

Beste de savoir

Rédiger sur ZdS

23 avril 2020

Table des matières

1.	Mise en forme de texte classique	2
1.1.	Paragraphe	2
1.2.	Titres	2
1.3.	Emphases	3
1.4.	Barrer	3
1.5.	Alignement du texte	4
1.6.	Indices et exposants	4
2.	Listes	5
3.	Liens et emails	5
3.1.	Liens et emails avec texte d’ancrage	5
3.2.	Liens présentés sous forme d’URL ou d’email	6
4.	Tableaux et lignes	6
4.1.	Des tableaux simples	6
4.2.	Tableaux complexes	6
4.3.	Légendes de tableaux	7
4.4.	Lignes horizontales	7
5.	Formules mathématiques	8
6.	Code	8
6.1.	Bloc de code	8
6.2.	Mise en évidence de lignes de code	9
6.3.	Début de la numérotation	9
6.4.	Code inline	9
7.	Médias	10
7.1.	Images	10
7.2.	Vidéos	10
7.3.	Touches	11
7.4.	Smileys	11
8.	Blocs spéciaux	12
8.1.	Balises attention, erreur, information, question et secret	12
8.2.	Citations	13
8.3.	Blocs à titre et blocs neutres	13
9.	Mentionner des membres	13
10.	Notes et abréviations	14
10.1.	Abréviations	14
10.2.	Notes de bas de page	14
11.	Caractères réservés et commentaires	14
12.	La typographie	14

Zeste de Savoir utilise un langage de formatage de texte très proche du Markdown, auquel ont été ajoutées quelques extensions courantes. Ce langage est utilisé sur [ZdS](#) pour écrire les tutoriels, les articles, les messages de forums ou MP, etc.

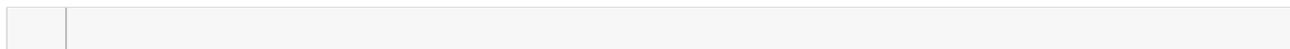
1. Mise en forme de texte classique

On ne présente plus le Markdown, ce langage de formatage alliant la simplicité d'écriture à la facilité de lecture. Sa syntaxe très légère permet en effet de lire du simple texte de façon presque aussi agréable que s'il était réellement mis en forme. Mais aussi pratique que soit le Markdown, il ne permet pas de tout baliser. Les indices et exposants, par exemple, n'ont pas de représentations Markdown. Il existe donc des extensions permettant de compléter le Markdown natif. Certaines de ces extensions, très courantes, ont été reprises sur [ZdS](#), d'autres ont été spécifiquement implémentées pour répondre aux besoins précis de [ZdS](#). Nous vous présentons donc ici les syntaxes utilisables sur [ZdS](#).

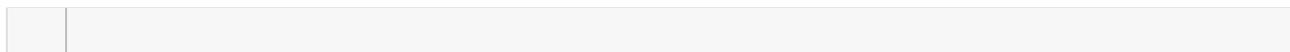
1. Mise en forme de texte classique

1.1. Paragraphes

Les paragraphes s'écrivent naturellement, en sautant une ligne :



Tout comme avec le Markdown standard, on change de paragraphe en **sautant une ligne**. Un simple retour à la ligne ne suffit donc pas et sera interprété comme un simple espace :

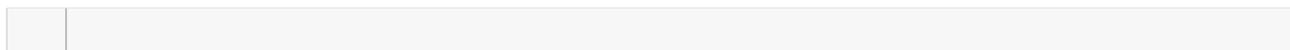


Si vraiment vous tenez à revenir à la ligne sans changer de paragraphe, comme ceci par exemple, alors il suffit de terminer la première ligne par deux espaces.

De plus, le Markdown standard autorise l'insertion de HTML, mais pour des raisons de sécurité nous avons choisi de ne pas laisser cette possibilité. Si vous écrivez du HTML, celui-ci apparaîtra donc tel quel dans votre texte.

1.2. Titres

Les titres sont précédés d'un ou plusieurs croisillons suivant le niveau hiérarchique voulu. Ainsi, un titre de premier niveau s'écrit avec un seul croisillon, un titre de deuxième niveau avec deux croisillons, etc.



