

Logiciel

tablecalc : faire des calculs dans une table comme avec Excel

Parfois, on peut avoir besoin de faire des calculs simples dans la table DokuWiki.

À l'aide de ce plugin, vous pourrez insérer des formules de style Excel dans le tableau

Pré-requis

Installation

Recherchez et installez le plugin **tablecalc** à l'aide du gestionnaire d'extensions.

<http://narezka.org/cfd/msgdb/740/tablecalc.zip>

Configuration

Utilisation

Syntaxe

Pour effectuer un calcul, insérez une formule de style XL dans la feuille de calcul. Toute expression doit commencer par `~~=` et se terminer par `~~`.

Vous pouvez spécifier

- directement une plage (comme le fait XL)
- ou utiliser une fonction **range()**

La plage consiste en une référence à la cellule de départ et à la cellule d'arrivée, comme ceci :

r0c4



Notez que les index de ligne (r) et de colonne (c) commencent à partir de zéro (0)

	0	1	2
--	---	---	---

	0	1	2
0	r0c0	r0c1	r0c2
1	r1c0	r1c1	r1c2
2	r2c0	r2c1	r2c2

Les préfixes de ligne et de colonne peuvent être permutés. Ceci est équivalent à ce qui précède :

```
c4r0
```

Vous pouvez également faire référence à plusieurs cellules dans une plage :

```
| r0c0 : r1c1 | |
```

De plus, vous pouvez utiliser plusieurs plages :

```
r0c0 : r1c1 , r0c3 : r1c4
```

Il est recommandé de ne pas utiliser de références pour des cellules inexistantes. Par exemple, ce n'est pas correct (bien que cela fonctionnera, renvoyant « 3 ») :



```
| 1 |  
| 2 |  
| ~~=sum( r0c0 : r99c99 )~~ |
```

Utilisez plutôt des constructions comme celle-ci :

```
| 1 |  
| 2 |  
| ~~=sum( range( 0 , 0 , col() , row() - 1 ) )~~ |
```

Fonctions

The following functions are implemented:

Func	Description
average(range)	Renvoie la moyenne de la plage spécifiée
cell(column,row)	Renvoie la valeur numérique de la cellule (colonne, ligne)
check(condition>true>false)	Exécute l'instruction true, si la condition n'est pas zéro
col()	Retourne la colonne courante
compare(a;b;operation)	Comparaison mathématique entre a et b. Renvoie zéro lorsque les conditions de l'opération ne sont pas remplies
count(range)	Renvoie le nombre d'éléments dans la plage spécifiée

Func	Description
countif(range;b;operation)	Compter les valeurs dans la plage qui correspondent à la comparaison avec b. Le type de comparaison est spécifié en opération
label(string)	Lie l'étiquette à la table
max(range)	Renvoie la valeur maximale dans la plage spécifiée
min(range)	Renvoie la valeur minimale dans la plage spécifiée
range(column1,row1,column2,row2)	Renvoie la plage interne pour d'autres fonctions
round(number;decimals)	Renvoie le nombre, arrondi aux décimales spécifiées
row()	Retourne la ligne courante
sum(range)	Renvoie la somme de la plage spécifiée
calc()	Obsolète, désormais inutile

Bien que vous puissiez utiliser deux points comme délimiteur dans les fonctions, le point-virgule est préférable et recommandé.

Opérateurs

La plupart des opérateurs arithmétiques Javascript standard sont pris en charge, mais certains ¹⁾ sont en conflit avec le balisage de la table, les opérateurs suivants sont donc disponibles :

Opérateur	Description
+	Addition et plus unaire
-	Soustraction et unaire négatif
*	Multiplication
/	Division
%	Modulo (reste de division)
&	ET logique
«	Décaler à gauche
»	Décaler à droite

Exemples

1er exemple

```
| 1 | 2 | ~=r0c0+r0c1~~ | ~=10.2+1.5~~ |
```

1	2
---	---

2e exemple

```
| 1 | 2 |  
| 3 | 4 |  
| ~=sum(r0c0:r1c1)~~ ||
```

1	2
---	---

3	4

3e exemple

	1		2		3		4	
	5		6		7		8	
	~~=sum(r0c0:r1c1,r0c3:r1c4)~~							

1	2	3	4
5	6	7	8

4e exemple

	1	
	2	
	3	
	4	
	5.74	
	6	
	7	
	8	
	9	
	10	
	11	
	~~=sum(range(col(),0,col(),row()-1))~~	

