VNote Documentation

Release 3.10.1

Le Tan

Nov 18, 2021

User Documentation

1 Why VNote **3**
- 1.1 What is VNote . . . 3
- 1.2 Why Another Markdown Wheel . . . 3

2 Get Started **5**
- 2.1 Main Interface . . . 5
- 2.2 Ready To Go . . . 6

3 Notes Management **7**
- 3.1 Notebook . . . 7

4 Configuration **9**
- 4.1 Components of Configuration . . . 10
- 4.2 Make VNote Portable . . . 10

5 Snippet **13**
- 5.1 Snippet Management . . . 13
- 5.2 Define A Snippet . . . 15
- 5.3 Apply A Snippet . . . 15
- 5.4 Examples . . . 17

6 Template **19**

7 Themes and Styles **21**
- 7.1 Themes . . . 21
- 7.2 Samples . . . 22

8 Image Host **25**
- 8.1 Configuration . . . 26

9 Export **29**
- 9.1 General Settings . . . 30
- 9.2 Markdown . . . 30
- 9.3 HTML . . . 31
- 9.4 PDF . . . 32
- 9.5 Custom . . . 33

10 Frequently Asked Questions **35**

10.1 How to Specify Custom MathJax Script? 35
10.2 Crash After Version Update 35
10.3 Interface Is Freezed 35
10.4 Cursor Is Not Visible in Edit Mode 35
10.5 Crash When Opening A Note 35

A pleasant note-taking platform.

VNote is open source, and available on GitHub.

For latest documentation, please check VNote Home Page.

There are documents in several languages.

- English
- Japanese ()

The main documentation for the site is organized into a couple of sections:

- *User Documentation*
- dev-docs

CHAPTER 1

Why VNote

1.1 What is VNote

At the very beginning, VNote is a **V**im-inspired **Note**-taking application, designed specially for **Markdown**. It is not yet just another Markdown editor. VNote is designed to be a powerful Markdown editor with sweet note management, or a note-taking application with pleasant Markdown experience.

Now VNote is intended to be a pleasant note-taking platform, supporting more document types in the future.

VNote is free and open source. You could get VNote for Linux, Windows, and macOS.

1.2 Why Another Markdown Wheel

1.2.1 Insights About Markdown

Markdown, as a simple mark language, unlike rich text, was born with a **gap between edit and read**. There are about three ways to handle this gap:

1. As one extreme, some editors just treat Markdown as **plain text**. Users may lose themselves in the messy black characters. It is **hard to keep track** of the information of the note.
2. Most Markdown editors use two side-by-side panels to **edit and preview Markdown notes simultaneously**. It makes things easier since users could see a pleasant typesetting and layout while editing the text. However, two panels may occupy the whole screen and users keep moving eyes left and right, which will be a big distraction.
3. As another extreme, some editors convert the Markdown elements in place immediately after the typing, which makes editing Markdown just like editing rich text document in Word with some shortcuts. This may conflict with what Markdown is intended to be.

Since most editors choose the second way to handle the gap, people always think of preview when it comes to Markdown. It may be a **MISUNDERSTANDING** about Markdown. Designed as a simple mark language, Markdown is intended to help to keep track of the information of the text when editing and provide a beautiful typesetting when reading after being converted to HTML.

1.2.2 Tradeoff: VNote's Way

VNote tries to minimize the gap and provide the best-effort WYSIWYG for Markdown via **tuned syntax highlights and some other features**. By helping to keep track of the content, there is no need to preview or alter the text immediately after being typed.

