

 eBook Gratuit

APPRENEZ Markdown

eBook gratuit non affilié créé à partir des
contributeurs de Stack Overflow.

#markdown

Table des matières

À propos.....	1
Chapitre 1: Démarrer avec Markdown.....	2
Exemples.....	2
Formatage de base.....	2
Bonjour le monde.....	2
Bonjour le monde.....	2
Chapitre 2: Citations.....	4
Remarques.....	4
Exemples.....	4
Citant.....	4
Citations imbriquées.....	4
Chapitre 3: Code.....	5
Syntaxe.....	5
Exemples.....	5
Code en ligne.....	5
Mise en évidence de la syntaxe (StackExchange).....	5
Blocs de code en retrait.....	6
Blocs de code clôturés.....	6
Blocs de code en retrait dans les listes.....	7
Chapitre 4: Créer des listes.....	8
Exemples.....	8
Listes numérotées.....	8
Listes à puces.....	9
Listes imbriquées.....	10
Chapitre 5: Dialectes / Saveurs.....	11
Remarques.....	11
Exemples.....	11
Désactivation de la pile.....	11
Doxygen Markdown.....	12
@ref <nom> ["texte"].....	12

@section <nom-section> (titre de la section).....	12
Exemple.....	12
Sortie.....	12
introduction.....	12
Paragraphe du corps.....	13
GitHub Flavored Markdown.....	13
Exemples GFM.....	13
Mise en évidence de la syntaxe.....	13
Chapitre 6: Formatage du texte.....	15
Remarques.....	15
Exemples.....	15
Audacieux.....	15
Italique.....	15
Barré.....	15
Gras + italique.....	15
Règles horizontales.....	16
HTML.....	16
Rubrique 1.....	17
Indice / Superscript.....	17
Sauts de ligne et paragraphes.....	17
Chapitre 7: Images.....	18
Syntaxe.....	18
Remarques.....	18
Exemples.....	18
Images en ligne.....	18
Images de style de référence.....	19
Chapitre 8: Les entêtes.....	21
Exemples.....	21
En-têtes Atx-Style.....	21
Ceci est un en-tête de premier niveau (<h1>).....	21
Ceci est un en-tête de second niveau (<h2>).....	21

Etc.....	21
Ceci est un en-tête avec quelques hachages à la fin.....	21
En-têtes de style setext.....	21
Tout sur les chiens.....	21
Le premier roman.....	22
Un autre en-tête.....	22
Un autre en-tête.....	22
Chapitre 9: les tables.....	23
Remarques.....	23
Exemples.....	23
Créer une table.....	23
Pipe dans un contenu cellulaire.....	24
Chapitre 10: Marko Spoilers ou texte masqué.....	26
Remarques.....	26
Exemples.....	26
Créer des spoilers.....	26
Chapitre 11: Mise en relation.....	27
Exemples.....	27
Lien Inline.....	27
Lien de référence.....	27
Ancres nommés (lien vers un fragment de page).....	27
Crédits.....	28

À propos

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [markdown](#)

It is an unofficial and free Markdown ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official Markdown.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapitre 1: Démarrer avec Markdown

Exemples

Formatage de base

Chaque type de format est lié à une balise HTML.

Le premier en-tête fait référence à la `<h1>` et est visualisé comme suit:

Bonjour le monde

et il est écrit en soulignant le texte avec `=` :

```
Hello World
=====
```

ou en ajoutant `#` au texte:

```
# Hello World
```

La deuxième rubrique fait référence à la `<h2>` et est visualisée comme suit:

Bonjour le monde

et il est écrit soulignant le texte avec `-` :

```
Hello World
-----
```

ou en ajoutant `##` au texte:

```
## Hello World
```

(Le nombre de caractères `#` devant votre texte correspond au numéro de l'en-tête. `###` serait `<h3>`, etc.)

