

GUIDE COMPLET • NIVEAU DÉBUTANT

Markdown

Apprendre le Markdown

Écrire du texte mis en forme, simplement, avec quelques caractères. Le guide pas-à-pas pour tout comprendre et commencer dès aujourd'hui.

Syntaxe expliquée

Exemples avant / après

Aide-mémoire

Exercices

Un cours progressif, illustré et sans prérequis — pour **prendre des notes**, rédiger une **documentation**, un **README** ou un **blog**.

Sommaire

- 01 **Qu'est-ce que le Markdown ?** — définition, intérêt, usages, histoire
- 02 **Premiers pas** — comment ça marche et où écrire
- 03 **Les titres** — structurer un document
- 04 **Paragraphe et sauts de ligne**
- 05 **Mettre le texte en forme** — gras, italique, barré
- 06 **Les listes** — à puces, numérotées, tâches
- 07 **Les liens**
- 08 **Les images**
- 09 **Afficher du code**
- 10 **Les citations**
- 11 **Les tableaux**
- 12 **Autres éléments utiles** — séparateurs, échappement, HTML
- 13 **Les variantes de Markdown** — CommonMark & GitHub
- 14 **Bonnes pratiques**
- 15 **Aide-mémoire complet**
- 16 **Exercices & corrigés**
- 17 **Pour aller plus loin**

Qu'est-ce que le Markdown ?

Le Markdown est un langage de mise en forme très simple : on écrit du texte normal, et quelques petits symboles ajoutés ici et là indiquent « ceci est un titre », « ce mot est en gras », « voici une liste ». C'est tout. Pas de boutons, pas de souris, pas de menus.

Une définition simple

Markdown est un **langage de balisage léger**. « Léger » signifie qu'il utilise des caractères ordinaires de votre clavier (comme #, * ou -) plutôt que des balises compliquées. Vous écrivez dans un simple fichier texte, et un logiciel se charge de transformer ce texte en un document joliment mis en forme — le plus souvent une page web (du HTML).

L'idée maîtresse, voulue par son créateur, est que le texte source reste **lisible et agréable tel quel**, même sans être transformé. Quand vous écrivez `**important**`, on comprend tout de suite que le mot est mis en avant, sans avoir besoin de le voir « rendu ».

VOUS ÉCRIVEZ (TEXTE BRUT)

```
# Ma liste de courses

J'ai besoin de lait, de pain et de _pommes_.

- Tomates
- Fromage
```

CE QUE LE LECTEUR VOIT (RENDU)

Ma liste de courses

J'ai besoin de **lait**, de pain et de pommes.

- Tomates
- Fromage

Pourquoi l'utiliser ?

Le Markdown s'est imposé partout pour de bonnes raisons. En voici les principales :

- ◆ **C'est rapide.** Vos mains restent sur le clavier. Pas besoin de cliquer sur un bouton « gras » : vous tapez deux astérisques et vous continuez à écrire.
- ◆ **C'est lisible.** Même sans mise en forme, le fichier source se lit naturellement. Un titre ressemble à un titre, une liste ressemble à une liste.

- ◆ **C'est durable et universel.** Un fichier Markdown est un simple fichier texte. Il s'ouvre sur n'importe quel ordinateur, n'importe quel système, aujourd'hui comme dans vingt ans. Aucun risque qu'un logiciel disparaisse et rende vos documents illisibles.
- ◆ **C'est portable.** Le même texte s'affiche correctement sur des dizaines de plateformes différentes (sites, applications de notes, messageries...).
- ◆ **Ça se transforme facilement.** Un document Markdown peut être converti en page web, en PDF, en document Word, en diapositives... à partir d'une seule source.

Où le rencontre-t-on ?

Probablement plus souvent que vous ne le pensez. Le Markdown est utilisé pour :

- les fichiers **README** qui présentent les projets informatiques (sur GitHub, GitLab...);
- la **documentation** technique et les wikis ;
- les applications de **prise de notes** (Obsidian, Notion, Joplin, Bear...);
- les messageries professionnelles comme **Slack** ou **Discord** (qui en utilisent une partie) ;
- de nombreux **blogs** et générateurs de sites web ;
- les forums, les commentaires, et même les échanges avec des assistants IA.

