

Rédiger ses cours en Markdown .md

Sommaire

- Rédiger ses cours en Markdown .md
 - Sommaire
 - Préambule
 - Utiliser Markdown
 - Titres
 - Mise en surbrillance
 - Listes
 - Liens
 - Espacer
 - Images
 - Tableaux
 - Code
 - Diagrammes
 - Rédiger des formules mathématiques
 - Introduction
 - Alphabet grec
 - Flèches
 - Exposants et indices
 - Fonctions
 - Texte
 - Ecriture
 - Matrices
 - Alignement
 - Displaystyle
 - Personnaliser ses fichiers
 - Table des matières (VS Code)
 - Feuilles de style
 - Exporter ses fichiers (VS Code)
 - Annexe
 - Feuille de style de ce document

Préambule

Markdown est un langage de balisage léger. Comme tout langage, il est possible de le rédiger dans un éditeur de texte, mais il est préférable d'utiliser un éditeur de texte enrichi :

- Gratuits :
 - [Obsidian](#)
 - [StackEdit](#) (en ligne)
 - [Visual Studio Code](#) avec l'extension [Markdown All in One](#)

- Payant :
 - [Typora](#)

Etant un utilisateur de **VS Code**, ce document est plus axé sur ce dernier et les exemples qui seront donnés seront ceux que j'ai dû utiliser en **3A ICy**.

Même si le langage Markdown a sa propre syntaxe, il est possible d'ajouter des balises HTML pour personnaliser le document.

Utiliser Markdown

Titres

```
# Titre de niveau 1

## Titre de niveau 2

...
```

Titre de niveau 1 : "Rédiger son cours en Markdown `.md`"

Titre de niveau 2 : "Les bases"

Mise en surbrillance

```
*italique* ou _italique_

**gras** ou __gras__

***italique et gras*** ou __italique et gras__

> citation
```

italique, **gras** et ***italique et gras***

```
citation
```

Listes

- élément 1
 - élément 1.1

* élément 2

* élément 2.1

1. élément 3

1. élément 3.1

- élément 1
 - élément 1.1
- élément 2
 - élément 2.1

1. élément 3

1. élément 3.1

Liens

[Google](<https://www.google.com/>)

[Les bases](<#les-bases>)

[Google](#)

[Les bases](#)

Espacer

 : saut de ligne

--- : ligne horizontale

Images

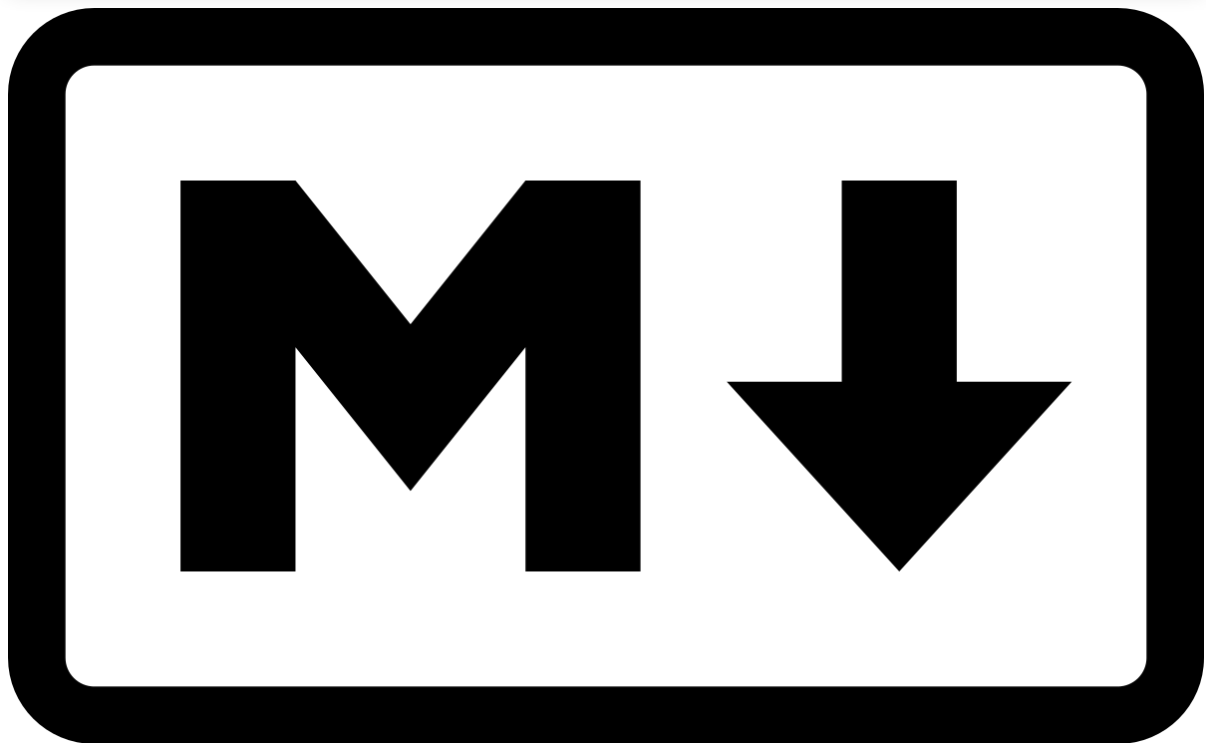
Taille fixe :