Le code Inline fait référence à la `<pre><code>` et est visible comme:

```
Hello World
```

et il est écrit sourroundinding le texte avec ``` :

```
`Hello World`
```

Les blocs de code font référence à la `<pre><code>` et sont visualisés comme suit:

```
Hello World
```

et il est écrit en ajoutant 4 (espaces) devant chaque ligne:

```
    Hello World
```

Le devis fait référence à la `<blockquote>` et est visualisé comme:

Bonjour le monde

et c'est écrit en ajoutant un `>` avant chaque paragraphe:

```
> Hello World
```

L'accent fait référence à la `` et est visualisé comme suit:

Bonjour le monde

et il est écrit sourroundinding le texte avec `*`

```
*Hello World*
```

Le gras fait référence à la `` et est visualisé comme suit:

Bonjour le monde

et c'est écrit en entourant le texte avec 2 `*`

```
**Hello World**
```

L'accent est mis sur la `` et est visualisée comme suit:

Bonjour le monde

et c'est écrit en entourant le texte avec 3 `*`

```
***Hello World***
```

Lire Démarrer avec Markdown en ligne: <https://riptutorial.com/fr/markdown/topic/529/demarrer-avec-markdown>

Chapitre 2: Citations

Remarques

- Certaines implémentations de Markdown, telles que Stack Exchange, prennent en charge les [devis spoiler](#), qui se ressemblent, mais masquent le contenu du devis jusqu'à ce que vous cliquiez dessus.

Exemples

Citant

En ajoutant > devant une ligne, vous pouvez créer un texte cité!

```
> I am a quote
```

Je suis une citation!

Citations imbriquées

Vous pouvez imbriquer des citations simplement en incluant plusieurs > caractères, comme ceci:

```
> Often makes no sense.  
> > Commenting above your quote...
```

Cela n'a souvent aucun sens.

Commentant au dessus de votre devis...

Lire Citations en ligne: <https://riptutorial.com/fr/markdown/topic/546/citations>

Chapitre 3: Code

Syntaxe

- *inline* : ``Code entre backticks`` ou `<code> Code entre ces balises HTML </code>`
- *multiligne* : `___Code qui est en retrait sur chaque ligne` ou `<pre> <code> Plusieurs lignes de code entre ces balises HTML </code> </pre>`

Exemples

Code en ligne

Markdown prend en charge l'ajout de code en ligne `like this` , obtenu en encapsulant du texte dans des raccourcis:

```
`code here`
```

Vous pouvez également placer votre code inline entre les balises HTML `<code> et </code>` .

Considérez le code de démarcation suivant:

```
`This` is an inline code block! <code>This</code> is one too!
```

Cela produirait la sortie suivante:

```
This est un bloc de code en ligne! This un aussi!
```

Si vous devez inclure un backtick dans le code en ligne, vous pouvez utiliser plusieurs backticks pour commencer et terminer le bloc de code en ligne, comme ceci:

```
``code containing a backtick (`) character``
```

Cela produirait la sortie suivante:

```
code containing a backtick (`) character
```

Utilisez `\` pour échapper à des backticks. Par exemple:

```
\`a`
```

sera rendu comme

```
`a`
```

Mise en évidence de la syntaxe (StackExchange)

Sur les sites StackExchange, les extraits de code peuvent fournir une mise en évidence de la syntaxe facultative. Sur des sites comme Stack Overflow, la langue par défaut est dérivée des balises utilisées dans la question associée (le cas échéant). En outre, le langage de mise en évidence de la syntaxe d'un extrait de code peut également être [défini en ajoutant un commentaire HTML au corps du texte](#) .

```
<!-- language: lang-vb -->

Sub ShowVB()
Dim i As Long
For i = 1 To 2
    If i = 3 Then
        MsgBox "How did that happen?"
    End If
Next
End Sub
```

Un tel commentaire changera le langage de mise en évidence de la syntaxe pour tous les extraits de code ultérieurs, ce qui peut être plutôt utile, en particulier lorsque plusieurs langues sont impliquées dans un article.

Le ci-dessus sera rendu avec la mise en évidence de Visual Basic en tant que:

```
Sub ShowVB()
Dim i As Long
For i = 1 To 2
    If i = 3 Then
        MsgBox "How did that happen?"
    End If
Next
End Sub
```

Blocs de code en retrait

Vous pouvez créer des extraits de code multilignes en mettant en retrait chaque ligne avec au moins quatre espaces ou un onglet:

```
#include <stdio.h>

int main() {
    printf("Hello World!\n");
    return 0;
}
```

Blocs de code clôturés

Certains analyseurs permettent de désigner le code en ajoutant trois raccourcis avant et après une section de code.

```
...
<p><em>This</em> is an HTML example!</p>
...
```

En option, de nombreux analyseurs permettent d'ajouter une mise en évidence de la syntaxe en spécifiant la langue du code immédiatement après le premier ensemble de raccourcis:

```
```html
<p>This is an HTML example!</p>
```
```

Résultat:

```
<p><em>This</em> is an HTML example!</p>
```

Blocs de code en retrait dans les listes

Lorsque vous ajoutez des blocs de code en retrait dans une [liste](#), vous devez d'abord insérer une ligne vierge, puis indenter le code plus avant. Les différentes versions de Markdown ont des règles différentes pour cela.