Markdown ou traitement de texte ?

Un logiciel comme Word vous montre directement le résultat final : vous voyez le gras pendant que vous l'appliquez. C'est ce qu'on appelle le « WYSIWYG » (what you see is what you get). Le Markdown fonctionne autrement : vous écrivez des instructions dans le texte, et le rendu est calculé ensuite. En échange de ce petit changement d'habitude, vous gagnez en rapidité, en légèreté des fichiers et en liberté totale sur l'outil utilisé.

À RETENIR

Markdown ne remplace pas Word pour tout. Il brille quand le contenu compte plus que la mise en page sophistiquée : notes, documentation, articles, listes, README. Pour une plaquette commerciale très graphique, un outil de PAO reste plus adapté.

Un peu d'histoire

Le Markdown a été créé en **2004** par John Gruber, avec l'aide d'Aaron Swartz. L'objectif : permettre d'écrire pour le web sans avoir à taper du HTML à la main. Le langage a connu un tel succès que chaque plateforme a fini par ajouter ses propres petites variantes — au point qu'il a fallu, plus tard, le standardiser. Nous reviendrons sur ces « variantes » au chapitre 13 ; pour débiter, sachez simplement que **l'immense majorité de la syntaxe est commune partout.**

Premiers pas

Avant de plonger dans la syntaxe, comprenons en deux minutes comment le Markdown « fonctionne » et où vous pouvez l'écrire dès maintenant.

Le principe : source → rendu

Tout repose sur deux états du même contenu. D'un côté, le **texte source** : ce que vous tapez, rempli de petits symboles. De l'autre, le **rendu** : le document mis en forme qu'un programme produit à partir de la source. Tout au long de ce cours, chaque notion sera présentée sous cette forme « ce que vous écrivez » puis « ce que ça donne ». Prenez le réflexe de comparer les deux.

Où écrire du Markdown ?

Vous n'avez besoin de rien installer pour essayer. Voici trois manières de commencer :

- ◆ **Dans un éditeur en ligne.** Des sites comme dillinger.io ou stackedit.io affichent la source à gauche et le rendu à droite, en direct. Idéal pour apprendre.
- ◆ **Dans une application de notes.** Obsidian, Notion, Joplin ou Typora comprennent le Markdown nativement.
- ◆ **Dans un simple fichier texte.** Créez un fichier avec l'extension `.md` (par exemple `notes.md`) et ouvrez-le dans un éditeur de code comme VS Code.

ASTUCE

Gardez un éditeur en ligne ouvert dans un onglet pendant la lecture de ce cours. Recopiez chaque exemple « Vous écrivez » et observez le rendu apparaître. On apprend le Markdown bien plus vite en testant qu'en lisant.

La règle d'or de la syntaxe

Une seule chose à garder en tête avant tout le reste : en Markdown, **les lignes vides comptent**. Une ligne vide sépare deux blocs (deux paragraphes, un titre et un paragraphe, etc.). De même, certains symboles doivent être **suivis d'une espace** pour fonctionner. Par exemple, `# Titre` (avec une espace) crée un titre, alors que `#Titre` (sans espace) ne fait rien de spécial sur de nombreux systèmes.

ATTENTION

Le détail qui piège tous les débutants : oublier l'espace après un symbole, ou coller deux blocs sans ligne vide entre eux. Si un rendu vous semble faux, vérifiez d'abord ces deux points.

CHAPITRE 03

Les titres

Les titres structurent un document : ils créent une hiérarchie, comme les chapitres et les sections d'un livre. C'est souvent le premier élément qu'on apprend, et le plus utile.

La syntaxe

On crée un titre en plaçant un ou plusieurs croisillons `#` en début de ligne, suivis d'une **espace**, puis du texte. Le nombre de croisillons détermine le niveau : un seul pour le titre principal, deux pour un sous-titre, et ainsi de suite jusqu'à six.