Quatre niveaux hiérarchiques sont disponibles. J'attire d'ailleurs votre attention sur ce point car il est très important de donner la bonne hiérarchie à vos titres lorsque vous rédigerez vos tutoriels.

1. Mise en forme de texte classique

1.3. Emphases

Les emphases permettent de mettre un morceau de votre texte en valeur. Deux types d'emphases sont disponibles : l'italique et le gras.

Pour mettre du texte en *italique*, utilisez l'astérisque ou le souligné (*underscore*) :

ou



Si la syntaxe avec *underscore* est utilisée en milieu de mot, alors le texte ne sera pas transformé. Ainsi, `truc_bidule_mush` ne sera pas transformé alors que `truc*bidule*mush` le sera. Cela tient du fait que les expressions avec des *underscores* sont communes en informatique comme `Mon_super_nom_de_fichier.txt` par exemple.

Pour mettre du texte en **gras**, le principe est le même, en doublant le symbole :

ou

Par souci de simplicité et de lisibilité, vous ne pourrez pas mettre du texte en couleur, le souligner, changer sa taille ou bien encore en changer la police.

1.4. Barrer

Barrer du texte (~~comme ceci~~) se fait en utilisant deux tildes successifs avant et après la portion de texte concernée :

Pour information, il s'agit de la syntaxe utilisée par Pandoc.

1. Mise en forme de texte classique

1.5. Alignement du texte

Par défaut, le texte est bien évidemment aligné à gauche. Comme nous le verrons plus loin, certains éléments sont centrés automatiquement, comme les images seules dans leur paragraphe par exemple. Vous n'avez donc en général pas à vous soucier de l'alignement du texte : le site s'en charge pour vous.

Dans les rares cas où vous souhaiteriez centrer volontairement un texte (si l'envie vous prenait d'écrire un poème par exemple), vous pourriez néanmoins utiliser la syntaxe ci-dessous :

```
    
```

Le texte est simplement entouré de deux petites flèches (tiret et chevron) de directions inversées. Pour aligner à droite, on utilise deux flèches dirigées vers la droite :

```
    
```

Il est impossible d'imbriquer des alignements. Cela n'aurait de toute façon pas de sens (comment aligner à droite un texte centré?).

Encore une fois, l'alignement est géré automatiquement dans la majorité des cas. N'en abusez pas, cela pourrait gêner la lecture.

Enfin, sachez qu'il est impossible de justifier du texte sur le site.

1.6. Indices et exposants

Là encore, ce sont les syntaxes de Pandoc qui sont utilisées pour mettre en indice ou en exposant une portion de texte.

On utilise l'accent circonflexe pour l'exposant. Si par exemple on veut écrire que 2^{10} vaut 1024, alors on écrira :

```
    
```

Pour l'indice, comme dans H_2O par exemple, on utilise cette fois le tilde :

```
    
```

2. Listes

Vous pouvez utiliser deux types de liste :

- les listes non ordonnées (comme la présente) ;
- les listes ordonnées par chiffres arabes.

C'est peut-être la syntaxe la plus intuitive du Markdown ! Il suffit de matérialiser les puces par des tirets :

```
— Liste non ordonnée
```

Ou bien par des chiffres :

```
1. Liste ordonnée
```

Prenez simplement garde à bien sauter une ligne **avant et après** vos listes.

Pour faire une sous-liste, indentez les puces imbriquées avec **quatre** espaces :

```
— Liste ordonnée
  — Liste ordonnée imbriquée
```

On obtient ainsi :

- Ma très belle ;
- liste ;
 - avec une sous-liste ;
- à puces.

3. Liens et emails

Il existe deux façons d'écrire des liens : avec ou sans texte d'ancrage.

3.1. Liens et emails avec texte d'ancrage

Pour faire un [lien](#) sur un morceau de texte (qu'on appelle donc texte d'ancrage, ici le mot "lien"), on utilise la syntaxe suivante :

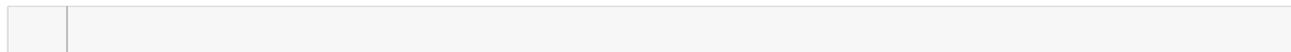
```
[lien] (url)
```

Attention à ne pas mettre d'espace entre la partie concernant le texte d'ancrage (entre crochets) et la partie concernant l'URL (entre parenthèses).