1. StackExchange exige que le code soit mis en retrait de 8 caractères au lieu des 4 habituels (*espaces remplacés par * pour plus de clarté*) :

```
1.*Listitem1
2.*Listitem2

*****code here
3.*Listitem3
```

2. Les spécifications telles que [CommonMark](#) nécessitent que le bloc de code soit indenté de 4 caractères à partir du début du texte de l'élément de liste. (*Espaces remplacés par * pour plus de clarté*) :

```
1.****Listitem1
2.****Listitem2

*****code here
3.****Listitem3
```

Lire Code en ligne: <https://riptutorial.com/fr/markdown/topic/553/code>

Chapitre 4: Créer des listes

Exemples

Listes numérotées

```
1. Lists
2. Can be
3. Numbered
```

1. Des listes
2. Peut être
3. Numéroté

Notez que les nombres eux-mêmes sont ignorés:

```
1. This is the first item
5. This is the fifth item
7. This is the seventh item
```

1. Ceci est le premier article
2. Ceci est le cinquième élément
3. Ceci est le septième article

Cependant, le premier numéro est utilisé pour lancer la numérotation:

```
3. This list starts at #3
2. However, this item is #4, despite being prefixed with `2.`
```

3. Cette liste commence à # 3
4. Cependant, cet élément est le numéro 4, malgré le préfixe 2.

Cela peut être utilisé pour reprendre une liste après avoir été interrompue par un autre texte / une image / un tableau / etc.

```
My very favorite colors are:
1. Blue
5. Red

(I like red because that's the best flavor of Skittle. But I digress.)

3. Orange
9. [etc]
```

Mes couleurs préférées sont:

1. Bleu
2. rouge

(J'aime le rouge parce que c'est la meilleure saveur de Skittle. Mais je m'égare.)

3. Orange

4. [etc]

Vous pouvez également annoter un élément de liste de cette manière sans interrompre la numérotation:

```
My very favorite colors are:  
1. Blue  
5. Red  
(I like red because that's the best flavor of Skittle. But I digress.)  
9. Orange  
11. [etc]
```

Notez qu'il n'y a pas de ligne vide entre 5. et l'instruction entre parenthèses, et qu'il y a deux espaces à la fin de "Red", nous obtenons donc:

Mes couleurs préférées sont:

1. Bleu

2. rouge

(J'aime le rouge parce que c'est la meilleure saveur de Skittle. Mais je m'égare.)

3. Orange

4. [etc]

Sans les deux espaces, cette section serait:

1. Bleu

2. Rouge (j'aime le rouge parce que c'est la meilleure saveur de Skittle. Mais je m'égare.)

... à cause de la manière dont Markdown traite les sauts de ligne.

Listes à puces

```
Characters for bulleted lists:  
* Asterisks  
+ Plus signs  
- Minus signs
```

Caractères pour les listes à puces:

- Astérisques
- Signes plus
- Signes moins

Notez s'il vous plaît:

Pour obtenir les meilleurs résultats, vous devez utiliser le même caractère car, comme vous

pouvez le voir dans l'exemple ci-dessous, différents signes font que la liste est interrompue.

```
Characters for bulleted lists:
```

```
* Asterisks 1
* Asterisks 2
* Asterisks 3
+ Plus signs 1
+ Plus signs 2
+ Plus signs 3
- Minus signs 1
- Minus signs 2
- Minus signs 3
```

Caractères pour les listes à puces:

- Astérisques 1
- Astérisques 2
- Astérisques 3

- Signes plus 1
- Signes plus 2
- Signes plus 3

- Signes moins 1
- Signes moins 2
- Signes moins 3

Listes imbriquées

```
1. Lists can be nested
  * Four spaces
    - Eight spaces
      + Twelve spaces
2. And back
```

1. Les listes peuvent être imbriquées

- Quatre espaces
 - Huit espaces
 - Douze espaces

2. Et retour

Lire Créer des listes en ligne: <https://riptutorial.com/fr/markdown/topic/554/creer-des-listes>

Chapitre 5: Dialectes / Saveurs

Remarques

Les variantes de Markdown avec une syntaxe différente sont appelées "saveurs". Les versions de Markdown sont répertoriées sur <https://github.com/jgm/CommonMark/wiki/Markdown-Flavors> .