VOUS ÉCRIVEZ

```
# Titre de niveau 1
## Titre de niveau 2
### Titre de niveau 3
#### Titre de niveau 4
```

RÉSULTAT

Titre de niveau 1

Titre de niveau 2

Titre de niveau 3

Titre de niveau 4

Plus le niveau augmente, plus le titre est « petit » et secondaire. Le niveau 1 est généralement réservé au titre unique du document ; on utilise les niveaux 2 et 3 pour les sections et sous-sections.

ATTENTION

N'oubliez pas l'espace après les `#`. `## Section` fonctionne ; `##Section` risque de s'afficher tel quel, sans devenir un titre.

ASTUCE

Ne « sautez » pas de niveaux (par exemple un titre 1 suivi directement d'un titre 4). Une hiérarchie régulière rend le document plus clair pour les lecteurs et pour les outils qui génèrent automatiquement un sommaire.

Paragraphe et sauts de ligne

Cela paraît évident, mais la gestion des paragraphes et des retours à la ligne est la source numéro un de surprises chez les débutants. Voyons précisément comment ça marche.

Créer un paragraphe

Un paragraphe est simplement du texte. Pour **séparer deux paragraphes**, on laisse une **ligne vide** entre eux. C'est la ligne vide qui crée la séparation, pas le simple retour à la ligne.

VOUS ÉCRIVEZ

```
Ceci est le premier paragraphe.  
  
Ceci est le second paragraphe,  
séparé du premier par une ligne vide.
```

RÉSULTAT

Ceci est le premier paragraphe.

Ceci est le second paragraphe, séparé du premier par une ligne vide.

Remarquez bien : dans l'exemple, le second paragraphe est écrit sur deux lignes dans la source, mais il s'affiche sur une seule. **Un simple retour à la ligne dans la source ne crée pas de retour à la ligne dans le rendu.** Le texte est « recollé ».

Forcer un retour à la ligne

Si vous voulez réellement aller à la ligne à l'intérieur d'un même paragraphe (sans créer un nouveau paragraphe), il existe deux méthodes. La plus fiable : terminer la ligne par une **contre-oblique** `\`. L'autre méthode, plus discrète, consiste à laisser **deux espaces** en fin de ligne.

VOUS ÉCRIVEZ

```
Roses sont rouges,\  
Violettes sont bleues.
```

RÉSULTAT

Roses sont rouges,
Violettes sont bleues.

À RETENIR

Ligne vide = nouveau paragraphe (un grand espacement). **Contre-oblique** ou **deux espaces** en fin de ligne = simple retour à la ligne (texte serré, même bloc).

Mettre le texte en forme

Voici comment mettre des mots en valeur : le gras pour ce qui est important, l'italique pour une nuance ou un terme étranger, et le barré pour signaler ce qui n'est plus d'actualité.

Gras et italique

On entoure le texte de symboles. Une paire d'**astérisques** de chaque côté donne l'italique ; une double paire donne le **gras** ; une triple paire combine les deux. On peut aussi utiliser le tiret bas `_` à la place de l'astérisque, le résultat est identique.

VOUS ÉCRIVEZ

```
Du texte *en italique*.  
Du texte **en gras**.  
Du texte ***gras et italique***.  
On peut aussi écrire _italique_ et __gras__.
```

RÉSULTAT

Du texte en italique.

Du texte **en gras**.

Du texte **gras et italique**.

On peut aussi écrire italique et **gras**.

Le barré

Pour rayer du texte, on l'entoure de **deux tildes** `~` de chaque côté. Pratique pour montrer une correction ou une tâche annulée.

VOUS ÉCRIVEZ

```
Le prix est de ~50 euros~ 35 euros.
```

RÉSULTAT

Le prix est de ~~50 euros~~ 35 euros.

ASTUCE

Privilégiez l'astérisque `*` pour le gras et l'italique. Le tiret bas `_` peut poser problème au milieu d'un mot (par exemple dans un nom de fichier comme `mon_fichier_2`), où il risque d'être interprété par erreur comme de l'italique.

ATTENTION

Le barré `~` ne fait pas partie du Markdown d'origine : c'est une extension. Il fonctionne sur la plupart des plateformes modernes (GitHub, Discord, Slack...), mais pas sur absolument toutes.

Les listes

Les listes organisent l'information en points distincts. Markdown en propose deux types principaux — à puces et numérotées — ainsi qu'une variante très pratique : les listes de tâches.