1
2
3
4
5.74
6
7
8
9
10
11

V

	1	
	2	
	3	
	4	
	5	
	6	

```
|7|
|8|
|9|
|11|
|~~=average(range(col(),0,col(),row()-1))~~|
```

1
2
3
4
5
6
7
8
9
11

6e exemple

```
| ~~=label(ex6_1)~~1 | 2 |
| 3 | 4 |

Sum: **~~=sum(ex6_1.c0r0:c99r99)~~**
```

1	2
3	4

Sum:

7e exemple

```
| **~~=label(ex7_1)~~11** | ~~=sum(ex7_2.c0r0:c99r99)~~ |
| 13 | 14 |

| ~~=label(ex7_2)~~1 | 2 |
| 3 | 4 |

Somme : **~~=sum(ex7_1.c0r0:c1r1)~~**
```

11	
13	14
1	2
3	4

Somme :

8e exemple

```
| **~~=min(c0r1:c0r3)~~** | **~~=max(c1r1:c1r3)~~** |
| 1 | 7 |
| 2 | 8 |
| 3 | 9 |
```

1	7
2	8
3	9

9e exemple

```
| 1 | ~~=check(cell(0,row()),#True,#False)~~ |
| 0 | ~~=check(cell(0,row()),#True,#False)~~ |
| x | ~~=check(cell(0,row()),#True,#False)~~ |
|   | ~~=check(cell(0,row()),#True,#False)~~ |
| **** | ~~=check(cell(0,row()),#True,#False)~~ |
```

1
0
x

10e exemple

```
| 1 | 2 | 1=2 |
~~=check(compare(cell(0,row()),cell(1,row()),#=#),#True,#False)~~ |
| 3 | 3 | 3=3 |
~~=check(compare(cell(0,row()),cell(1,row()),#=#),#True,#False)~~ |
| 4 | 5 | 4<5 |
~~=check(compare(cell(0,row()),cell(1,row()),#<),#True,#False)~~ |
| 6 | 7 | 6>7 |
~~=check(compare(cell(0,row()),cell(1,row()),#>),#True,#False)~~ |
| 8 | 9 | 8>9 |
~~=check(compare(cell(0,row()),cell(1,row()),#>),#True,#False)~~ |
| 10 | 10 | 10≥10 |
~~=check(compare(cell(0,row()),cell(1,row()),#>=),#True,#False)~~ |
| 11 | 11 | 11≤11 |
~~=check(compare(cell(0,row()),cell(1,row()),#>=),#True,#False)~~ |
| 12 | 12 | 12≠12 |
~~=check(compare(cell(0,row()),cell(1,row()),#!=),#True,#False)~~ |
```

1	2	1=2
3	3	3=3
4	5	4<5
6	7	6>7

8	9	8>9
10	10	10≥10
11	11	11≤11
12	12	12≠12

11e exemple

Operator	Equation	Result
+	~~= 3 + 2 ~~	5
-	~~= 3 - 2 ~~	1
*	~~= 3 * 2 ~~	6
/	~~= 6 / 2 ~~	3
%	~~= 7 % 2 ~~	1
unary +	~~= +2 ~~	2
unary -	~~= -2 ~~	-2
&	~~= 3 & 2 ~~	2
<<	~~= 2 << 7 ~~	256
>>	~~= 8 >> 2 ~~	2

12e exemple

```
|~~=r0c1+5~~|6|
```

6

13e exemple

```
| ~~=label(ex13_1)~~pommes | 32 |
| oranges | 54 |
| pêches | 75 |
| pommes | 86 |
```

Pommes: ~~=countif(ex13_1.r0c0:r9c0,#pommes,#=)~~

pommes	32
oranges	54
pêches	75
pommes	86

Pommes:

Désinstallation

Voir aussi

- **(en)** <https://www.dokuwiki.org/plugin:tablecalc>

Basé sur « [Article](#) » par dokuwiki.org.

¹⁾

^ | ~

From:
<https://nfrappe.fr/doc-0/> - **Documentation du Dr Nicolas Frappé**

Permanent link:
<https://nfrappe.fr/doc-0/doku.php?id=logiciel:internet:dokuwiki:plugins:tablecalc:start>

Last update: **2022/10/01 23:26**