Une saveur Markdown peut être implémentée dans divers langages de programmation et applications logicielles. Les bibliothèques qui implémentent un analyseur Markdown sont répertoriées à l'adresse <https://github.com/markdown/markdown.github.com/wiki/Implementations> et <https://www.w3.org/community/markdown/wiki/MarkdownImplementations> .

Plusieurs versions de Markdown et leurs différences sont documentées sur <http://flavoredmarkdown.com> .

Pour tester et comparer le code Markdown dans différentes versions et implémentations de ces versions, vous pouvez utiliser l'outil en ligne [Babelmark](#) .

Exemples

Désactivation de la pile

C'est la saveur du démarquage utilisé par Stack Overflow et d'autres sites Stack Exchange. Lorsque vous répondez à une question ou ajoutez de la documentation, vous utilisez cette balise. Cette réponse est faite de démarque SO

Voir la [documentation officielle](#)

Les principales choses que SO markdown ajoute sont sous "Ajouts Stack Exchange" sur cette page Web. En particulier, SO ajoute des *balises* comme `[tag:tag]` et des liens de raccourci comme `[meta]` (pas dans les documents), *spoilers* :

Ceci est un spoiler

```
>! This is a spoiler
```

et dévotions linguistiques personnalisées

```
<!-- language: java -->
...
This text is formatted as if it were Java code
...
```

```
This text is formatted as if it were Java code
```

Doxygen Markdown

Doxygen est un outil de documentation de code couramment utilisé (pour les langages tels que C++, C# et Java), qui prend également en charge l'utilisation de Markdown. En plus de la syntaxe Markdown standard, il existe un certain nombre d' [éléments spécifiques à Doxygen](#) .

Les fonctions principales sont l'utilisation des balises `@ref` pour les références et les éléments `@page` , `@section/@subsection` `@anchor` et `@anchor` auxquels ils peuvent faire référence.

@ref <nom> ["texte"]

Cet élément crée un lien (c.-à-d. Une référence) vers une section, une sous-section, une page ou une ancre nommée qui a été définie ailleurs dans la documentation. (voir [référence Doxygen](#))

Le premier paramètre (`name`) doit correspondre au nom de la section, de la sous-section, de la page ou de l'ancre auquel vous souhaitez créer un lien.

Le deuxième paramètre facultatif (`text`) doit être encapsulé entre guillemets et définir ce que le lien de référence apparaîtra sur la page. S'il n'est pas utilisé, le lien apparaîtra comme titre utilisé dans la référence.

@section <nom-section> (titre de la section)

Cet élément définit un nom de section. Il est visuellement équivalent à un élément `#` dans Markdown, mais il définira également une référence qui peut être liée à partir d'autres sections de votre documentation. (voir [référence Doxygen](#))

Le premier paramètre `section-name` définit le nom de référence pouvant être utilisé par l'élément `@ref` . Cela ne peut contenir aucun espace.

Le `section title` deuxième paramètre est une chaîne de mots (qui peut être séparée par des espaces) qui définit le titre de la section qui apparaîtra sur votre page.

Exemple

```
@section Intro Introduction

This is some text in my introduction.

@section Body Body Paragraph

This is some text in my body, where I refer to the @ref Intro.
```

Sortie

introduction

Ceci est un texte dans mon introduction.

Paragraphe du corps

C'est un texte dans mon corps, où je fais référence à l'introduction.

NB: Le mot Introduction ci-dessus apparaîtra comme un lien qui passera directement au titre Introduction.

GitHub Flavored Markdown

[GitHub Flavored Markdown](#) (parfois abrégé en GFM) facilite le travail avec les [balises](#) sur [GitHub.com](#).