Listes à puces (non ordonnées)

Commencez chaque ligne par un tiret `-` suivi d'une espace. Les symboles `*` et `+` fonctionnent aussi, mais le tiret est le plus courant et le plus lisible.

VOUS ÉCRIVEZ

- Pommes
- Bananes
- Oranges

RÉSULTAT

- Pommes
- Bananes
- Oranges

Listes numérotées (ordonnées)

Commencez chaque ligne par un chiffre suivi d'un point et d'une espace, comme `1.`. Détail surprenant et bien pratique : **les numéros que vous écrivez n'ont pas d'importance** ; le rendu renumérote automatiquement dans l'ordre. Vous pouvez donc tout écrire avec des `1.` et laisser le système faire le calcul.

VOUS ÉCRIVEZ

1. Préchauffer le four
1. Préparer la pâte
1. Enfournier 25 minutes

RÉSULTAT

1. Préchauffer le four
2. Préparer la pâte
3. Enfournier 25 minutes

Listes imbriquées

Pour créer un sous-niveau, **décalez** la ligne vers la droite (généralement de deux à quatre espaces) avant le tiret. On peut ainsi emboîter listes à puces et listes numérotées.

VOUS ÉCRIVEZ

- ```
- Fruits
 - Pommes
 - Bananes
- Légumes
 - Carottes
```

### RÉSULTAT

- Fruits
  - Pommes
  - Bananes
- Légumes
  - Carottes

## Listes de tâches (cases à cocher)

Très utiles pour les to-do lists, elles s'écrivent comme une liste à puces, en ajoutant `[ ]` pour une case vide ou `[x]` pour une case cochée.

### VOUS ÉCRIVEZ

- ```
- [x] Acheter le billet
- [x] Faire la valise
- [ ] Partir à l'aéroport
```

RÉSULTAT

- Acheter le billet
- Faire la valise
- Partir à l'aéroport

ATTENTION

Comme le barré, les listes de tâches sont une extension (GitHub, Obsidian, Notion les gèrent). Sur une plateforme qui ne les prend pas en charge, elles s'afficheront comme une liste à puces ordinaire avec des crochets visibles.

CHAPITRE 07

Les liens

Un lien associe un texte cliquable à une adresse web. La syntaxe se retient facilement : le texte entre crochets, l'adresse entre parenthèses, collés l'un à l'autre.

Le lien classique

On écrit le texte visible entre **crochets** `[]`, immédiatement suivi de l'adresse entre **parenthèses** `()`.

VOUS ÉCRIVEZ

```
Visitez [le site de Wikipédia](https://fr.wikipedia.org).
```

RÉSULTAT

Visitez [le site de Wikipédia](https://fr.wikipedia.org).

Ajouter une infobulle

On peut ajouter un titre qui apparaît au survol de la souris, en l'indiquant entre guillemets après l'adresse.

VOUS ÉCRIVEZ

```
[Notre blog](https://exemple.fr "Cliquez pour lire nos articles")
```

RÉSULTAT

[Notre blog](https://exemple.fr) (une infobulle « Cliquez pour lire nos articles » apparaît au survol)

Le lien automatique

Pour transformer une adresse en lien sans texte particulier, il suffit de l'entourer de **chevrons** `< >`. Sur beaucoup de plateformes modernes, une adresse complète devient même un lien automatiquement, sans aucun symbole.

VOUS ÉCRIVEZ

```
<https://exemple.fr>
```

RÉSULTAT

<https://exemple.fr>

Les liens de référence

Quand un texte contient beaucoup de liens, on peut séparer le texte de l'adresse pour garder la source lisible. On donne une étiquette au lien, et on définit l'adresse plus loin (souvent en bas du document).

VOUS ÉCRIVEZ

```
J'utilise [cet éditeur][1] tous les jours.
```

```
[1]: https://exemple.fr
```

RÉSULTAT

J'utilise [cet éditeur](#) tous les jours.

ASTUCE

Pour un lien interne vers un autre fichier Markdown du même dossier, indiquez simplement son nom : `[Voir le guide](guide.md)` .

