] => Array ( [0] => **1** [1] => De quelle couleur est le ciel ? [2] => [3] => radio buton [4] => 1 [5] => 0

[6] => **1** [7] => Blanc [8] => 0 )

[1] => Array ( [0] => **1** [1] => De quelle couleur est le ciel ? [2] => [3] => radio buton [4] => 1 [5] => 0

[6] => **1** [7] => Rouge [8] => 0 )

[2] => Array ( [0] => **1** [1] => De quelle couleur est le ciel ? [2] => [3] => radio buton [4] => 1 [5] => 0

[6] => **1** [7] => Bleu [8] => 1 ) //ok

[3] => Array ( [0] => 2 [1] => Quelle est la couleur de la mer ? [2] => [3] => radio buton [4] => 1 [5] => 0 [6] => 2 [7] => Bleu [8] => 0 ) [4] => Array ( [0] => 2 [1] => Quelle est la couleur de la mer ? [2] => [3] => radio buton [4] => 1 [5] => 0 [6] => 2 [7] => Jaune [8] => 0 ) [5] => Array ( [0] => 2 [1] => Quelle est la couleur de la mer ? [2] => [3] => radio buton [4] => 1 [5] => 0 [6] => 2 [7] => Grise [8] => 1 ) [6] => Array ( [0] => 3 [1] => Quel est la couleur de tes yeux ? [2] => [3] => radio buton [4] => 1 [5] => 0 [6] => 3 [7] => Rouge [8] => 0 ) [7] => Array ( [0] => 3 [1] => Quel est la couleur de tes yeux ? [2] => [3] => radio buton [4] => 1 [5] => 0 [6] => 3 [7] => Noir [8] => 0 ) [8] => Array ( [0] => 3 [1] => Quel est la couleur de tes yeux ? [2] => [3] => radio buton [4] => 1 [5] => 0 [6] => 3 [7] => Gris [8] => 1 ) )

Une image contenant table

Description générée automatiquement

V3 : Avec un tri aléatoire suivi d’un groupe by questionId :

Cf résultat ci-dessus

SELECT

question\_id, question\_question, question\_guideline, widget\_name, quiz\_question\_weight, quiz\_question\_numorder,

answer\_idquestion, answer\_answer, answer\_ok

FROM quiz\_question

LEFT JOIN question ON quiz\_question\_idquiz = **1** AND question\_id = quiz\_question\_idquestion

LEFT JOIN answer ON answer\_idquestion = question\_id

LEFT JOIN widget ON widget\_id = question\_idwidget

WHERE question\_status = 'inline' AND answer\_status = 'inline'

ORDER BY question\_id, answer\_ok

Avec tri :

SELECT

question\_id, question\_question, question\_guideline, widget\_name, quiz\_question\_weight, quiz\_question\_numorder,

answer\_idquestion, answer\_answer, answer\_ok

FROM quiz\_question

LEFT JOIN question ON quiz\_question\_idquiz = **1** AND question\_id = quiz\_question\_idquestion

LEFT JOIN answer ON answer\_idquestion = question\_id

LEFT JOIN widget ON widget\_id = question\_idwidget

WHERE question\_status = 'inline' AND answer\_status = 'inline'

ORDER BY RAND()

Soit :

SELECT randomset.\* FROM (

SELECT question\_id, question\_question, question\_guideline, widget\_name, quiz\_question\_weight, quiz\_question\_numorder, answer\_idquestion, answer\_answer, answer\_ok FROM quiz\_question LEFT JOIN question ON quiz\_question\_idquiz = **1** AND question\_id = quiz\_question\_idquestion LEFT JOIN answer ON answer\_idquestion = question\_id LEFT JOIN widget ON widget\_id = question\_idwidget WHERE question\_status = 'inline' AND answer\_status = 'inline' ORDER BY RAND()

) AS randomset

// liste des questions :

SELECT question\_id FROM quiz\_question, question WHERE question\_status = 'inline' AND quiz\_question\_idquiz = **1** AND question\_id = quiz\_question\_idquestion ORDER BY RAND()

SELECT

distinct a.ID,

a.col2,

(SELECT

SUM(b.size)

FROM

tableb b

WHERE

b.id = a.col3)