Les principales caractéristiques de GFM incluent:

- indentation de code
- support de liste de tâches
- référencement facile des problèmes GitHub
- Détection automatique du nom d'utilisateur et du SHA de GitHub
- détection automatique d'url
- support emoji

Exemples GFM

Mise en évidence de la syntaxe

Avec Markdown, un bloc de code peut être généré avec trois backticks:

```
```
```

Sans coloration syntaxique, le code écrit en C apparaît comme ceci

```
#include <stdio.h>

int main(void)
{
 printf("Hello World\n");
 return (0);
}
```

Cependant, **avec la** coloration syntaxique, le code écrit en C apparaît comme ceci:

```
#include <stdio.h>

int main(void)
{
 printf("Hello World\n");
 return (0);
}
```

Pour mettre en évidence le code, ajoutez simplement un identificateur de langue facultatif pour activer la mise en évidence de la syntaxe dans votre bloc de code isolé.

```
```C
<code text here>
```
```

Ces améliorations visent à améliorer la qualité de la documentation et des conversations incluses dans les fichiers `README` , les [listes de diffusion](#) , les [requêtes d'extraction](#) et les [problèmes](#) sur la plate-forme.

Lire Dialectes / Saveurs en ligne: <https://riptutorial.com/fr/markdown/topic/1865/dialectes---saveurs>

---

# Chapitre 6: Formatage du texte

## Remarques

La mise en forme du texte dans le démarquage nécessite généralement des caractères au début et à la fin du texte.

## Exemples

### Audacieux

Le texte en gras peut être créé en entourant le texte par des astérisques doubles ou des soulignés doubles:

```
Bolded text
__Also bolded text__
```

Résultat:

**Texte en gras**

**Texte en gras**

### Italique

Les italiques peuvent être créés en entourant le texte d'astérisques ou de traits de soulignement:

```
Italicized text
Also italicized
```

Résultat:

*Texte en italique*

*Aussi en italique*

### Barré

Pour créer un ~~texte barré~~, entourez le texte de `~~double tildes~~`.

Remarque: sur StackExchange, ce format n'est pas inclus. Utilisez plutôt la balise HTML `<s>text</s>`. (Dans le chat, vous pouvez utiliser `---three hyphens---`.)

### Gras + italique

Creating ***bold italic*** text is simply a matter of using both ***bold*** (two asterisks) and *italic* (one asterisk) at the same time, for a total of three asterisks on either side of the text you want to format at once.

Créer du texte en ***italique gras*** consiste simplement à utiliser simultanément les deux **caractères gras** (deux astérisques) et *italique* (un astérisque), pour un total de trois astérisques de chaque côté du texte que vous souhaitez formater en même temps.

## Règles horizontales

You can create a horizontal break to divide your text by placing three (or more) underscores

—

or asterisks

\*\*\*

or hyphens

---

on their own line.

Vous pouvez créer une rupture horizontale pour diviser votre texte en plaçant trois traits de soulignement (ou plus)

\_\_\_\_\_

ou astérisques

\_\_\_\_\_

ou des traits d'union

\_\_\_\_\_

sur leur propre ligne.

Il peut y avoir des espaces entre les caractères et une règle horizontale peut être suivie immédiatement par une autre:

```
-- --
* * * *
and the spaces don't have to be evenly distributed
* ****
```

\_\_\_\_\_

et les espaces ne doivent pas être répartis uniformément

\_\_\_\_\_

## HTML

Certaines balises HTML peuvent également être utilisées dans Markdown.

- `<b>bold</b>` **gras**
- `<i>italic</i>` *italic*
- `<a href="http://stackoverflow.com/">link</a>` [lien](http://stackoverflow.com/)
- `<kbd>Ctrl</kbd>` Ctrl

Les ancres nommées peuvent également être utilisées pour faciliter la navigation dans le document. Notez que Stack Overflow Markdown ne semble pas le supporter.

```

Heading 1
Text under the heading
Click on a link like [Go to Heading 1](#heading1) to go to that named anchor.
```

## Rubrique 1

Texte sous l'en-tête Cliquez sur un lien comme Aller à l'en-tête 1 pour accéder à l'ancre nommée.

### Indice / Superscript

`x<sub>2</sub>` produit x <sub>2</sub>

`x<sup>2</sup>` produit x <sup>2</sup>

### Sauts de ligne et paragraphes

Terminez une ligne avec deux espaces ou plus pour créer un saut de ligne.