![logo-markdown](<images/markdown.png>)

Taille modifiable :

```

```



Tableaux

```
| Colonne 1 | Colonne 2 | Colonne 3 |  
| :- | :-: | -: |  
| gauche | centré | droite |
```

Colonne 1	Colonne 2	Colonne 3
gauche	centré	droite

Code

```
`print("Eurêka !")`
```

En ligne : `print("Eurêka !")`

```
...  
  
print("Eurêka !")  
...
```

En bloc :

```
print("Eurêka !")
```

Nous pouvons préciser le langage pour avoir une coloration syntaxique. Voici quelques exemples supportés par VS Code :

- Python : `python` ou `py`

```
print("Eurêka !")
```

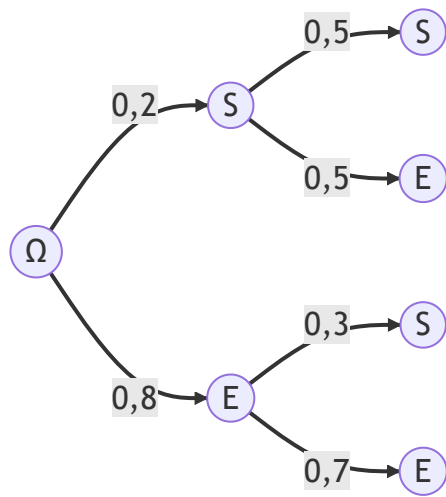
- C : `c`
- Java : `java`
- Lisp : `lisp` (racket n'est pas supporté)
- Dart : `dart`
- SQL : `sql`
- Mermaid : `mermaid` (pour les [diagrammes](#))

Diagrammes

Mermaid mériterait un document à lui tout seul, on ne traite ici que deux cas très basiques. Plus d'informations ici : [Documentation mermaid](#)

- Organigramme (utile pour arbres de probabilités) :

```
```mermaid
graph LR
A((A)) -- 0,2 --> B1((B1))
A -- 0,8 --> B2((B2))
B1 -- 0,5 --> C1((C1))
B1 -- 0,5 --> C2((C2))
B2 -- 0,3 --> C3((C3))
B2 -- 0,7 --> C4((C4))
```
```