FROM

tablea a

SELECT rq.question\_id, rq.question\_question, rq.question\_guideline,

ra.quiz\_question\_weight, ra.quiz\_question\_numorder, ra.answer\_idquestion, ra.answer\_answer, ra.answer\_ok, ra.widget\_name

FROM ( SELECT question\_id, question\_question, question\_guideline FROM quiz\_question, question WHERE question\_status = 'inline' AND quiz\_question\_idquiz = **1** AND question\_id = quiz\_question\_idquestion ORDER BY RAND() ) AS rq,

(SELECT question\_id, question\_question, question\_guideline, widget\_name, quiz\_question\_weight, quiz\_question\_numorder, answer\_idquestion, answer\_answer, answer\_ok FROM quiz\_question LEFT JOIN question ON quiz\_question\_idquiz = **1** AND question\_id = quiz\_question\_idquestion LEFT JOIN answer ON answer\_idquestion = question\_id LEFT JOIN widget ON widget\_id = question\_idwidget WHERE question\_status = 'inline' AND answer\_status = 'inline' ORDER BY RAND()) AS ra

WHERE ra.question\_id = rq.question\_id

Une image contenant table

Description générée automatiquement

1. <!DOCTYPE html PUBLIC "-*//W3C//DTD XHTML 1.0 Strict//EN"*
2. "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3. <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" >
4. <head>
5. <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
6. <title>Exemple</title>
7. <script type="text/javascript">
8. var pouet = false;
9. window.onbeforeunload = function (){
10. if ( pouet == false){
11. return false;
12. }
13. }
14. function goToUrl(url){
15. pouet = true;
16. document.location.href = url;
17. return false;
18. }
19. </script>
20. </head>
21. <body>
22. <a href="http://www.google.com">lien normal</a><br/>
23. <a href="" onclick="return goToUrl('http://www.google.com')">lien perso</a>
24. </body>
25. </html>

Une image contenant table

Description générée automatiquement

//Getting questions and answers all at random :

* ORDER BY RAND(), quiz\_question\_numorder : ko
* **ORDER BY quiz\_question\_numorder, RAND() : ok**
* ORDER BY quiz\_question\_numorder, question\_id, RAND() : ko : questions 0 tjrs ds le même ordre
* ORDER BY quiz\_question\_numorder, RAND(), question\_id : ko

A priori, ce qui est placé après RAND() n’est pas pris en compte

SELECT question\_id, question\_question, question\_guideline, widget\_id, quiz\_question\_weight, quiz\_question\_numorder, answer\_idquestion, answer\_answer, answer\_ok, answer\_id

FROM quiz\_question

LEFT JOIN question ON quiz\_question\_idquiz = 1 AND question\_id = quiz\_question\_idquestion

LEFT JOIN answer ON answer\_idquestion = question\_id

LEFT JOIN widget ON widget\_id = question\_idwidget

WHERE question\_status = 'inline' AND answer\_status = 'inline'

ORDER BY quiz\_question\_numorder, RAND(), question\_id

**class\_questions\_list / \_\_construct($login) :**

SELECT question\_id, question\_question, question\_status, question\_lastmodificationdate, question\_idwidget, question\_keyword\_idkeyword

FROM question LEFT JOIN question\_keyword ON question\_keyword\_idquestion = question\_id

WHERE question\_loginadmin = 'cma'

ORDER BY question\_id DESC

Une image contenant table

Description générée automatiquement

**class\_keywords\_list / \_\_construct($login) :**

SELECT y.keyword\_id, y.keyword\_word, COUNT(\*) FROM

(SELECT keyword\_id, keyword\_word, question\_keyword\_idkeyword

FROM keyword LEFT JOIN question\_keyword ON question\_keyword\_idkeyword = keyword\_id

WHERE keyword\_loginadmin = 'cma' AND question\_keyword\_idkeyword IS NOT NULL) AS y

GROUP BY y.keyword\_id

Une image contenant table

Description générée automatiquement

SELECT keyword\_id, keyword\_word, question\_keyword\_idkeyword

FROM keyword LEFT JOIN question\_keyword ON question\_keyword\_idkeyword = keyword\_id

