Bloc Dashboard : Element de tableau de bord - Configuration des données de sortie

Ce panneau de paramètres détermine les choix des sorties en mode “table de données”. Certains paramètres, comme par exemple les paramètres de cache, peuvent affecter d'autres formes de sorties.

La fonction première de ce panneau est de choisir les colonnes de sorties dans le résultat de la requête (colonnes nommées) et d'alimenter un affichage en table de données. Les tables de données peuvent être de trois format :

- Pour des enregistements simples, fournissant des données “à plat” vous opterez pour la sortie “Linéaire”.
- Pour des enregistrements présentant une donnée à deux dimensions, vous pouvez choisir la “Table croisée”. La table croisée utilise une dimension simple comme colonnes de la table de rendu, mais peut utiliser un enseble de dimensions de sorties pour créer les lignes de la matrice.
- Pour des enregsitrements présentant une caractéristique arborescente reconnaissable* alors vous pouvez activer le rendu “hiérarchique”.

Paramètres

> Note : Cette documentation est un draft. Les requêtes proposées en exemple ne sont pas encore vérifiées

Dans toutes les entrées de type “liste”, le séparateur entre champ **DOIT ETRE** le point-virgule. Un certain nonmbre de paramètres secondaires admettent un nombre d'éléments de liste en fonction du nombre d'éléments d'un paramètre primaire. Vous trouverez ci-dessous les informations correspondantes.

Colonnes de sortie

Les colonnes de sortie permettent de choisir quelles colonnes de la requête seront utiisées pour l'affichage. Ce choix peut être différent suivant les différents modes de rendus. Certaines colonnes “techniques” n'ont pas à participer à l'affichage.

Les colonnes de sortie doivent être désignées par leur nom d'ALIAS. Par exemple, pour une requête :

```
SELECT
   id as id,
   shortname as sn,
   fullname as fn
FROM
   {course}
```

vous pourrez ne choisir “que” de ne sortir les colonnes sn et fn :

```
sn;fn
```

Cas particulier : Colonnes sommatives

Vous pouvez avoir besoin, par exemple pour piloter des courbes cumulées, d'extraire une forme “sommative” d'une colonne à résultat numérique. Par exmemple si vous voulez compter le nombre de création de nouveaux cours de manière cumulée par mois sur toute l'année.

Pour cela vous pouvez utiliser la syntaxe suivante, en lieu et place du simple nom aliasé de colonne :

Soit la requête :

```
SELECT
   DATE_FORMAT('%Y_%m', FROM_UNIXTIME(timecreated)) as month,
   COUNT(*) as cccount
FROM
   {courses}
GROUP BY
   month
```

Vous pouvez invoquer la somme cumulative du nombre de créations par mois comme ceci :

```
month;S(cccount)
```

Format des données de sortie

Chaque colonne de sortie fournit une donnée qui peut être formattée avant affichage. Il est possible de spécifier des chaines de formattage sous le format de la fonction sprintf(). Certaines syntaxes additionnelles ont été ajoutée pour des effets non prévus par ces syntaxes standard.

Paramètre primaire associé : Colonnes de sorties

Exemple :

Soit la requête :

```
SELECT
   DATE_FORMAT('%Y_%m', FROM_UNIXTIME(timecreated)) as month,
   SUM(1) / 12 as meancreation
FROM
   {courses}
```

Il est alors possible de formater de manière précise la donnée de sortie de la colonne meancreation pour ne lui autoriser par exemple qu'une décimale en sortie :

```
%s;%0.1f
```

(Le premier format concerne la colonne “month” à exprimer sous forme “texte”)

Quelques valeurs spéciales sont traitées en plus des formats sprintf() :