CHAPTER 2

Get Started

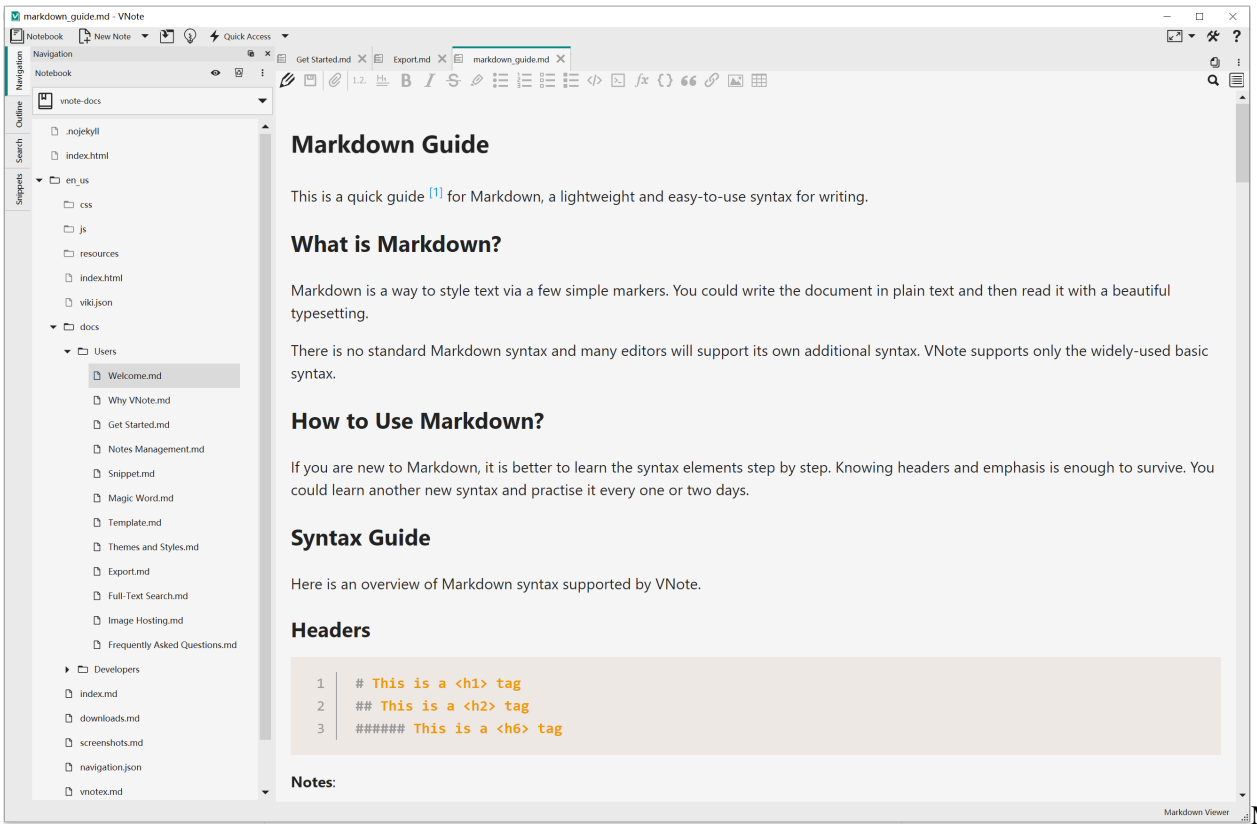
VNote is distributed on **Linux**, **Windows**, and **macOS**. You could always download standalone packages from the Downloads page.

It is recommended to download and use the **stable releases** of VNote. If you prefer to use some latest features of VNote, you could try the latest continuous build from `master` branch.

Besides the standalone packages, some package managers may already package and distribute VNote, such as *AUR*, and *homebrew cask*.

2.1 Main Interface

Here is a snapshot of VNote's main interface.



Main Interface

2.1.1 Notebooks, Folders, and Notes

VNote uses **Notebooks**, **Folders**, and **Notes** hierarchy for note management. The left-side panel is used to navigate through your notebooks, folders, and notes.

A notebook corresponds to a directory in your file system. You should choose an empty directory as your *Notebook Root Folder*. All the contents of this notebook will be stored within the Notebook Root Folder.

Once you have a notebook, you could create folders in this notebook. VNote supports infinite levels of folders.

2.1.2 Content Area

The middle area which occupies most of the space is the **Content Area**, which is used to read and edit notes. VNote supports opening multiple notes by tabs.