WHERE keyword\_loginadmin = 'cma' AND question\_keyword\_idkeyword IS NULL

Une image contenant table

Description générée automatiquement

(

SELECT y.keyword\_id, y.keyword\_word, COUNT(\*) AS count FROM

(SELECT keyword\_id, keyword\_word, question\_keyword\_idkeyword

FROM keyword LEFT JOIN question\_keyword ON question\_keyword\_idkeyword = keyword\_id

WHERE keyword\_loginadmin = 'cma' AND question\_keyword\_idkeyword IS NOT NULL) AS y

GROUP BY y.keyword\_id

) UNION

(

SELECT keyword\_id, keyword\_word, question\_keyword\_idkeyword

FROM keyword LEFT JOIN question\_keyword ON question\_keyword\_idkeyword = keyword\_id

WHERE keyword\_loginadmin = 'cma' AND question\_keyword\_idkeyword IS NULL

)

**class\_keywords\_list / \_\_construct($login) / constructKeywordsQuestionsList($login) :**

SELECT **question\_keyword\_idkeyword**, question\_question

FROM question\_keyword

LEFT JOIN question ON question\_id = **question\_keyword\_idquestion**

LEFT JOIN keyword ON **keyword**\_id = **question\_keyword\_idkeyword**

WHERE keyword\_loginadmin = 'cma'

ORDER BY **question\_keyword\_idkeyword**, question\_lastmodificationdate DESC

Une image contenant texte

Description générée automatiquement

SELECT question\_question, question\_id**, question\_keyword\_idkeyword**

FROM question\_keyword

LEFT JOIN question ON question\_id = **question\_keyword\_idquestion**

LEFT JOIN keyword ON **keyword**\_id = **question\_keyword\_idkeyword**

WHERE keyword\_loginadmin = 'cma'

ORDER BY **question\_id**

Une image contenant table

Description générée automatiquement

question\_lastmodificationdate DESC

SELECT a.quiz\_id, a.quiz\_title, a.quiz\_subtitle, a.quiz\_status, a.quiz\_creationdate, a.quiz\_lastmodificationdate, a.ycountquestions, b.zcountsessions FROM

(

(SELECT quiz\_id, quiz\_title, quiz\_subtitle, quiz\_status, quiz\_creationdate, quiz\_lastmodificationdate, COUNT(\*) AS ycountquestions FROM

(SELECT quiz\_id, quiz\_title, quiz\_subtitle, quiz\_status, quiz\_creationdate, quiz\_lastmodificationdate FROM quiz\_question, quiz WHERE quiz\_loginadmin = 'cma' AND quiz\_id = quiz\_question\_idquiz) AS y

GROUP BY quiz\_id)

UNION

(SELECT quiz\_id, quiz\_title, quiz\_subtitle, quiz\_status, quiz\_creationdate, quiz\_lastmodificationdate, quiz\_question\_idquiz FROM quiz LEFT JOIN quiz\_question ON quiz\_id = quiz\_question\_idquiz WHERE quiz\_loginadmin = 'cma' AND quiz\_question\_idquiz IS NULL)

) AS a,

(

(SELECT quiz\_id, COUNT(\*) AS zcountsessions FROM

(SELECT quiz\_id FROM session\_quiz, quiz, session WHERE quiz\_loginadmin = 'cma' AND quiz\_id = session\_quiz\_idquiz

AND session\_id = session\_quiz\_idsession AND (session\_enddate IS NULL OR (UTC\_TIMESTAMP() < FROM\_UNIXTIME(session\_enddate)))

) AS z

GROUP BY quiz\_id)

UNION

(SELECT quiz\_id, session\_quiz\_idquiz FROM quiz LEFT JOIN session\_quiz ON quiz\_id = session\_quiz\_idquiz WHERE quiz\_loginadmin = 'cma' AND session\_quiz\_idquiz IS NULL)

) AS b

WHERE a.quiz\_id = b.quiz\_id

ORDER BY a.quiz\_id DESC

Time :

Une image contenant table

Description générée automatiquement

Time ++ :

Une image contenant table

Description générée automatiquement

Avec un quiz sans question mais avec une session :

Sans time :

Une image contenant table

Description générée automatiquement

SELECT a.quiz\_id, a.quiz\_title, a.quiz\_subtitle, a.quiz\_status, a.quiz\_creationdate, a.quiz\_lastmodificationdate, a.ycountquestions, b.zcountsessions FROM

