

[documentation](#), [python](#), [animation](#), [em](#)

## Outils de documentation

### Enregistrer des sessions de terminal avec asciinema

Asciinema permet d'enregistrer des sessions de terminal, de les envoyer automatiquement en ligne, de les enregistrer localement ou de les rejouer dans une page web avec un lecteur approprié :

- <https://asciinema.org/>
- <https://github.com/asciinema/asciinema-player>

Sur linux, il est installable depuis les dépôts `sudo apt install asciinema`

Si nécessaire, il est possible de transformer ces enregistrements en gif, en svg animé ou en vidéo

- SVG : <https://github.com/marionebi/svg-term-cli>
- GIF : <https://github.com/asciinema/agg>
- MP4 : <https://github.com/lhr0909/asciinema-mp4>

### Transformer des liens en liens cliquables

Objectif : créer un fichier pdf avec des liens cliquables à partir de la liste de liens sous forme de fichier texte.

- installer linkificator pour firefox ( <https://addons.mozilla.org/fr/firefox/addon/linkificator/> )
- ouvrir le fichier texte dans firefox
- “imprimer” dans un fichier pdf

### Enregistrer des vidéos de l'écran

Voir [SimpleScreenRecorder](#)

### Réaliser des animations

#### sganimator

Animation à partir de dessins vectoriels au format svg, exportables sous forme de suite d'image, par exemple pour être assemblées en vidéo ou en gif.

<https://github.com/gweltou/SgAnimator>

#### manim

Adapté aux animations mathématiques. Les animations sont créées par des scripts python et exportées en vidéo mp4

- exemples : <https://docs.manim.community/en/stable/examples.html>
- <https://github.com/ManimCommunity/manim/>
- tuto : <https://github.com/malhotra5/Manim-Tutorial>
- <https://github.com/helblazer811/ManimML> adaptation pour présenter des concepts de machine learning

[sinecurveunitcircle-1.mp4](#)

## Visualiser des données

voir aussi [Data, analyse et visualisation](#)

### ObservableHQ

Carnet de notes permettant d'inclure différents types de cellules (calcul, visualisation) : <https://observablehq.com/>

### rawgraphs.io

Application en ligne avec de nombreux types de visualisation : <https://app.rawgraphs.io/>

## Cartographie

voir aussi [cartographie avec python](#)

### umap

Outils basé sur openstreetmap pour créer des cartes interactives (avec marqueurs, polygones, bulles au contenu multimedia, fonds de carte personnalisables, etc.)

Les cartes peuvent être intégrées à une page web ou sont directement accessibles par une URL.

<https://umap.openstreetmap.fr/fr/>

### leaflet.js

Bibliothèque javascript, à réserver aux codeu/r/se/s

<https://leafletjs.com/>

## Rédaction de documentation pour diffusion sur le web

Plusieurs possibilités, et il y en a sûrement d'autres...

- **flatdoc** (markdown, multi-colonnes, transformation du md à la volée par javascript)
  - <https://github.com/rstacruz/flatdoc>
  - <https://ricostacruz.com/flatdoc/>
  - <https://tommcfarlin.com/documentation-with-markdown/>
- **SGDoc** markdown → HTML, une fois pour toutes : <https://github.com/saturngod/SGDoc>
- **daux** PHP sur serveur, depuis CommonMark, une variante de markdown
  - <https://github.com/dauxio/daux.io>
  - [http://daux.io/Getting\\_Started.html](http://daux.io/Getting_Started.html)
- Jekyll + **just the docs** : <https://github.com/just-the-docs/just-the-docs>

Il existe aussi des thèmes intégrables dans des CMS :

- pour **Grav** : <https://github.com/getgrav/grav-theme-learn2>

## Wiki(s)

Une page ressource très complète sur dokuwiki : <https://dvillers.umons.ac.be/wiki/floss:dokuwiki>

## LateX

Des ressources : <https://dvillers.umons.ac.be/wiki/floss:latex>

## Écrire en markdown

Des éditeurs en ligne («in browser») :

- <https://stackedit.io>
- <https://dillinger.io/>
- <https://pandao.github.io/editor.md/en.html>

Avec Atom et le plugin markdown preview : <https://atom.io/packages/markdown-preview>  
Cheat sheets :

- <https://www.markdownguide.org/cheat-sheet/>
- <https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet>

## markdown à tout faire

Présentations / slides avec **marp**:

- <https://marp.app/>
- <https://github.com/marp-team/awesome-marp>
- exemple (intéressant aussi pour son sujet!) :  
<https://eyssette.forge.aeif.fr/marp-slides/slides/2022-2023/utiliser-le-markdown-pour-tout-faire#1>

Graphes / diagrammes / visualisation de données avec **mermaid** : <https://mermaid.js.org/intro/>

On peut écrire du code mermaid dans codium (ou vscode) grâce au plugin *markdown extended*

Ensuite on peut en faire le rendu dans l'éditeur web : <https://mermaid-js.github.io/mermaid-live-editor>

Il existe d'autres éditeurs web mais c'est celui-ci qui m'a donné les meilleurs résultats d'interprétation et le svg le plus léger

Cartes mentales avec **markmap**

- <https://markmap.js.org/docs/markmap#what-is-markmap>
- exemple / fichier html complet : <https://stackblitz.com/edit/markmap-autoloader?file=index.html>
- exemple / application en ligne : <https://mymarkmap.netlify.app/>

Avec codimd : <https://codimd.apps.education.fr/KsjwlfUJT8S8X0gs3IsMIA?both#>

## Conversions de format texte

Passer de dokuwiki à markdown à mediawiki à pdf, etc.

→ avec Pandoc, installable ou en ligne : <https://pandoc.org/try/>

## Mise en page de code

Produire un pdf avec coloration syntaxique, en ligne : <https://extensions.libreoffice.org/en/extensions/show/code-highlighter>

Il existe aussi une extension pour libreoffice : <https://nsspot.herokuapp.com/code2pdf/> (pour java : xcode sur fond blanc très bien)

## Typo ASCII / Dessin ASCII

Pour de grandes lettres composées de caractère ASCII : <http://www.jave.de/> avec aussi des fonctions de dessin

Article extrait de : <https://lesporteslogiques.net/wiki/> - **WIKI Les Portes Logiques**  
Adresse : [https://lesporteslogiques.net/wiki/ressource/logiciel/outils\\_documentation](https://lesporteslogiques.net/wiki/ressource/logiciel/outils_documentation)  
Article mis à jour: **2024/07/04 19:02**