2.1.3 Dock Widgets

At the left side, there are several dock widgets, including **Navigation**, **Outline**, **Search** and so on. They could be dragged and dropped freely and placed at the left/top/right/bottom of the main window.

2.2 Ready To Go

That is it! Now you could read, edit, and mange your notes in VNote!

CHAPTER 3

Notes Management

VNote adopts **notebooks-folders-notes** hierarchy for notes management. A notebook corresponds to a directory in the file system, which is called **Notebook Root Folder**. Folders of a notebook correspond to directories within the Notebook Root Folder. Notes inside a folder corresponds to files within that directory.

VNote may use some index files to keep track of all the notes within a notebook, so it is recommended to manage your notes within VNote.

3.1 Notebook

Notebook is an independent, self-explanatory container in VNote. A notebook is a **Notebook Root Folder** in the file system. The root folder contains all the notes and configuration files of that notebook.

3.1.1 Create A Notebook

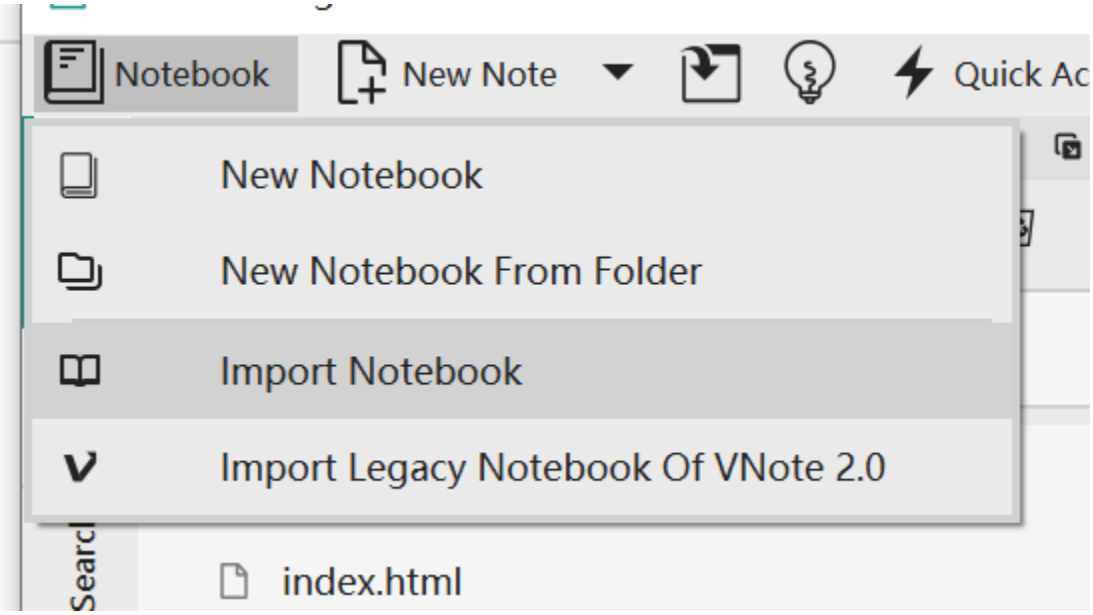
You could create a notebook directly from an empty Notebook Root Folder.

If you have already many Markdown files before using VNote, you could **New Notebook From Folder** to construct a notebook from existing folder.

3.1.2 Migrate and Import A Notebook

A notebook is an independent directory in the file system, so you could just copy or synchronize the *Notebook Root Folder* to migrate a notebook.