Le titre du lien (ici "Zeste de Savoir" entre guillemets) est optionnel. S'il est renseigné, il apparaît sur le lien au passage de la souris.

4. Tableaux et lignes

Les emails peuvent s'écrire de la même façon que les liens. Pensez simplement à ajouter la mention "mailto:" :



3.2. Liens présentés sous forme d'URL ou d'email

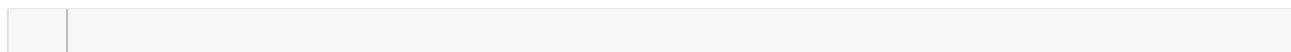
Si vous ne souhaitez pas utiliser de texte d'ancrage et ainsi rendre une URL ou un email cliquable, alors vous n'avez rien à faire : URL et emails seront automatiquement cliquables.

Pour les emails, vous n'avez donc même pas besoin de vous soucier du "mailto".

4. Tableaux et lignes

4.1. Des tableaux simples

Pour faire un tableau, la façon la plus simple est encore de le dessiner, à l'aide de barres verticales et de tirets :



L'exemple ci-dessus donnera donc :

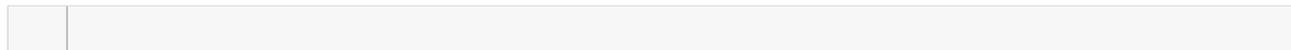
Nom	Age
Fred	39
Sam	38
Alice	35
Mathilde	35

Cette syntaxe est simple mais elle a ses limites : il est impossible de revenir à la ligne dans une cellule ou bien de fusionner des lignes ou des colonnes. Si vous avez vraiment besoin de faire cela, il vous faudra utiliser une autre syntaxe de tableau, plus lourde mais plus complète, comme nous allons le voir à présent.

4.2. Tableaux complexes

Pour des tableaux plus complexes, dans lesquels vous pourrez notamment revenir à la ligne dans une cellule et fusionner des lignes ou colonnes, il vous faut utiliser la syntaxe dite « grid-table » :

4. Tableaux et lignes



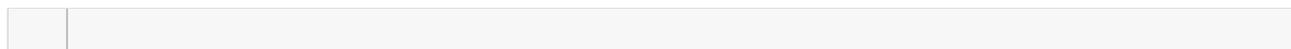
Ce qui vous donnera :

Table Headings		Here
Sub	Headings	Too
spans rows	column spanning	
	normal	cell
multi line cells too	cells can be <i>formatted</i> paragraphs	

4.3. Légendes de tableaux

Quelle que soit la syntaxe utilisée, vous pouvez indiquer une légende à votre tableau en ajoutant une ligne **Table: Ma légende** juste en dessous du tableau.

Le mot « Table » est optionnel, pas les deux-points. Il peut y avoir un espace entre « Table » et les deux-points.

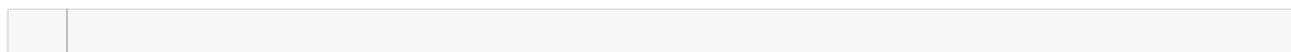


Nom	Age
Fred	39
Sam	38
Alice	35
Mathilde	35

TABLE 4.4. – Tableau des âges

4.4. Lignes horizontales

Pour tracer une ligne horizontale, le principe est le même : *dessinez-là*. La syntaxe est cette fois bien plus simple puisqu'elle n'est constituée que de trois tirets (ou plus, ça ne change rien au résultat) :



Voici le résultat :

5. Formules mathématiques

Une formule mathématique s'écrit à l'aide d'expressions TeX Math, en l'entourant de deux caractères dollars `$$` :

Ce qui donne :

$$a \cdot x^2 + b \cdot x + c = 0 \implies x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

En faisant ainsi, votre formule est en mode *displayed* : elle est dans son propre paragraphe et s'affiche en prenant ses aises. Ainsi les chiffres et autres symboles sont écrits en grande taille et sont facilement lisibles.

Si vous voulez écrire votre formule au sein même d'un paragraphe (comme ceci : $a \cdot x^2 + b \cdot x + c = 0$), alors n'utilisez cette fois qu'un seul caractère dollar `$` avant et après votre formule. Par exemple :

Comme pour les tableaux, vous pouvez mettre une légende à votre formule, en ajoutant une ligne **Equation: Ma légende** juste en dessous.

