

Markdown avec VS Code

Lien : code.visualstudio.com

Installation

Installer l'extension suivante (`Ctrl+Shift+X`) : **Markdown Extension Pack** qui fournira :

- [Markdown All in One](#) : raccourcis claviers, ...
- [Markdown Preview Enhanced](#) : vue Aperçu
- [Markdown PDF](#) : conversion au format PDF
- [markdownlint](#) : vérification syntaxique
- ...

Utilisation

VS Code prend en charge automatiquement les fichiers **Markdown** qui ont l'extension `.md`.

Il est possible de basculer entre l'éditeur et l'aperçu : `Ctrl+Maj+V` ou côte à côte avec `Ctrl+K V`.

On peut aussi utiliser *Commande Palette* (`Ctrl+Shift+P`) :

`Markdown: Open Preview to the Side`.

On peut voir et naviguer dans la structure du document Markdown avec la vue *Outline* dans l'*Explorer*.

Pour convertir son fichier, il faut :

- ouvrir *Commande Palette* (`Ctrl+Shift+P`) puis
`Markdown PDF: Export (pdf)` OU `Markdown PDF: Export (html)`
- ou un **clic droit** dans le document ouvert
- ou un **clic droit** dans la vue Aperçu (*Preview*).

Remarque : il existe une extension `vscode-pandoc` pour convertir dans beaucoup d'autres formats.

Configuration

Pour configurer les extensions Markdown, il faut aller dans Préférences → Settings `Ctrl+,` puis Extensions → Markdown ...

Il est possible d'éditer ses options avec : *Edit in settings.json* .

Quelques options à regarder pour la personnalisation, notamment par feuille de style (*.css*) :

```
// Police
"markdown.preview.fontFamily": "-apple-system, BlinkMacSystemFont, 'Segoe WPC',
'Segoe UI', 'Ubuntu', 'Droid Sans', sans-serif",

// Taille
"markdown.preview.fontSize": 14,
"markdown.preview.lineHeight": 1.6,

// Thèmes personnalisés à ajouter dans la liste
"markdown.styles": [
    "markdown-pdf-remarkable.css",
],

// Thème pour l'export HTML
"markdown.extension.print.theme": "light",

// Personnalisation pour l'export PDF
"markdown-pdf.displayHeaderFooter": true,
"markdown-pdf.headerTemplate": "<div style=\"font-size: 9px; margin-left: 1cm;\">
<span class='title'></span></div> <div style=\"font-size: 9px;
margin-left: auto; margin-right: 1cm; \"> <span class='date'></span></div>",
"markdown-pdf.footerTemplate": "<div style=\"font-size: 9px; margin: 0 auto;\">
<span class='pageNumber'></span> / <span class='totalPages'></span></div>",
"markdown-pdf.includeDefaultStyles": true,

// Sinon
"markdown-pdf.includeDefaultStyles": false,
"markdown-pdf.stylesRelativePathFile": true,
"markdown-pdf.styles": [
    "markdown-pdf-remarkable.css",
],

// Pour la vue Aperçu
"markdown-preview-enhanced.codeBlockTheme": "one-light.css",
"markdown-preview-enhanced.previewTheme": "github-light.css",
```

Exemple de fichier *.css* pour l'export PDF : [markdown-pdf-remarkable.css](#)

Autres extensions

Il existe beaucoup d'autres extensions pour **Markdown** :

- [Markdown Shortcuts](#) : des raccourcis pratiques

- [Markdown Extended](#) : export dans différents formats (HTML, PDF, PNG et JPEG)
- [Excel to Markdown table](#) : copier/coller de données Excel
- [CSV to Markdown Table Converter](#) : copier/coller de données CSV
- ...