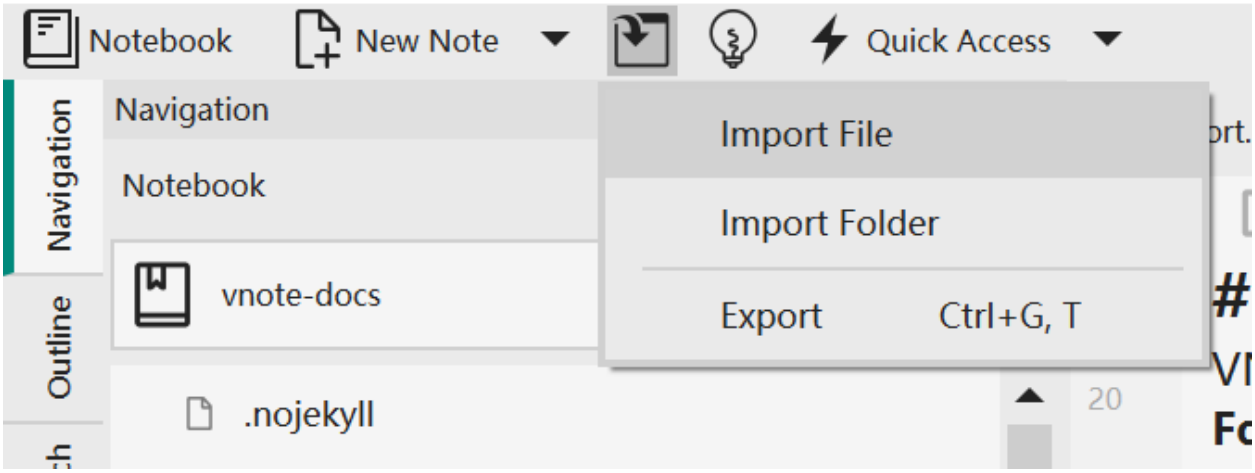
You could import an existing notebook into VNote by selecting its *Notebook Root Folder*. VNote will try to read the configuration files to restore the notebook.



Combining these, you could create your notebooks in a directory which is synchronized via third-party service, such as Dropbox and OneDrive, and then on another computer, you could import that directory into VNote as a notebook. With this, you could use VNote to edit and manage your notes, which will be synchronized by other trusted services, both at home and at work.

3.1.3 Import Files and Folders

You could import external files and folders into existing notebook.