CHAPITRE 08

Les images

Insérer une image se fait presque comme un lien. La seule différence : un point d'exclamation `!` placé juste devant.

La syntaxe

On écrit `!`, puis le **texte alternatif** entre crochets (la description de l'image), puis le **chemin ou l'adresse** de l'image entre parenthèses. Le texte alternatif s'affiche si l'image ne se charge pas et aide les lecteurs malvoyants : ne le négligez pas.

VOUS ÉCRIVEZ

```
![Un chaton roux endormi](images/chaton.jpg)
```

RÉSULTAT



L'image chaton.jpg s'affiche ici.
Texte alternatif : « Un chaton roux endormi »

Une image cliquable

Pour rendre une image cliquable, on combine les deux syntaxes : on place le code de l'image à l'intérieur des crochets d'un lien.

VOUS ÉCRIVEZ

```
[[Logo](logo.png)](https://exemple.fr)
```

RÉSULTAT

L'image logo.png s'affiche et devient un lien vers exemple.fr.

À RETENIR

Markdown ne permet pas, à lui seul, de redimensionner une image ou de la centrer. Pour cela, il faut recourir à du HTML (voir le chapitre 12) ou aux options propres à votre plateforme.

Afficher du code

Afficher du code (ou tout texte qui ne doit pas être interprété) dans une police à chasse fixe est l'un des grands points forts du Markdown. Deux formes existent : en ligne, et en bloc.

Code en ligne

Pour mettre un fragment court en évidence au milieu d'une phrase, entourez-le d'un **accent grave** ``` (la « backtick », touche souvent en haut à gauche du clavier) de chaque côté.

VOUS ÉCRIVEZ

Tapez la commande ``cd dossier`` pour vous déplacer.

RÉSULTAT

Tapez la commande `cd dossier` pour vous déplacer.

Bloc de code

Pour plusieurs lignes, encadrez-les par **trois accents graves** ````` sur une ligne avant et une ligne après. On parle de « bloc clôturé ». Tout ce qui se trouve entre les deux est affiché tel quel, sans interprétation.

VOUS ÉCRIVEZ

```
```  
def bonjour():
 print("Salut !")
```
```

RÉSULTAT

```
def bonjour():  
    print("Salut !")
```

La coloration syntaxique

En précisant le **langage** juste après les trois premiers accents graves, de nombreuses plateformes colorent automatiquement le code pour le rendre plus lisible.

VOUS ÉCRIVEZ

```
```python
nombre = 42
print(nombre)
```
```

RÉSULTAT

```
nombre = 42
print(nombre)
```

ASTUCE

Le code en ligne est aussi très pratique hors programmation : pour citer un nom de fichier, un raccourci clavier, un bouton de l'interface, ou pour montrer un symbole Markdown sans qu'il soit interprété.

CHAPITRE 10

Les citations

Une citation (ou « bloc de citation ») met en retrait un passage, typiquement pour rapporter les propos de quelqu'un ou mettre en avant une remarque.

Citation simple

Commencez la ligne par un chevron `>` suivi d'une espace.

VOUS ÉCRIVEZ

```
> La simplicité est la sophistication suprême.
```

RÉSULTAT

La simplicité est la sophistication suprême.

Citation sur plusieurs paragraphes

Pour qu'une ligne vide reste à l'intérieur de la citation, ajoutez un chevron sur la ligne vide elle-même.

VOUS ÉCRIVEZ

```
> Premier paragraphe de la citation.  
>  
> Second paragraphe de la même citation.
```

RÉSULTAT

Premier paragraphe de la citation.
Second paragraphe de la même citation.

Citation imbriquée

On peut imbriquer une citation dans une autre en doublant le chevron `>>`. Et à l'intérieur d'une citation, toute la mise en forme habituelle (gras, listes, titres...) reste disponible.

VOUS ÉCRIVEZ

```
> Il m'a dit :  
>> « Apprends le **Markdown** ! »
```

RÉSULTAT

```
Il m'a dit :  
« Apprends le Markdown ! »
```

CHAPITRE 11

Les tableaux

Les tableaux organisent des données en lignes et colonnes. Leur syntaxe utilise la barre verticale `|` (souvent accessible avec `AltGr` + `6`) pour séparer les cellules.