Si vous souhaitez écrire deux symboles `$` dans une même ligne, alors il vous faut **échapper** au moins l'un des deux (c'est-à-dire les faire précéder d'un antislash : `\$`) afin que le texte ne soit pas considéré comme une formule mathématique.



Pour en savoir plus sur l'utilisation des expressions mathématiques, vous pouvez consulter le tutoriel dédié : [Comment rédiger des maths sur Zeste de Savoir?](#) ↗

6. Code

6.1. Bloc de code

Il n'est pas rare d'illustrer son propos d'un petit exemple de code :

Listing 1 – Mon exemple de code

Pour cela, il existe plusieurs solutions.

Première solution : entourer votre code d'au moins trois accents graves ““ (`(Alt Gr) + 7`), avant et après :

Le langage utilisé sera détecté automatiquement et donc coloré en conséquence. Si tel n'est pas le cas, vous pouvez forcer le langage en l'indiquant à la suite des caractères ouvrants :

6. Code

La liste des langages supportés est celle de **pygment**, vous la trouverez [ici](#) . Les mots-clés à insérer pour déclencher la coloration sont les « short names » disponibles sur [cette page](#) . Seconde solution, faites précéder chaque ligne de quatre espaces ou bien d'une tabulation :

Là encore, vous pouvez mettre une légende à votre bloc de code en ajoutant, juste en dessous du bloc, une ligne `Code: Votre légende`.

6.2. Mise en évidence de lignes de code

Mettre en évidence une portion de code permet d'appuyer votre explication. Après le nom du langage, indiquez simplement les lignes à surligner avec la mention `hl_lines`. Vous pouvez surligner les lignes unes à unes ou par groupes. Le syntaxe est la suivante :

i

À venir

Actuellement, la mise en évidence des lignes de code ne fonctionne que dans les PDF. Cette fonctionnalité devrait arriver plus tard sur le site web.

6.3. Début de la numérotation

Il est possible de spécifier le début de numérotation. Par exemple :

On utilise le mot-clé `linenostart` de la même façon que le `hl_lines` vu précédemment.

i

À venir

Actuellement, on ne peut changer le début de numérotation des lignes de code que dans les PDF. Cette fonctionnalité devrait arriver plus tard sur le site web.

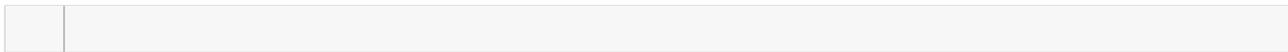
6.4. Code inline

Enfin, si vous souhaitez insérer un petit élément de code dans votre phrase (comme `print` par exemple), alors un seul accent grave autour du mot suffira :

7. Médias

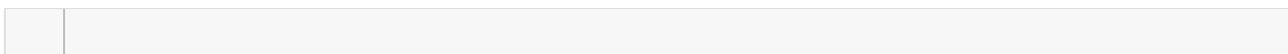
7.1. Images

L'insertion d'une image ressemble à celle d'un lien, à ceci près que le texte d'ancrage doit être remplacé par un texte alternatif :



Il y a une petite différence de comportement selon que vous placiez votre image seule dans un paragraphe (ou dans un bloc de texte type citation, information, secret, etc.) ou bien au cours d'une phrase dans votre texte.

Lorsque l'image est seule, alors elle est présentée comme figure **avec légende**. Ainsi, si on prend l'exemple suivant :



Alors le rendu sera la suivant :

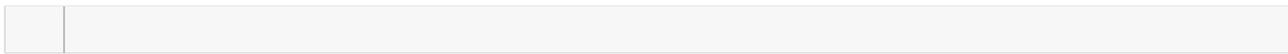
Bla bla bla



FIGURE 7.1. – Logo Creative Commons

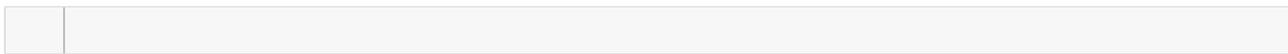
Bla bla bla encore

Si en revanche l'image est placée au cours d'un texte, alors le comportement sera plus classique et l'image apparaîtra naturellement dans la phrase :



Dans tous les cas, le texte alternatif **doit** être renseigné. Il sert à apporter la même information que l'image si celle-ci ne peut être chargée ou bien ne peut être vue (notamment pour les synthétiseurs vocaux pour les non-voyants).