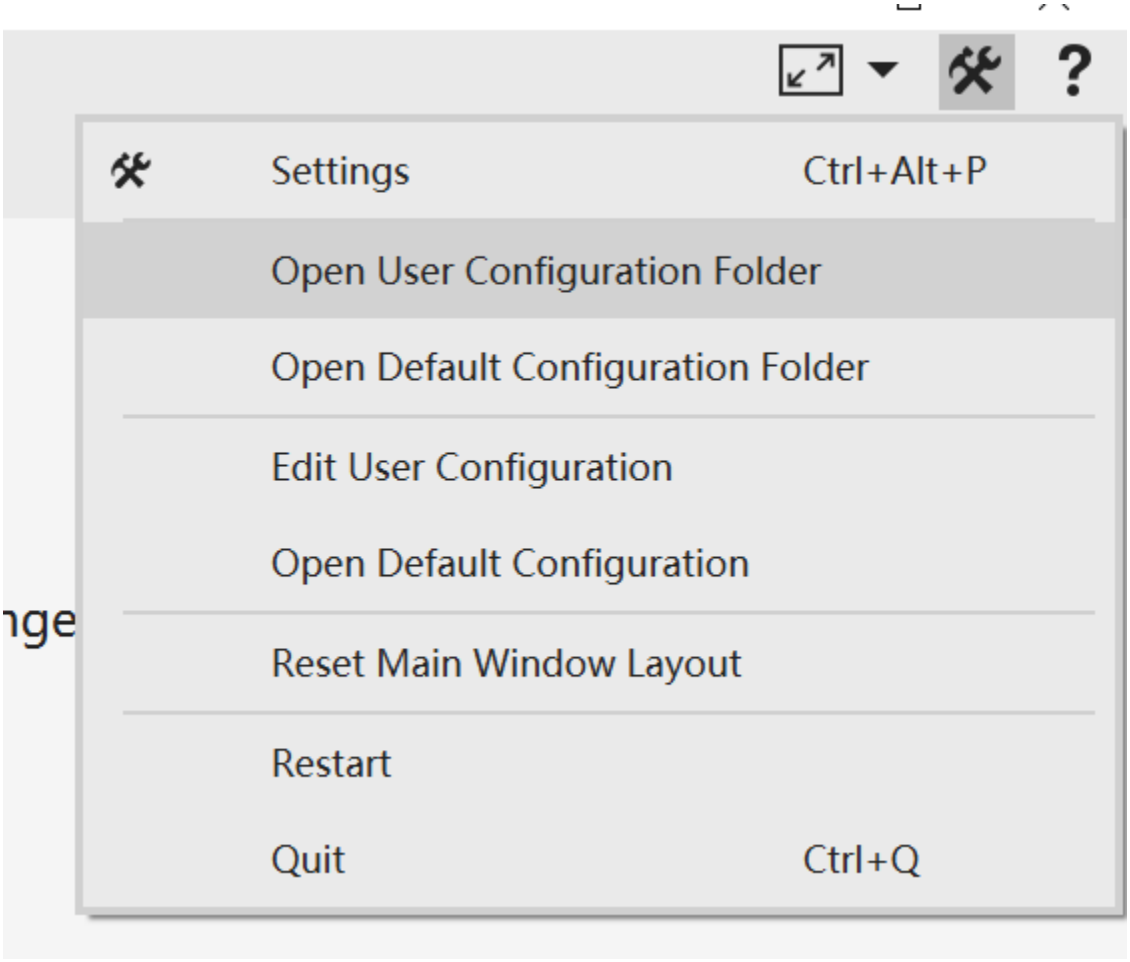


CHAPTER 4

Configuration

There are three main configurations for VNote.

- Default configuration;This is the application's default configuration. You should not change any of these files, since they will be overwritten on version update. To custom configuration, you could copy files to the **user configuration** folder and change them.
- User configuration;This is the user's configuration which will overrides the default configuration.
- Session configuration;This is configuration about session, such as notebook list, geometry state of the main window, and so on. It is the `session.json` file locating in the *user configuration* folder. To modify this file, you need to first close VNote since VNote will write to this file on close.



4.1 Components of Configuration

There are several folders and files under the configuration folder:

```
.
├── dicts (dictionaries for spellcheck)
├── docs (docs used by VNote to provide help)
├── syntax-highlighting (syntax highlighting files for text editor from Kate)
├── themes (themes of VNote)
├── web (resources used for read mode)
└── vnotex.json (configuration file)
```

For the `vnotex.json` file, please refer to the one in the default configuration folder, which contains comments about each field.

4.2 Make VNote Portable

Users may want to bundle the configuration files alongside with the executable file. Follow these steps to migrate the configuration folders:

1. Close VNote first;

2. Copy the default configuration folder `VNote` to the same folder as the executable file (such as `vnote.exe`) and rename it to `vnotex_files`;
3. Copy the user configuration folder `VNote` to the same folder as the executable file and rename it to `user_files`;