```
Ending a line with no spaces
or with just one space
doesn't create a line break.
Use two or more spaces
to create a line break.
```

```
Use an empty line to make a new paragraph.
```

Terminer une ligne sans espaces ou avec un seul espace ne crée pas un bec de ligne.

Utiliser deux espaces ou plus  
pour créer un saut de ligne.

Utilisez une ligne vide pour créer un nouveau paragraphe.

Lire Formatage du texte en ligne: <https://riptutorial.com/fr/markdown/topic/549/formatage-du-texte>

---

# Chapitre 7: Images

## Syntaxe

- ! [Alt text] (/ path / to / img.jpg)
- ! [Alt text] (/ path / to / img.jpg "Titre optionnel")
- ! [Alt text] [id]

... contenu intervenant ...

[id]: /path/to/img.jpg "Titre optionnel"

## Remarques

La syntaxe des images est la même que celle des [liens](#) , mais avec un point d'exclamation ! devant le texte (utilisé comme alt-text).

## Exemples

### Images en ligne

Dans ce type d'ajout d'image, l'URL de l'image est incluse à l'emplacement où l'image sera affichée. Si vous devez ajouter la même image plusieurs fois, vous devez inclure son URL à chaque fois.

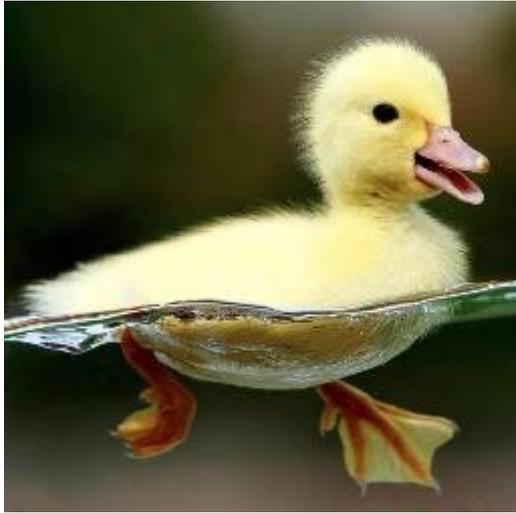
### Markdown Source

```
Picture of Duck:
```

```
![Duck] (http://i.stack.imgur.com/ukC2U.jpg)
```

### Sortie HTML

Photo de canard:



## Images de style de référence

Dans ce type d'ajout d'images, une image peut être utilisée plusieurs fois sans dupliquer son URL, ce qui en fait un bon choix lorsque vous utilisez une image plusieurs fois dans un document.

## Markdown Source

```
Picture of Duck:
```

```
![Duck][1]
```

```
Same picture of Duck:
```

```
![Same Duck][1]
```

```
[1]: http://i.stack.imgur.com/ukC2U.jpg
```

## Sortie HTML

Photo de canard:



Même photo de canard:



Lire Images en ligne: <https://riptutorial.com/fr/markdown/topic/698/images>

---

# Chapitre 8: Les entêtes

## Exemples

### En-têtes Atx-Style

Le texte préfixé par des signes de un à six livres (symboles de hachage, # ) devient un en-tête `<h1>` à `<h6>` , en fonction du nombre de signes dièse utilisés.

```
This is a first-level (`<h1>`) header
This is a second-level (`<h2>`) header
And so on.
```

---

**Ceci est un en-tête de premier niveau ( `<h1>` )**

**Ceci est un en-tête de second niveau ( `<h2>` )**

Etc.

Les en-têtes de style Atx peuvent éventuellement être fermés en ajoutant des signes dièse, qui sont ignorés.

```
This is a header with some trailing hashes
```

**Ceci est un en-tête avec quelques hachages à la fin**

À l'heure actuelle, la saveur de démarcation de Stack Exchange ne semble prendre en charge que trois niveaux d'en-tête ( ### ), bien que [Markdown supporte jusqu'à 6 niveaux](#).

### En-têtes de style setext

Pour créer un en-tête de premier niveau ( `<h1>` ), utilisez le signe égal ( = ) dans une ligne sous votre texte:

```
All About Dogs
=====
```

---

**Tout sur les chiens**

Utilisez des tirets ( - ) pour les en-têtes de deuxième niveau ( `<h2>` ):

The Debut Novel

-----

## Le premier roman

La ligne sous l'en-tête peut avoir n'importe quelle longueur.