La structure

Un tableau comporte trois éléments : une ligne d'**en-têtes**, une ligne de **tirets** qui sépare les en-têtes du contenu, puis les lignes de **données**. Chaque colonne est délimitée par une barre `|`.

VOUS ÉCRIVEZ

```
Prénom	Ville	Âge
Alice	Lyon	30
Bruno	Marseille	25
```

RÉSULTAT

| Prénom | Ville | Âge |
|--------|-----------|-----|
| Alice | Lyon | 30 |
| Bruno | Marseille | 25 |

Aligner les colonnes

On contrôle l'alignement grâce à la position des **deux-points** `:` sur la ligne de séparation : à gauche pour aligner à gauche, des deux côtés pour centrer, à droite pour aligner à droite.

VOUS ÉCRIVEZ

```
Gauche	Centré	Droite
a	b	c
```

RÉSULTAT

| Gauche | Centré | Droite |
|--------|--------|--------|
| a | b | c |

ASTUCE

Les colonnes n'ont pas besoin d'être parfaitement alignées dans la source : le rendu fonctionne même si vos barres ne tombent pas droit. Aligner visuellement aide simplement à la lecture. Pour les grands tableaux, un générateur en ligne fait gagner beaucoup de temps.

ATTENTION

Les tableaux sont une extension (GitHub, GitLab, la plupart des applications de notes). Ils ne figurent pas dans le Markdown d'origine ni dans la version la plus stricte de la norme CommonMark.

Autres éléments utiles

Quelques outils supplémentaires complètent votre boîte à outils : le séparateur horizontal, l'échappement des caractères spéciaux, et l'usage du HTML quand le Markdown ne suffit pas.

La ligne de séparation

Pour tracer un trait horizontal qui sépare deux sections, placez **trois tirets** `---` (ou trois astérisques, ou trois tirets bas) seuls sur une ligne, avec une ligne vide avant et après.

VOUS ÉCRIVEZ

```
Fin de la première partie.
```

```
---
```

```
Début de la seconde partie.
```

RÉSULTAT

Fin de la première partie.

Début de la seconde partie.

Échapper les caractères spéciaux

Comment écrire un astérisque ou un croisillon littéralement, sans qu'il soit interprété ? On le fait précéder d'une **contre-oblique** `\`. Cela « neutralise » le caractère.

VOUS ÉCRIVEZ

```
Pour mettre en gras, on écrit \\**mot\**.
```

RÉSULTAT

Pour mettre en gras, on écrit ****mot****.

Insérer du HTML

Quand le Markdown atteint ses limites (centrer un élément, régler la taille d'une image...), la plupart des plateformes acceptent que vous écriviez directement du **HTML** au milieu de votre texte. C'est

une porte de secours pratique, mais à utiliser avec parcimonie pour ne pas perdre la simplicité qui fait l'intérêt du Markdown.

VOUS ÉCRIVEZ

```
<p align="center">Ce texte est centré.</p>
```

RÉSULTAT

Ce texte est centré.

ATTENTION

Pour des raisons de sécurité, certaines plateformes (forums, messageries) **désactivent le HTML**. Si votre HTML s'affiche tel quel au lieu d'être interprété, c'est probablement le cas : revenez alors à du Markdown pur.

Les variantes de Markdown

Vous avez peut-être remarqué que certaines fonctionnalités sont notées « extension ». C'est parce que Markdown n'est pas un standard unique, mais une famille de « saveurs » (flavors) légèrement différentes. Pas d'inquiétude : 90 % de ce que vous avez appris est commun à toutes.

Pourquoi des variantes ?

Le Markdown d'origine de 2004 était volontairement minimal et laissait quelques zones d'ombre. Chaque plateforme a donc comblé les manques à sa façon, créant des dialectes. Deux d'entre eux dominent aujourd'hui.

CommonMark — la norme de référence

CommonMark est une **spécification précise** née pour mettre fin aux ambiguïtés : elle définit rigoureusement comment chaque symbole doit se comporter. C'est le socle commun, le « noyau » du langage. CommonMark couvre tout l'essentiel — titres, listes, liens, images, gras, italique, code — mais reste volontairement épuré : il **n'inclut pas** les tableaux, par exemple.