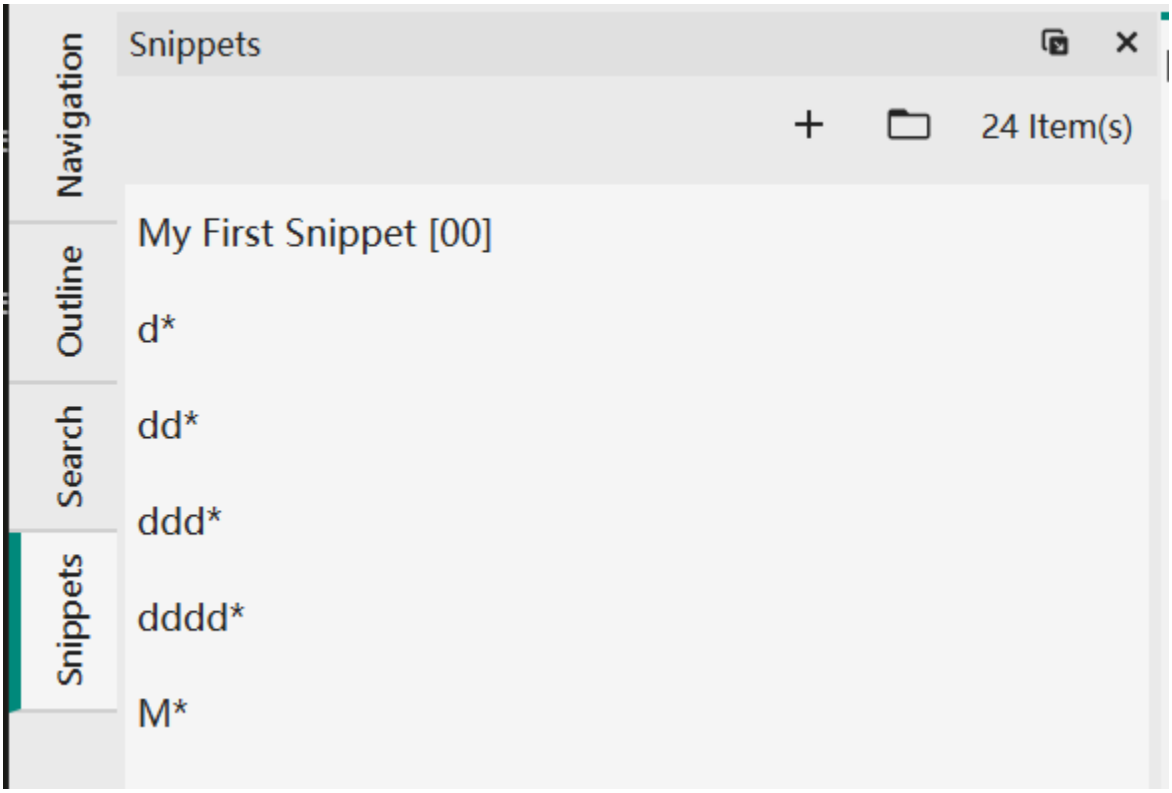
CHAPTER 5

Snippet

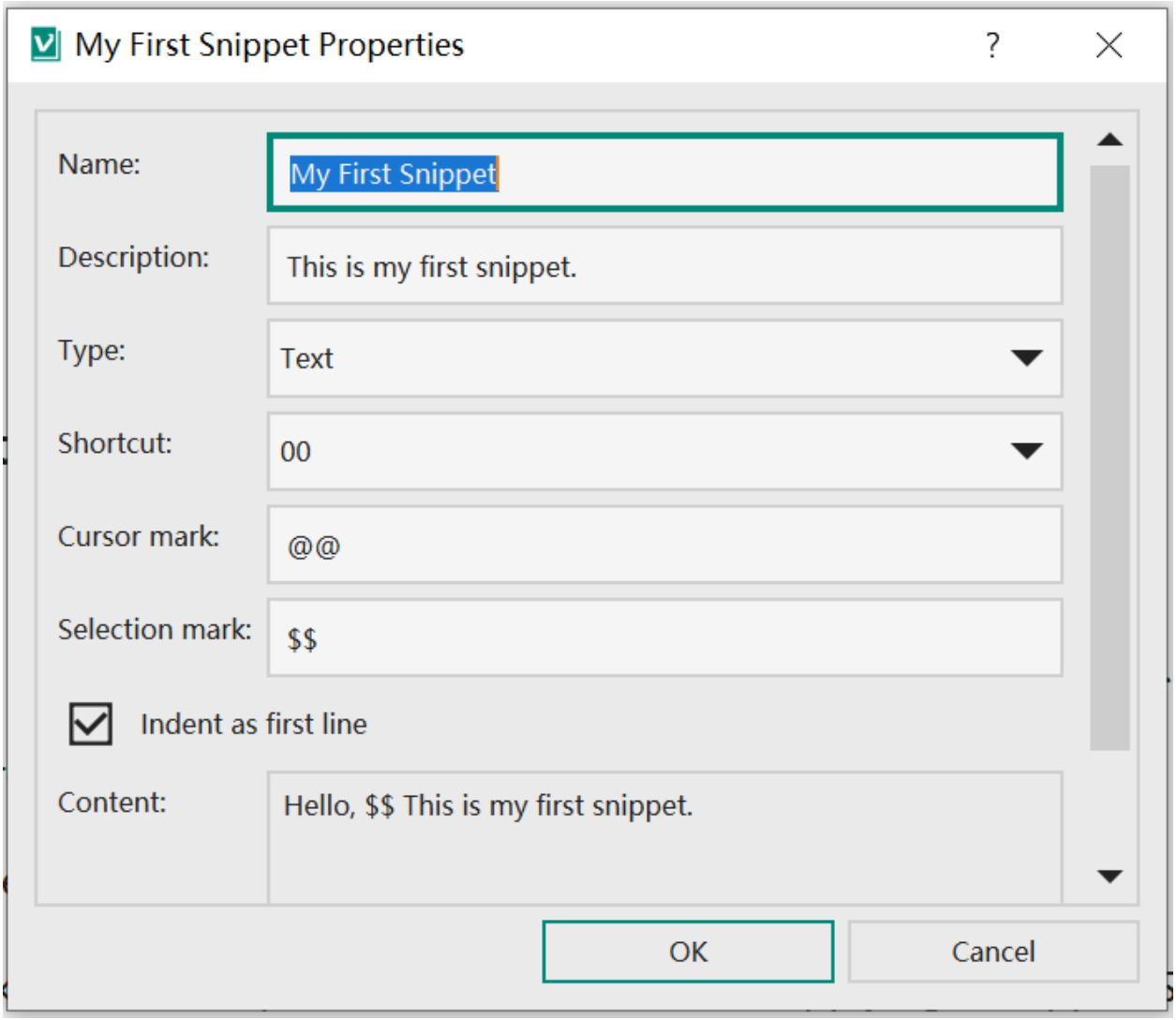
Snippet is designed to facilitate the repeated inputs of some words.

5.1 Snippet Management

There is a `json` file for a user-defined snippet. There are some built-in snippets, such as inputting current date. Built-in snippets are read-only and there is a $\star$ suffix in after their names.


Navigation
Outline
Search
Snippets
Snippets
24 Item(s)
My First Snippet [00]
d*
dd*
ddd*
dddd*
M*

5.2 Define A Snippet



- Snippet Name: The identifier of a snippet. It will be used to search for a snippet.
- Shortcut: You could assign a shortcut to s snippet to quickly locate a snippet. The shortcut is two digits.
- Cursor Mark: VNote uses Cursor Mark to mark the position of the cursor after applying a snippet. Should appears only once in the `Content`.
- Selection Mark: VNote uses Selection Mark to mark the position to insert the selected text before applying a snippet. Selection mark could appear multiple times in the `Content`. After applying a snippet, all the selection marks will be replaced with the selected text.