```
Another header
```

```
--
```

```
Another header
```

```
-
```

## Un autre en-tête

## Un autre en-tête

Lire Les entêtes en ligne: <https://riptutorial.com/fr/markdown/topic/532/les-entetes>

# Chapitre 9: les tables

## Remarques

Les tables ne sont prises en charge que dans certaines versions de Markdown, y compris [Markdown Extra](#) et [Github Flavored Markdown](#) , mais pas dans la [syntaxe Markdown d'origine](#) ou dans [CommonMark](#) .

Les tables de démarques ne sont [pas non plus prises en charge sur les sites Stack Exchange](#) (à l'exception de [la version bêta de la documentation](#) ).

## Exemples

### Créer une table

Les tables de démarques sont physiquement représentées à l'aide d'un tiret - pour séparer la ligne d'en-tête du contenu et de la colonne | pour les colonnes.

| Colonne | Colonne |
|---------|---------|
| Cellule | Cellule |

est produit par

```
Column	Column
Cell | Cell
```

Vous pouvez également remplir une table comme vous le souhaitez -

| Lettre | Chiffre | Personnage |
|--------|---------|------------|
| une    | 4       | \$         |
|        | 365     | (          |
| b      |         | ^          |

Le code de cette table:

```
Letter	Digit	Character
a | 4 | $
 | 365 | (
b | | ^
```

Markdown ignore l'espacement. Le même tableau pourrait être écrit comme ceci:

```
Letter	Digit	Character
a	4	$
365	(
b| |^
```

et toujours afficher le même:

| Lettre | Chiffre | Personnage |
|--------|---------|------------|
| une    | 4       | \$         |
|        | 365     | (          |
| b      |         | ^          |

**NOTE** : si vous avez besoin d'une colonne vide, vous devez ajouter un espace entre les tuyaux

Comme vous pouvez le voir, le code de la table n'a pas besoin de représenter l'espacement de la table - ce qui est accompli dans le cadre du démarquage.

Vous devez vouloir aligner le contenu d'une table. Tout ce que vous avez à faire est d'ajouter des deux points de cette manière:

Aligner la colonne:

: permet d'aligner une colonne. L'alignement à gauche est la norme.

```
Column	Column	Column
Left | Center | Right
align | align | align
```

| Colonne   | Colonne | Colonne |
|-----------|---------|---------|
| La gauche | Centre  | Droite  |
| aligner   | aligner | aligner |

## Pipe dans un contenu cellulaire

Si vous souhaitez utiliser un caractère de caractère ( | ) dans le contenu d'une cellule, vous devez y échapper avec une barre oblique inverse.

```
Column	Column
\| Cell \| \| Cell \|
```

Cela se traduit par le tableau suivant:

| Colonne | Colonne |
|---------|---------|
| Cellule | Cellule |

Lire les tables en ligne: <https://riptutorial.com/fr/markdown/topic/533/les-tables>

---

# Chapitre 10: Marko Spoilers ou texte masqué

## Remarques

Cette syntaxe de spoiler est spécifique à Stack Exchange et ne fait pas partie d'une syntaxe de markdown standard.

## Exemples

### Créer des spoilers

Les spoilers sont utilisés pour masquer du texte ou des images qui auraient un impact négatif sur l'expérience d'un autre utilisateur. Ils peuvent être créés avec >!

```
>!This is hidden until your cursor hovers on top of it
```

Ceci est caché jusqu'à ce que votre curseur se place dessus

Remarque: Cela ne fait pas partie des spécifications de balisage standard et ne sera pas rendu dans tous les analyseurs de démarques.

Lire [Marko Spoilers ou texte masqué en ligne](https://riptutorial.com/fr/markdown/topic/531/marko-spoilers-ou-texte-masque): <https://riptutorial.com/fr/markdown/topic/531/marko-spoilers-ou-texte-masque>

---

# Chapitre 11: Mise en relation

## Exemples

### Lien Inline

Le formulaire pour un lien en démarque est le suivant.

```
[Shown Text] (Link)
```

Par exemple, [ceci vous mènera à Example.com](#) est créé avec

```
[This will take you to Example.com] (http://www.example.com)
```

### Lien de référence

Les URL des liens peuvent être spécifiées ultérieurement dans le document.

### Réduction

```
[Text1][1] will link to the first link, and [Text2][2] to the second.
You [can reuse][1] names, and give longer names [like this one][a link].
You can also link text [like this] without giving the reference an explicit name.
```

```
[1]: http://www.google.com
[2]: http://stackoverflow.com/
[a link]: http://example.org/
[like this]: http://stackexchange.com/
```

### Sortie

**Text1** sera lié au premier lien et **Text2** au second. Vous pouvez réutiliser des noms et donner des noms plus longs comme celui-ci . Vous pouvez également lier un texte comme celui-ci sans donner à la référence un nom explicite.

### Ancres nommés (lien vers un fragment de page)

créer une destination avec