Il est possible de définir à la fois un texte alternatif et une légende, en utilisant le mot-clé **Figure** de la même façon que pour les légendes de tableaux ou blocs de code :



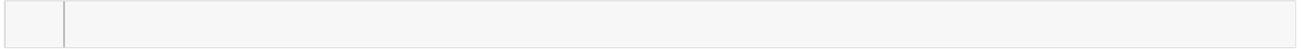
Ainsi, le texte alternatif et la légende sont bien différents.

7.2. Vidéos

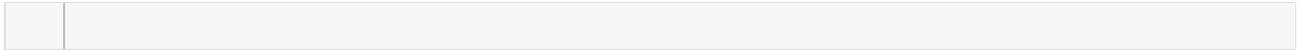
Les vidéos doivent être insérées avec une syntaxe dédié : `!(URL Vidéo)`. Elles ne peuvent être inline (au sein d'une phrase).

Pour insérer une vidéo on peut donc faire :

7. Médias



ou avec une légende :



Par exemple :

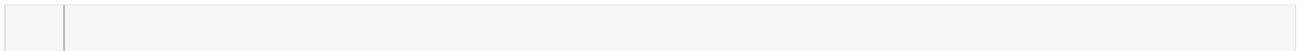
ÉLÉMENT EXTERNE (VIDEO) —

Consultez cet élément à l'adresse <https://www.youtube.com/embed/oavMtUWDBTM?feature=oembed>.

Ma super légende

7.3. Touches

Pour représenter une touche, utilisez deux barres verticales avant et après le nom de la touche :



Vous pouvez bien sûr mettre `ce que vous voulez` comme nom de touche.

7.4. Smileys

Que serait un forum sans smileys ? Un forum plus agréable ? Peut-être. Il n'empêche que les fameux smileys sont incontournables. Sur [ZdS](#), les smileys que vous écrivez seront automatiquement transformés en image. Ci-dessous une liste (non exhaustive) des smileys disponibles :

Entrée	Résultat
:)	
:D	
;)	
:p	
:lol:	
:euh:	
:(
:o	
:colere2:	
o_0	

^^	
:-°	
:ange:	
:colere:	
:diable:	
:magicien:	
:ninja:	
>_<	
:pirate:	
:'(
:honte:	
:soleil:	
:waw:	
:zorro:	

TABLE 7.6. – Liste non exhaustive des smileys

N’oubliez pas : l’abus de smileys est dangereux pour votre santé et celle de vos proches, utilisez-les avec modération.

8. Blocs spéciaux

8.1. Balises attention, erreur, information, question et secret

Les tutoriels et articles de [ZdS](#) sont parsemés de balises telles que la balise ”information” :

i

Ceci est une balise d’information.
Cool, non ?

Elle se fait avec la syntaxe suivante :

```
<div data-bbox="93 754 908 776" style="background-color: #f0f0f0; border: 1px solid #ccc; padding: 5px; min-height: 20px;">
```

Ou dans sa version raccourcie :

```
<div data-bbox="93 813 908 836" style="background-color: #f0f0f0; border: 1px solid #ccc; padding: 5px; min-height: 20px;">
```

Les balises disponibles sont :

- attention
- erreur
- information

9. Mentionner des membres

- question
- secret

La balise "secret" (appelée "spoiler" sur certains sites) a ceci de spécial qu'elle masque son contenu par défaut et ne le rend visible qu'au clic de l'utilisateur.

8.2. Citations

Les citations permettent de séparer votre propos de celui que vous rapportez. D'ailleurs, si l'on en croit ce vieux proverbe nous venant d'une petite planète quelque part aux confins de Bételgeuse, il ne faut pas s'en priver :

Les citations, c'est bien.

Petite planète quelque part aux confins de Bételgeuse

On utilise pour cela un chevron devant chaque début de ligne, avec optionnellement votre source, écrite de la même façon que les légendes (avec le mot-clé `Source`) :

8.3. Blocs à titre et blocs neutres

Parfois vous désirez mettre en avant une information (par exemple un théorème) mais aucun des blocs spéciaux ne vous convient. Vous pouvez alors utiliser un bloc *neutre*.