(

(SELECT quiz\_id, quiz\_title, quiz\_subtitle, quiz\_status, quiz\_creationdate, quiz\_lastmodificationdate, COUNT(\*) AS ycountquestions FROM

(SELECT quiz\_id, quiz\_title, quiz\_subtitle, quiz\_status, quiz\_creationdate, quiz\_lastmodificationdate FROM quiz\_question, quiz WHERE quiz\_loginadmin = 'cma' AND quiz\_id = quiz\_question\_idquiz) AS y

GROUP BY quiz\_id)

UNION

(SELECT quiz\_id, quiz\_title, quiz\_subtitle, quiz\_status, quiz\_creationdate, quiz\_lastmodificationdate, quiz\_question\_idquiz FROM quiz LEFT JOIN quiz\_question ON quiz\_id = quiz\_question\_idquiz WHERE quiz\_loginadmin = 'cma' AND quiz\_question\_idquiz IS NULL)

) AS a,

(

(SELECT quiz\_id, COUNT(\*) AS zcountsessions FROM

(SELECT quiz\_id FROM session\_quiz, quiz WHERE quiz\_loginadmin = 'cma' AND quiz\_id = session\_quiz\_idquiz) AS z

GROUP BY quiz\_id)

UNION

(SELECT quiz\_id, session\_quiz\_idquiz FROM quiz LEFT JOIN session\_quiz ON quiz\_id = session\_quiz\_idquiz WHERE quiz\_loginadmin = 'cma' AND session\_quiz\_idquiz IS NULL)

) AS b

WHERE a.quiz\_id = b.quiz\_id

ORDER BY a.quiz\_id DESC

**Test maj quiz :**

**Départ :**

* Quiz ‘Couleurs’
  + Ciel (0, 1) >>>>>>> update 4, 3
  + yeux (0, 1) >>>>>>> RAS (0, 1)
  + mer (0, 1) >>>>>>> supp
* Soustitre: parentheses ôtées

**Opérations :**

1. + Q2 >>>>>>> créa 4, 3
2. X ciel
3. + ciel
4. Maj ciel : 4, 3
5. + Q3
6. X Q3
7. X mer
8. Maj Q2 : 4, 3

**Résultat 1 :**

Quiz :  
updatedquizdid (QUIZID) : 1   
updatedquiz (old quiz) : Couleurs  
up\_quiz : 1 (Soustitre: parentheses ôtées)  
quiz\_title (new quiz) : Couleurs  
quiz\_title\_old (old quiz) : Couleurs  
quiz\_subtitle : sous-titre bidon (Soustitre: parentheses ôtées)  
quiz\_subtitle\_old : (sous-titre bidon)  
quiz\_status : inline  
quiz\_status\_old : inline  
  
Questions :  
nb-questions\_original : 3  
  
Old questions to detete :  
  
deleteQuestion(1) : inline - 20230208 - De quelle couleur est le ciel ?  
deleteQuestion(2) : inline - 20230208 - Quelle est la couleur de la mer ?   
  
Old answers to update :  
  
  
New answers to create :  
nb-max-questions-new : 3  
  
  
createQuestion(draft - checkbox - Q2, 4, 3) **>>create = add => id needed**createQuestion(inline - radio - De quelle couleur est le ciel ?, 4, 3) **>>détecter le vieux ciel et le passer en ‘ ‘**

**Résultat 2 :**

Quiz :  
updatedquizdid (QUIZID) : 1  
updatedquiz (old quiz) : Couleurs  
up\_quiz : 0  
quiz\_title (new quiz) : Couleurs  
quiz\_title\_old (old quiz) : Couleurs  
quiz\_subtitle : (sous-titre bidon)  
quiz\_subtitle\_old : (sous-titre bidon)  
quiz\_status : inline  
quiz\_status\_old : inline  
  
Questions :  
nb-questions\_original : 3  
  
Old questions to detete :  
  
deleteQuestion(2) : inline - checkbox - Quelle est la couleur de la mer ?  
  
Old answers to update :  
  
updateQuestion(1, 4, 3) : inline - radio - De quelle couleur est le ciel ?  
  
New answers to create :  
nb-max-questions-new : 2  
  
  
createQuestion(25, 4, 3) : inline - radio - De quelle couleur est le ciel ?