Diagrammes

Il est possible d'intégrer des diagrammes dans **Markdown** :

- [Markdown Preview Mermaid Support](#) : prise en charge des diagrammes et organigrammes [Mermaid](#)
- [PlantUML](#) : prise en charge des diagrammes UML (et d'autres) [PlantUML](#)
- [Graphviz Markdown Preview](#) : création de graphiques avec [Graphviz](#)

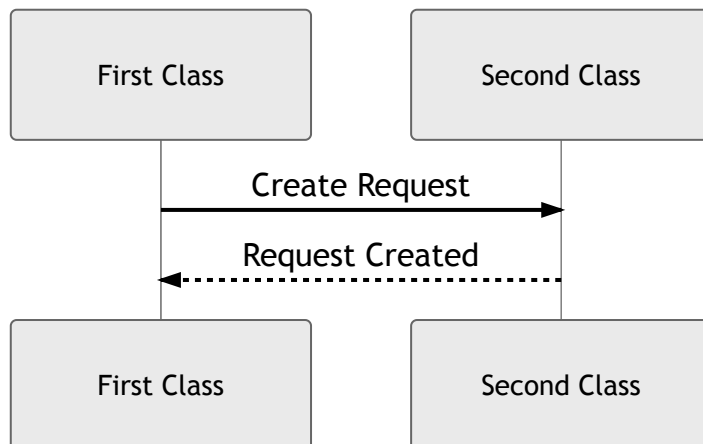
Exemple Mermaid

```

```mermaid
sequenceDiagram
 participant First Class
 participant Second Class

 First Class ->> Second Class: Create Request
 Second Class -->> First Class: Request Created
...

```



## Exemple PlantUML

```

````plantuml
@startuml
hide footbox
actor User
participant "First Class" as A
participant "Second Class" as B
participant "Last Class" as C

User -> A: DoWork
activate A

A -> B: Create Request
activate B

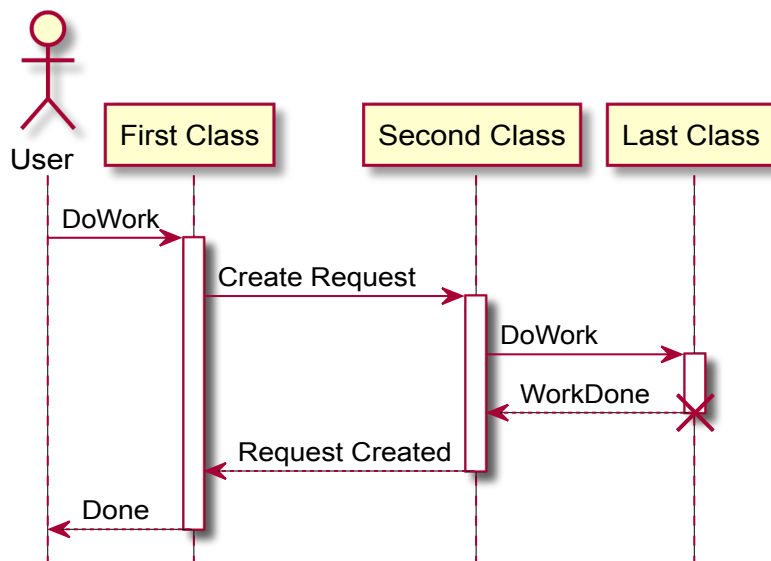
B -> C: DoWork
activate C
C --> B: WorkDone
destroy C

B --> A: Request Created
deactivate B

A --> User: Done
deactivate A

@enduml
````

```



## Exemple Graphviz

```
```graphviz
digraph example {
    node [shape=box, fontname=Helvetica, fontsize=12, color=black];
    a [ label="QObject" ];
    b [ label="Exemple" fillcolor=lightblue,style=filled ];
    a -> b [ arrowhead="normal", fillcolor=white,style=filled,dir=back ];
}
```
```

```
digraph example {
 node [shape=box, fontname=Helvetica, fontsize=12, color=black];
 a [label="QObject"];
 b [label="Exemple" fillcolor=lightblue,style=filled];
 a -> b [arrowhead="normal", fillcolor=white,style=filled,dir=back];
}
```