Ce bloc aura forcément un titre, ce qui aidera à mettre le contenu en avant. Par exemple :

Théorème de Pythagore

Un triangle ABC est rectangle en a si et seulement si $AB^2 + AC^2 = BC^2$

Notons que tous les autres blocs peuvent aussi avoir un titre, il se présente de la même manière, c'est-à-dire en mettant `[[nom_du_bloc | titre du bloc]]`. Ils ne sont par contre pas obligatoires.

9. Mentionner des membres

Il est possible de mentionner des membres dans un message avec `@pseudo` ou `@**pseudo-compose**`. Ces membres recevront une notification, il ne faut donc pas en abuser !

Pour éviter les abus, seulement les 15 premiers membres mentionnés dans le message reçoivent une notification.

10. Notes et abréviations

10.1. Abréviations

Il est souvent utile de préciser la signification d'une abréviation (notamment d'un acronyme ou d'un sigle), sans toutefois la faire figurer dans le corps du texte. En utilisant la syntaxe suivante, la signification apparaîtra au passage de la souris sur l'abréviation :

```
 
```

Bienvenue, donc, sur **ZdS**!

10.2. Notes de bas de page

Toujours dans l'idée d'enrichir votre texte¹, vous pouvez utiliser des notes de base de page :

```
 
```

Les notes sont alors numérotées automatiquement. Vous n'avez à vous soucier que du nom que vous donner à votre note.

11. Caractères réservés et commentaires

Si vous avez besoin d'écrire un caractère réservé entrant en conflit avec l'une des syntaxes décrites ici (l'astérisque par exemple), vous pouvez le faire en le précédant d'un antislash : `*`. Enfin, vous pouvez mettre une partie de votre texte en commentaires en le mettant entre des balises spécifiques :

```
 
```

La partie commentée n'apparaîtra tout simplement pas dans le rendu final.

12. La typographie

Afin de vous aider à rédiger un contenu agréable à lire, le moteur markdown de zeste de savoir inclus un module de typographie automatique.

Cela signifie que certains caractères (par exemples les apostrophes anglais) seront remplacés par des caractères typographiquement corrects.

De même, nous nous occupons de mettre les espaces insécables aux bons endroits.



Espace quoi ?

Lorsque vous écrivez une phrase longue, votre navigateur va automatiquement passer à la ligne une fois la limite atteinte. Pour faire cela, il va "couper" la phrase. Le meilleur endroit pour couper une phrase, c'est l'espace entre deux mots.

Mais voilà, il existe certaines règles de typographie qui rendent ce comportement gênant. Par exemple, lorsque vous vous préparez à faire une liste et que vous terminez votre phrase par `:`.

1. Sans pour autant l'alourdir.

12. La typographie

Logiquement, vous devez entourer votre caractère `:` d'espaces. Ce qui peut amener un cas étrange : votre phrase étant trop longue, votre navigateur met les `:` à la ligne.

C'est ce problème que résout l'espace insécable. Son nom elle ne l'usurpe pas : c'est une espace qui ne peut pas être coupée. En fait l'espace qui est avant les `:` doit être une espace insécable.

Le problème c'est que la majorité des claviers et des systèmes d'exploitation n'ont pas de moyen simple de faire cette espace. Et c'est pourquoi nous la faisons pour vous. En plus ça vous évitera de vous demander quels types et combien d'espaces vous devez mettre autour d'un signe de ponctuation.

12.0.1. Aide mémoire :

Caractère	Remplacement
<code>%O</code>	
<code><</code>	<code>«</code>
<code>></code>	<code>»</code>
<code>--mot--</code>	<code>mot</code>
<code>---mot---</code>	<code>mot</code>
<code>...</code>	
<code>.:;!?()</code>	les espaces sécables ou non sont correctement placées autour des signes

TABLE 12.8. – Tableau des remplacements

N'hésitez pas à commenter ce tuto pour remonter toute erreur ou imprécision. Merci !

Liste des abréviations

zds Zeste de Savoir. 1, 2, 11, 12, 14