```

```

lien vers la destination

```
[link text] (#destinationLinkName)
```

Lire Mise en relation en ligne: <https://riptutorial.com/fr/markdown/topic/545/mise-en-relation>

# Crédits

| S. No | Chapitres                      | Contributeurs                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-------|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1     | Démarrer avec Markdown         | <a href="#">Community</a> , <a href="#">Giacomo Garabello</a> , <a href="#">hairboat</a> , <a href="#">J F</a> , <a href="#">Jeremy Banks</a> , <a href="#">jkdev</a> , <a href="#">Matt Clark</a> , <a href="#">RamenChef</a> , <a href="#">Raystafarian</a> , <a href="#">TiernanO</a>                                                                                                                                                                                     |
| 2     | Citations                      | <a href="#">intboolstring</a> , <a href="#">Karsten 7.</a> , <a href="#">OverZealous</a> , <a href="#">Quill</a> , <a href="#">Wolfgang</a>                                                                                                                                                                                                                                                                                                                                  |
| 3     | Code                           | <a href="#">Christopher Muller</a> , <a href="#">David Fullerton</a> , <a href="#">DavidG</a> , <a href="#">ganesshkumar</a> , <a href="#">J F</a> , <a href="#">Jens Erat</a> , <a href="#">jkdev</a> , <a href="#">John Slegers</a> , <a href="#">Keith Hall</a> , <a href="#">Luke Hefson</a> , <a href="#">Marco Bonelli</a> , <a href="#">Mario</a> , <a href="#">Nathan Arthur</a> , <a href="#">Raystafarian</a> , <a href="#">Sweeper</a> , <a href="#">Wolfgang</a> |
| 4     | Créer des listes               | <a href="#">Giacomo Garabello</a> , <a href="#">hairboat</a> , <a href="#">Raystafarian</a> , <a href="#">Wolfgang</a>                                                                                                                                                                                                                                                                                                                                                       |
| 5     | Dialectes / Saveurs            | <a href="#">Doc</a> , <a href="#">gemmakbarlow</a> , <a href="#">jkdev</a> , <a href="#">josephsw</a> , <a href="#">yeungegs</a>                                                                                                                                                                                                                                                                                                                                             |
| 6     | Formatage du texte             | <a href="#">Ajedi32</a> , <a href="#">Ana</a> , <a href="#">chrki</a> , <a href="#">hairboat</a> , <a href="#">Karsten 7.</a> , <a href="#">Keith Hall</a> , <a href="#">Null</a> , <a href="#">Paul Jacobson</a> , <a href="#">Raystafarian</a> , <a href="#">scenography</a> , <a href="#">Wolfgang</a>                                                                                                                                                                    |
| 7     | Images                         | <a href="#">manetsus</a> , <a href="#">Nathan Arthur</a> , <a href="#">Wolfgang</a>                                                                                                                                                                                                                                                                                                                                                                                          |
| 8     | Les entêtes                    | <a href="#">Alex Warren</a> , <a href="#">chrki</a> , <a href="#">hairboat</a> , <a href="#">jkdev</a> , <a href="#">Nathan Arthur</a> , <a href="#">Raystafarian</a> , <a href="#">scenography</a> , <a href="#">Typothecary</a> , <a href="#">Wolfgang</a>                                                                                                                                                                                                                 |
| 9     | les tables                     | <a href="#">CD..</a> , <a href="#">ganesshkumar</a> , <a href="#">Giacomo Garabello</a> , <a href="#">Hexaholic</a> , <a href="#">J F</a> , <a href="#">jkdev</a> , <a href="#">Raystafarian</a>                                                                                                                                                                                                                                                                             |
| 10    | Marko Spoilers ou texte masqué | <a href="#">ganesshkumar</a> , <a href="#">Raystafarian</a> , <a href="#">Wolfgang</a>                                                                                                                                                                                                                                                                                                                                                                                       |
| 11    | Mise en relation               | <a href="#">Ander Biguri</a> , <a href="#">bumbeishvili</a> , <a href="#">intboolstring</a> , <a href="#">Nathan Arthur</a> , <a href="#">Thunderforge</a> , <a href="#">Wolfgang</a>                                                                                                                                                                                                                                                                                        |