GitHub Flavored Markdown (GFM) — la version la plus répandue

Le « Markdown de GitHub » est un **sur-ensemble** de CommonMark : il contient tout CommonMark, et ajoute des fonctionnalités très utiles au travail collaboratif et à la documentation. C'est, en pratique, le dialecte que vous rencontrez le plus souvent aujourd'hui. Ses ajouts principaux :

- ◆ les **tableaux** ;
- ◆ les **listes de tâches** (cases à cocher) ;
- ◆ le texte **barré** avec `~` ;
- ◆ les **liens automatiques** (une adresse devient cliquable sans symbole) ;
- ◆ une gestion plus souple des listes.

À RETENIR

Tout document CommonMark valide s'affiche à l'identique en GFM : GFM ne fait qu'**ajouter**, jamais retirer. En clair : apprenez le noyau commun, ajoutez les extensions GFM quand la plateforme les accepte, et vous serez à l'aise quasiment partout.

Comment savoir ce qui marche où ?

Règle simple : les éléments du noyau (titres, paragraphes, gras, italique, listes, liens, images, citations, code) fonctionnent **partout**. Les extensions (tableaux, barré, tâches) fonctionnent sur les

plateformes modernes mais pas toutes. En cas de doute, faites un test rapide sur la plateforme visée. Pour un contenu qui doit voyager partout, restez sur le noyau.

Bonnes pratiques

La syntaxe ne fait pas tout. Voici les habitudes qui distinguent un document Markdown propre, durable et agréable à relire.

- ◆ **Aérez avec des lignes vides.** Une ligne vide avant et après les titres, les listes, les blocs de code et les tableaux évite la plupart des bugs de rendu. En cas de doute, ajoutez une ligne vide.
- ◆ **Soyez cohérent.** Choisissez un style et tenez-vous-y : toujours `-` pour les puces, toujours `**` pour le gras, toujours `#` pour les titres. Un document uniforme est plus facile à lire et à modifier.
- ◆ **Respectez la hiérarchie des titres.** Un seul titre de niveau 1 par document, et pas de saut de niveau. Vos sommaires automatiques vous remercieront.
- ◆ **Écrivez de vrais textes alternatifs.** Pour chaque image, décrivez ce qu'elle montre. C'est essentiel pour l'accessibilité.
- ◆ **Une phrase par ligne (optionnel).** Pour les documents suivis dans un outil de versions, écrire une phrase par ligne facilite le repérage des modifications. Souvenez-vous : un simple retour à la ligne ne casse pas le paragraphe au rendu.
- ◆ **Prévisualisez.** Vérifiez régulièrement le rendu pendant l'écriture, surtout pour les tableaux et les listes imbriquées.
- ◆ **Gardez le HTML pour les exceptions.** Son usage occasionnel est utile ; en abuser fait perdre la lisibilité et la portabilité qui font tout l'intérêt du Markdown.

ASTUCE

La plus grande qualité d'un fichier Markdown se vérifie ainsi : ouvrez-le en texte brut, sans rendu. S'il reste clair et agréable à lire tel quel, c'est gagné.

Aide-mémoire complet

Toute la syntaxe du cours, rassemblée sur une page à garder sous la main. Les éléments marqués (ext.) sont des extensions qui peuvent ne pas être disponibles partout.

Élément	Vous écrivez
Titre (niveaux 1 à 6)	# Titre ## Sous-titre ### ...
Gras	**texte** ou __texte__
Italique	<i>*texte*</i> ou <i>_texte_</i>
Gras + italique	<i>***texte***</i>
Barré (ext.)	~texte~
Liste à puces	- élément
Liste numérotée	1. élément
Liste de tâches (ext.)	- [] à faire - [x] fait
Lien	[texte](https://adresse.fr)
Lien avec infobulle	[texte](https://adresse.fr "titre")
Image	![description](image.jpg)
Code en ligne	`code`
Bloc de code	``` ... ``` (sur 3 lignes)
Citation	> citation
Tableau (ext.)	A B puis --- ---
Ligne de séparation	---
Retour à la ligne	fin de ligne + \
Échapper un caractère	* \# _

Exercices & corrigés

La meilleure façon de retenir le Markdown est de pratiquer. Essayez d'écrire la source correspondant à chaque consigne, puis comparez avec le corrigé. Idéalement, testez vos réponses dans un éditeur en ligne.