- Formats destinés aux sorties SQL sur fichier :
  - NUMERIC : donnée numérique dans une syntaxe d'INSERT
  - TEXT : donnée texte dans une syntaxe d'INSERT
- Formats spéciaux :
  - **%0 :** Ne pas afficher la donnée
  - **/%***regexp***%/ :** Permet l'extraction d'une partie de la donnée de sortie selon les deux règles suivantes :
    - si *regexp* contient un motif de sous-capture (motif parenthésé) alors la première sous-capture est retenue.
    - si *regexp* ne contient aucun motif de sous-capture, c'est la capture principale du motif qui est retenue.
  - **%hms :** Format converti en heures:minutes:secondes si la donnée d'origine est un nombre de secondes entier
  - **%hm :** Format converti en heures:minutes si la donnée d'origine est un nombre de secondes entier
  - **%D :** La date au format Moodle si la donnée est un timestamp unix

Amplification négative

en ajoutant un '-' (moins) devant le code de format, on déclenchera une mis een forme spéciale pour dissocier les valeurs négatives (texte rouge).

Insertion d'autres colonnes de sortie

Dans certains cas, par exemple, pour fabriquer des URLs dynamiques réutilisant certaines colonnes de sortie, il est nécessaire d'invoquer des résultats cachés ou provenant de ces colonnes. Une syntaxe spéciale permet de récupérer la valeur de sortie (de la même ligne de résultat) :

- **%{***colname***} :** où *colname* est l'un des alias de sortie de la requête initiale.

Nom des colonnes de sortie

Les colonnes de sortie peuvent etre “traduites” en libellés intelligibles pour l'affichage aux utilisateurs. Déifnissez une liste de labels utilisés sur les écrans. Vous donnerez autant de libellés que le paramètre primaire associé.

Exemple :

```
Mois;Nombre de cours créés
```

Paramètre primaire associé : Colonnes de sorties

Taille de pagination

Si la requête cible peut conduire à une sortie d'un très grand nombre de résultats, il peut être convenable, pour la performance des consultations de limiter le nombre de sorties. Si ce paramètre n'est pas défini, toutes les données seront affichées, sinon une pagination sera mise en place dans la

requête.

Précaution : Cette limitation peut affecter la cohérence de résultats affichés sous forme de table croisée ou sous forme hiérarchique. Utilisez là avec prudence pour ces formes d'affichage.

La pagination des résultats peut être imposée par certains réglages globaux du bloc Dashboard (Protection contre les résultats massifs).

Activation du cache de résultat

Le cache de résultat stocke en base de données les résultats bruts de la requête “finale” entièrement paramétrée avec ses composantes dynamiques. Les enregistrements produits par la requête sont préservés dans une table à part pour consultation rapide par la suite, pendant une certaine durée de vie. L'utilisation du cache est particulièrement intéressante sur des requêtes complexes avec beaucoup de jointures. Cela revien à stocker une “vue” en cache, sans intervention technique dans la base de données elle-même.

TTL du cache de résultat

C'est la durée (en minutes) pendant laquelle les données seront conservées suite à une requête d'alimentation du cache.

Nettoyer la table

La production de grandes tables avec beaucoup de données similaires peut rendre la lecture difficile lors de la consultation directe sur écran. La fonction de nettoyage supprime l'affchage d'une donnée si la valeur du résultat précédent est identique pour cette colonne.

Attention : Cette fonction n'est utilisable sans danger que pour des requêtes où toutes les valeurs sont exprimées. Elle ne permettrait pas de différentier par exemple une case “nettoyée” d'une ligne suivante qui ne contiendrait pas de données du tout.

Table triable

Si vous activez cette option, vous n'avez plus besoin de préciser une clause ORDER BY explicite pour ordonner vos résultats. Le bloc Dashboard vous proposera des commutateurs d'ordre de lecture en haut de chaque colonne des tables de sortie.

Colonne de séparation des sous totaux

si vous avez une colonne numérique (données aggrégables, ce qui exclut le cas de colonnes “ordinales” dont la valeur ne sert qu'à indexer ou ordonner les choses), vous désirerez parfois obtenir, en plus des totaliseurs globaux, des sous-totaux aggrégés suivant une dimension particulière de votre

table.

Vous devez entrer dans cette colonne le nom de la colonne pilotant l'aggrégation des sommes. Les sous-totaux affichés sont ceux des totaliseurs globaux qui ont été définis pour ce tableau de bord.