- Diagramme de classes :

```

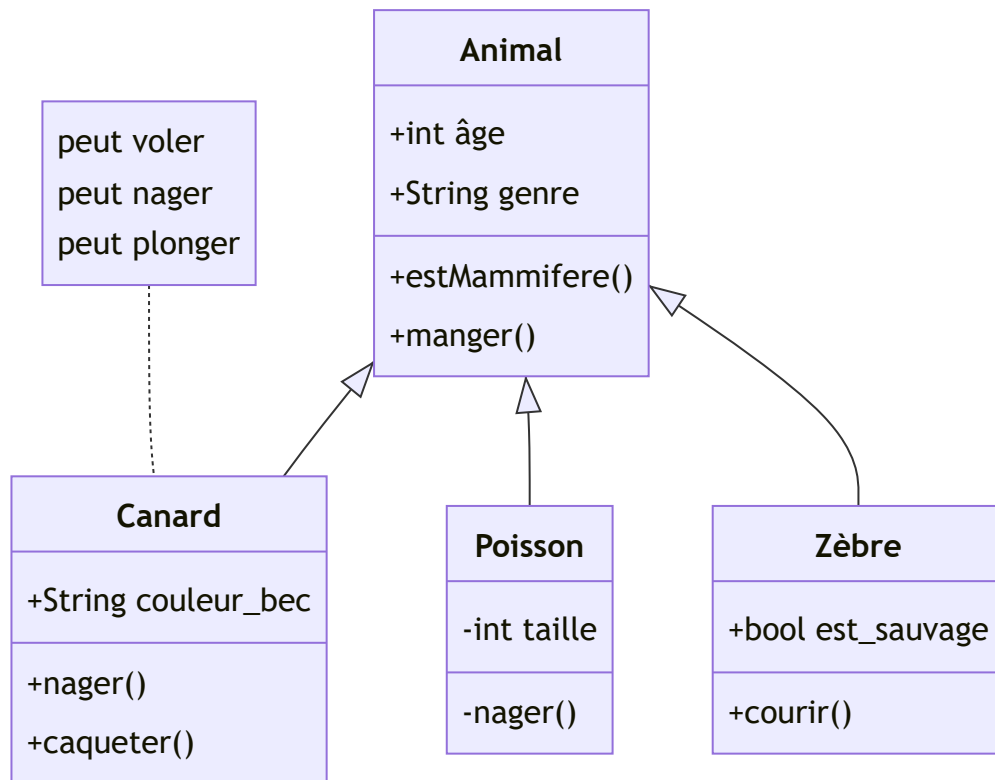
```mermaid

title: Les animaux

classDiagram
 Animal <|-- Canard
 note for Canard "peut voler\npeut nager\npeut plonger"
 Animal <|-- Poisson
 Animal <|-- Zèbre
 Animal : +int âge
 Animal : +String genre
 Animal: +estMammifere()
 Animal: +manger()
 class Canard{
 +String couleur_bec
 +nager()
 +caqueter()
 }
 class Poisson{
 -int taille
 -nager()
 }
 class Zèbre{
 +bool est_sauvage
 +courir()
 }
```

```

Les animaux



Rédiger des formules mathématiques

Cette fiche n'est pas exhaustive, pour plus d'informations : [LaTeX cheatsheet](#)

Introduction

Pour insérer des formules mathématiques, il est possible d'utiliser du code **LaTeX**.

En **LaTeX**, pour écrire une formule mathématique, il faut l'entourer de deux symboles : `$` pour une formule en ligne et `$$` pour une formule centrée.

Les symboles mathématiques sont définis par des commandes. Pour les utiliser, il faut commencer par un antislash `\` suivi du nom de la commande.

- En ligne, on peut avoir des éléments à côté de la formule :

`$y = x + 1$`

En ligne : $y = x + 1$

- Centré, on ne peut pas avoir d'éléments à côté de la formule :

`$$y = x + 1$$`

Centré :

$$y = x + 1$$

Alphabet grec

Nous pouvons faire toutes les lettres de l'alphabet grec (majuscules en minuscules) :

| \LaTeX | Résultat |
|-----------------|----------|
| \alpha | α |
| \Alpha | A |
| \omega | ω |
| \Omega | Ω |
| ... | ... |

Flèches

| \LaTeX | Résultat |
|--|---------------------------------|
| \rightarrow | \rightarrow |
| \longrightarrow | \longrightarrow |
| \Rightarrow | \Rightarrow |
| \Longrightarrow | \Longrightarrow |
| \Leftrightarrow | \Leftrightarrow |
| \iff | \iff |
| \mapsto | \mapsto |
| \longmapsto | \longmapsto |
| \rightsquigarrow | \rightsquigarrow |
| $\text{\xrightarrow[dessous]{dessus}}$ | $\xrightarrow[dessous]{dessus}$ |

Exposants et indices

Les exposants et indices sont définis par $\text{\^{\{exposant\}}}$ et $\text{_{\{indice\}}}$

| \LaTeX | Résultat |
|------------------|----------|
| \x^2 | x^2 |
| \x_2 | x_2 |
| $\text{\x^{2n}}$ | x^{2n} |
| $\text{\x_{2n}}$ | x_{2n} |