Exercice 1 — Une fiche de présentation

Consigne : écrivez un titre principal « Mon profil ». En dessous, un paragraphe qui dit : Je m'appelle Léa et j'apprends le Markdown. (le prénom en gras, « Markdown » en italique). Puis une liste à puces de deux loisirs.

CORRIGÉ

```
# Mon profil

Je m'appelle Léa et j'apprends le Markdown.

- La randonnée
- La photographie
```

RENDU ATTENDU

Mon profil

Je m'appelle **Léa** et j'apprends le *Markdown*.

- La randonnée
- La photographie

Exercice 2 — Une recette numérotée

Consigne : écrivez un sous-titre (niveau 2) « Pâtes à l'ail ». Puis une liste numérotée de trois étapes. Dans la première étape, mettez le mot « bouillir » en gras.

CORRIGÉ

```
## Pâtes à l'ail
```

1. Faire **bouillir** l'eau
2. Ajouter les pâtes
3. Égoutter et servir

RENDU ATTENDU

Pâtes à l'ail

1. Faire **bouillir** l'eau
2. Ajouter les pâtes
3. Égoutter et servir

Exercice 3 — Un lien et une citation

Consigne : écrivez une phrase contenant un lien vers <https://commonmark.org> dont le texte cliquable est « la documentation ». En dessous, ajoutez une citation : Le Markdown, c'est écrire vite.

CORRIGÉ

```
Consultez [la documentation](https://commonmark.org).
```

```
> Le Markdown, c'est écrire vite.
```

RENDU ATTENDU

Consultez [la documentation](https://commonmark.org).

| Le Markdown, c'est écrire vite.

Exercice 4 — Pour aller plus loin

Consigne : créez un petit tableau à deux colonnes (« Tâche » et « Statut ») avec deux lignes, puis une liste de tâches reprenant les mêmes éléments, l'une cochée, l'autre non. Vérifiez le rendu sur une plateforme compatible GFM.

CORRIGÉ

```
| Tâche      | Statut  |  
|-----|-----|  
| Rédiger   | Terminé |  
| Relire    | En cours |
```

- [x] Rédiger
- [] Relire

RENDU ATTENDU

Tâche	Statut
Rédiger	Terminé
Relire	En cours

- Rédiger
- Relire

Pour aller plus loin

Vous connaissez maintenant tout le nécessaire pour écrire en Markdown au quotidien. Voici comment continuer à progresser.

S'entraîner

- ◆ Adoptez le Markdown pour vos **notes personnelles** dès aujourd'hui : c'est le meilleur entraînement, sans pression.
- ◆ Rédigez le **README** d'un de vos projets, ou une petite fiche de documentation.
- ◆ Utilisez un **éditeur à double volet** (source + rendu en direct) pour voir l'effet de chaque modification instantanément.

Notions à explorer ensuite

Quand vous serez à l'aise avec le noyau, plusieurs extensions méritent le détour selon vos besoins : les **notes de bas de page**, les **formules mathématiques** (notation LaTeX), les **diagrammes** (avec Mermaid, pris en charge par certaines plateformes), ou encore les **blocs d'avertissement** mis en avant. Toutes dépendent de la plateforme : vérifiez sa documentation.

Ressources de référence

- **commonmark.org** — la spécification officielle, avec un didacticiel interactif en dix minutes.
- La **documentation Markdown de votre plateforme** (GitHub, GitLab, Obsidian, Notion...) — pour connaître précisément les extensions disponibles.
- Un **éditeur en ligne** (dillinger.io, stackedit.io) — pour tester sans rien installer.

LE MOT DE LA FIN

Le Markdown s'apprend en une heure et se maîtrise en l'utilisant. Sa force tient dans une idée simple : un texte qui reste lisible par un humain, tout en étant transformable par une machine. Vous tenez désormais cette compétence. À vous d'écrire !