Contrainte : pour que les sous-totaux soient produits, il faut que les lignes du résultats soient évaluées dans un ordre qui permet l'aggrégation et l'affichage du sous-total à la fin de chaque groupe de valeur d'aggrégation. C'est pour cela que :

- La table doit être triable
- Le tri actif doit être celui de la colonne pilote de l'aggrégation des sous-totaux

Exemple :

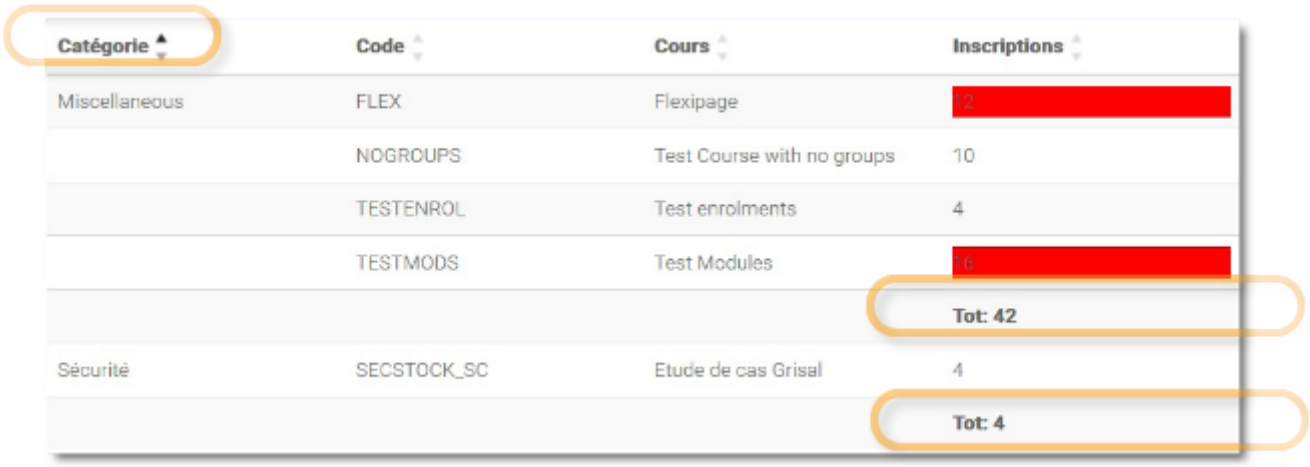
Soit la requête :

```
SELECT
  ue.id as ueid,
  c.shortname as sn,
  c.fullname as fn,
  cc.name as cat,
  COUNT(*) as enrols
FROM
  {course} c,
  {course_categories} cc,
  {enrol} e,
  {user_enrolments} ue
WHERE
  c.category = cc.id AND
  c.id = e.courseid AND
  e.id = ue.enrolid AND
  e.status = 0
<%%FILTERS%%>
<%%PARAMS%%>
GROUP BY
sn
```

qui compte le nombre d'inscriptions dans chaque cours, ce dernier qualifié par sa catégorie. Je souhaiterais pouvoir afficher les sous-totaux d'inscription dans chaque catégorie.

1. Commencer par ajouter un totaliseur sur les inscriptions (*enrol*)
2. Puis désigner la colonne *cat* comme pilote de sous-totaux
3. Rendez la table triable
4. Affichez le résultat et triez sur la colonne “Catégories” (*cat*)

les sous-totaux vont être affichés



Catégorie	Code	Cours	Inscriptions
Miscellaneous	FLEX	Flexipage	12
	NOGROUPS	Test Course with no groups	10
	TESTENROL	Test enrolments	4
	TESTMODS	Test Modules	16
			Tot: 42
Sécurité	SECSTOCK_SC	Etude de cas Grisal	4
			Tot: 4

[Revenir à l'index du guide utilisateur]

From:
https://docsen.activeprolearn.com/ - **Documentation Moodle ActiveProLearn**

Permanent link:
https://docsen.activeprolearn.com/doku.php?id=blockdashboardconfigoutputfields&rev=1448129071

Last update: **2026/01/13 08:58**