Fonctions

Les fonctions sont définies par `\nomFonction{paramètre 1}{paramètre 2}...`

| \LaTeX | Résultat |
|--------------------------|---------------|
| <code>\sin(x)</code> | $\sin(x)$ |
| <code>\cos(x)</code> | $\cos(x)$ |
| <code>\tan(x)</code> | $\tan(x)$ |
| <code>\arcsin(x)</code> | $\arcsin(x)$ |
| <code>\arccos(x)</code> | $\arccos(x)$ |
| <code>\arctan(x)</code> | $\arctan(x)$ |
| <code>\ln(x)</code> | $\ln(x)$ |
| <code>\log(x)</code> | $\log(x)$ |
| <code>\exp(x)</code> | $\exp(x)$ |
| <code>\sqrt{x}</code> | \sqrt{x} |
| <code>\sqrt[n]{x}</code> | $\sqrt[n]{x}$ |
| <code>\frac{x}{y}</code> | $\frac{x}{y}$ |

Texte

Nous pouvons écrire du texte, et le mettre en gras ou en italique.

| \LaTeX | Résultat |
|---|--------------------------|
| <code>\text{texte}</code> | texte |
| <code>\textbf{texte en gras}</code> | texte en gras |
| <code>\textit{texte en italique}</code> | <i>texte en italique</i> |

Ecriture

Nous avons façons d'écrire des lettres.

| \LaTeX | Résultat |
|---------------------------|----------------|
| <code>\mathbb{R}</code> | \mathbb{R} |
| <code>\mathbf{R}</code> | R |
| <code>\mathcal{R}</code> | \mathcal{R} |
| <code>\mathfrak{R}</code> | \mathfrak{R} |
| <code>\mathit{R}</code> | <i>R</i> |

| L ^A T _E X | Résultat |
|---------------------------------|---------------|
| <code>\mathscr{R}</code> | \mathscr{R} |
| <code>\mathtt{R}</code> | \mathtt{R} |

Cas particuliers :

| L ^A T _E X | Résultat |
|---------------------------------|--------------|
| <code>\R</code> | \mathbb{R} |
| <code>\N</code> | \mathbb{N} |
| <code>\Z</code> | \mathbb{Z} |

Matrices

```


\left(\begin{matrix}
a & b & c \\
d & e & f \\
g & h & i
\end{matrix}\right)


```

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

Nous pouvons personnaliser l'élément à gauche et à droite de la matrice.

```


\left[\begin{matrix}
a & b & c \\
d & e & f \\
g & h & i
\end{matrix}\right] \text{ ou }
\left.\begin{matrix}
a & b & c \\
d & e & f \\
g & h & i
\end{matrix}\right|


```

$$\left[\begin{matrix} a & b & c \\ d & e & f \\ g & h & i \end{matrix} \right] \text{ ou } \left. \begin{matrix} a & b & c \\ d & e & f \\ g & h & i \end{matrix} \right|$$

Nous pouvons utiliser `\pmatrix` pour avoir directement des parenthèses.

```


$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$


```

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

Systemes d'équations

```


$$\begin{cases} x + y = 1 \\ x - y = 2 \end{cases}$$


```

$$\begin{cases} x + y = 1 \\ x - y = 2 \end{cases}$$

Alignement

Nous pouvons aligner les équations grâce à `align`, ça s'alignera sur le signe `&`.

```


$$\begin{aligned} f &: \mathbb{R} \longrightarrow \mathbb{R} \\ x &\longmapsto x^2 \end{aligned}$$


```

$$\begin{aligned} f &: \mathbb{R} \longrightarrow \mathbb{R} & (1) \\ x &\longmapsto x^2 & (2) \end{aligned}$$

Nous pouvons enlever les nombres à droite grâce avec `align*`.

```


$$\begin{aligned*} f &: \mathbb{R} \longrightarrow \mathbb{R} \\ x &\longmapsto x^2 \end{aligned*}$$


```

$$\begin{aligned*} f &: \mathbb{R} \longrightarrow \mathbb{R} \\ x &\longmapsto x^2 \end{aligned*}$$

Displaystyle

Nous pouvons rendre les équations en ligne plus agréables grâce à `displaystyle`.

```
\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}
```

$$\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$$

```
\displaystyle\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}
```

$$\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$$

Personnaliser ses fichiers

Table des matières (VS Code)

Nous pouvons créer une table des matières en faisant `Ctrl+Shift+P` puis en tapant `Markdown All in One: Create Table of Contents`.

Feuilles de style

Comme en html, nous pouvons importer des fichiers css pour personnaliser nos fichiers grâce à la commande `<link href="NomDuFichier.css" rel="stylesheet"></link>`

| élément | description |
|---|-----------------------|
| <code><h1></code> , <code><h2></code> , ... | Titre 1, Titre 2, ... |
| <code><body></code> | Corps du document |
| <code></code> | Liste |
| <code></code> | Élément de liste |
| <code><table></code> | Tableau |
| <code><pre></code> | Préformaté (code) |
| <code><code></code> | Code |
| <code><blockquote></code> | Citation |
| <code><hr></code> | Ligne horizontale |
| <code></code> | Image |
| <code><a></code> | Lien |

(Voir [annexe](#) pour la feuille de style utilisée dans ce document.)

Exporter ses fichiers (VS Code)

Pour exporter un fichier en html, il suffit de faire **Ctrl+Shift+P** puis de taper **Markdown All In One : Print current document to HTML**.

Il existe des extensions pour exporter en **pdf** mais celles-ci ne sont pas fiables à 100%. Personnellement, j'exporte mon fichier en html puis l'ouvre dans un navigateur pour l'enregistrer en pdf (impression).

Annexe

Feuille de style de ce document

```
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@400;600;700&display=swap');

:root {
  /* === Couleur === */
  --color: 210; /* 210 = bleu, 100 = vert, 270 = violet */

  /* Variations */
  --base: hsl(var(--color), 30%, 50%);
  --h1: hsl(var(--color), 30%, 20%);
  --h2: hsl(var(--color), 30%, 30%);
  --h3: hsl(var(--color), 30%, 40%);
  --h4: hsl(var(--color), 30%, 50%);
  --h5: hsl(var(--color), 30%, 60%);
  --code: hsl(var(--color), 30%, 30%);
  --light-color: hsl(var(--color), 30%, 90%);
  --lighter-color: hsl(var(--color), 30%, 95%);

  /* === Police === */
  --body-font: 'Montserrat', sans-serif;

  --h1-size: 2rem;
  --h2-size: 1.5rem;
  --h3-size: 1.3rem;
  --h4-size: 1.1rem;
  --h5-size: .9rem;
}
```

```
body {
    font-family: var(--body-font);
}

h1 {
    text-align: center;
    color: var(--h1);
    font-size: var(--h1-size);
    border-bottom: 2px solid;
}

h2 {
    color: var(--h2);
    font-size: var(--h2-size);
}

h3 {
    color: var(--h3);
    font-size: var(--h3-size);
}

h4 {
    color: var(--h4);
    font-size: var(--h4-size);
}

h5 {
    color: var(--h5);
    font-size: var(--h5-size);
}

h1, h2, h3, h4, h5 {
    font-weight: 600;
}

strong {
    color: var(--base);
}

em {
    color: var(--base);
}

a:link,
a:visited,
a:active {
```

```
    color: var(--base);
    text-decoration: underline;
}

a:hover {
    font-weight: bold;
}

.toc a:link,
.toc a:visited,
.toc a:active {
    text-decoration: none;
}

code {
    background-color: var(--light-color);
    color: var(--code);
    padding: .2rem;
    border-radius: .5rem;
}

pre code {
    background-color: transparent;
    padding: 0;
    border-radius: 0;
}

pre {
    border-radius: 5px;
    padding: 1rem;
    margin: 1rem 0 1rem 0;
    box-shadow: rgba(0, 0, 0, 0.2) 0px 4px 12px;
}

blockquote {
    background-color: var(--light-color);
    color: var(--base);
}

.center {
    display: flex;
    justify-content: center;
    align-items: center;
    margin-top: 1rem 0 1rem 0;
}
```

```
table {
  margin: .75rem 0;
  padding: 0;
  border-collapse: collapse;
}

table tr {
  margin: 0;
  padding: 0;
  background-color: var(--lighter-color);
}

table tr:nth-child(2n) {
  background-color: #fff;
}

table tr th {
  margin: 0;
  padding: .35em .75em;
  border: 1px solid;
}

table tr td {
  margin: 0;
  padding: .35em .75em;
  border: 1px solid;
}

table code {
  background-color: transparent;
  padding: 0;
  border-radius: 0;
}

hr {
  border: 0;
  background-color: var(--light-color)
}
```

/ div pour la couleur des puces de la table des matières,
n'est pas générées par Markdown ALL In One */*

```
.toc {
```



```
color: var(--base);
```

```
}
```