5.3 Apply A Snippet

5.3.1 Snippet Panel

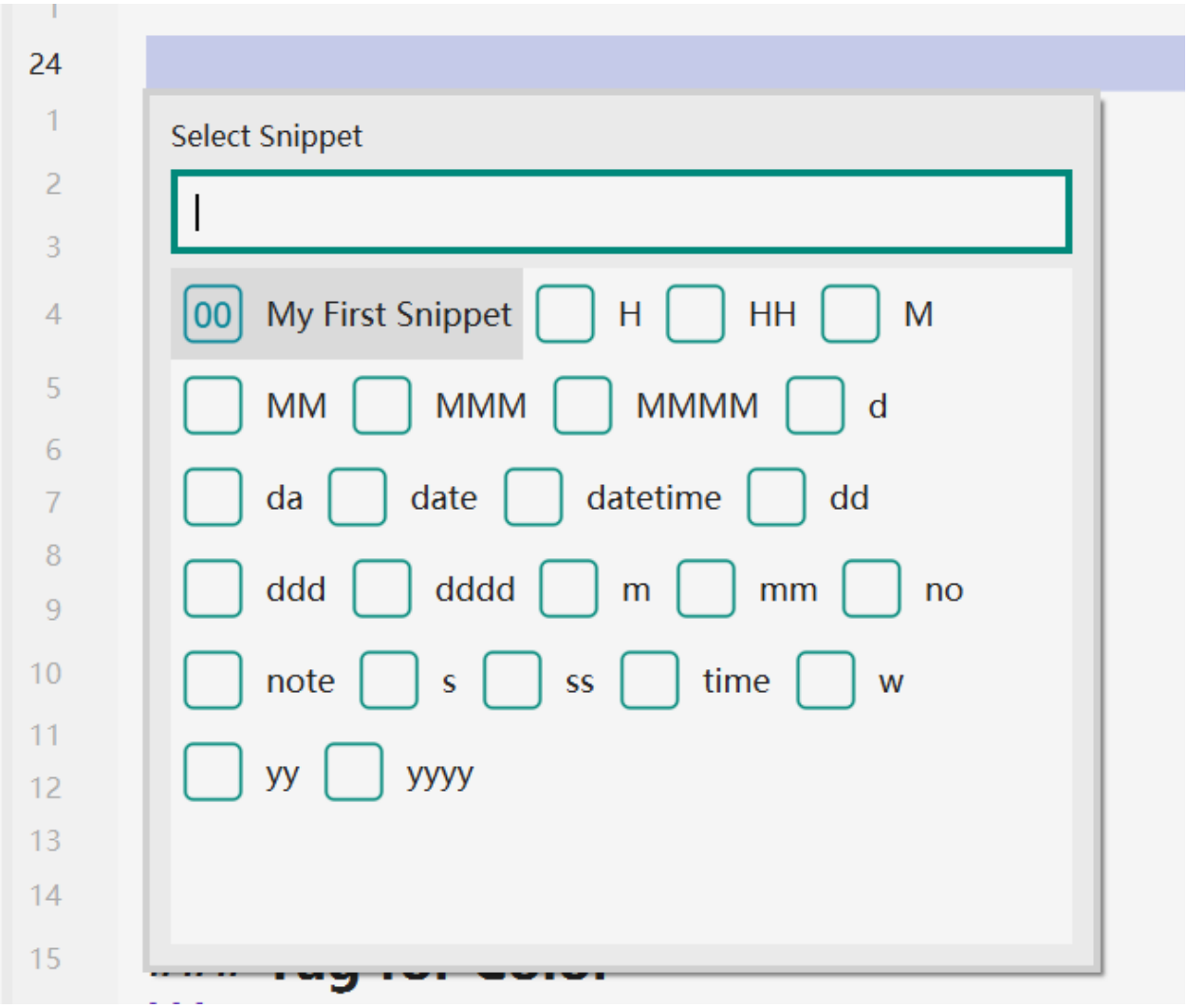
Place the cursor at the proper position and double click a snippet in the snippet panel to apply a snippet.

5.3.2 Shortcuts

In editor, press `Ctrl+G, I` will call out a panel containing all the snippets.

You could directly type `00` to apply `My First Snippet`. Or you could type keyword `my` to search snippets by name and press `Enter` to apply the first hit snippet.

You could press `Tab` to focus on the snippet list and use `Ctrl+H/J/K/L` to navigate through items and press `Enter` to apply the selected snippet.



5.3.3 Symbol

Another simple way to apply a snippet is type `%snippet_name%` in the editor directly and press `Ctrl+G, I` to translate that symbol into snippet `snippet_name`.

Many line edit widgets in VNote also support snippet in this way, such as:

- New Note dialog;
- New Notebook dialog;
- Note template;

5.4 Examples

5.4.1 Insert Code Block with CPP

```
```cpp
@@
```
```

5.4.2 Comment Out Selected Text

```
<!-- $$@@ -->
```

5.4.3 Tag for Color

```
<font color=red>$$@@</font>
```

Or:

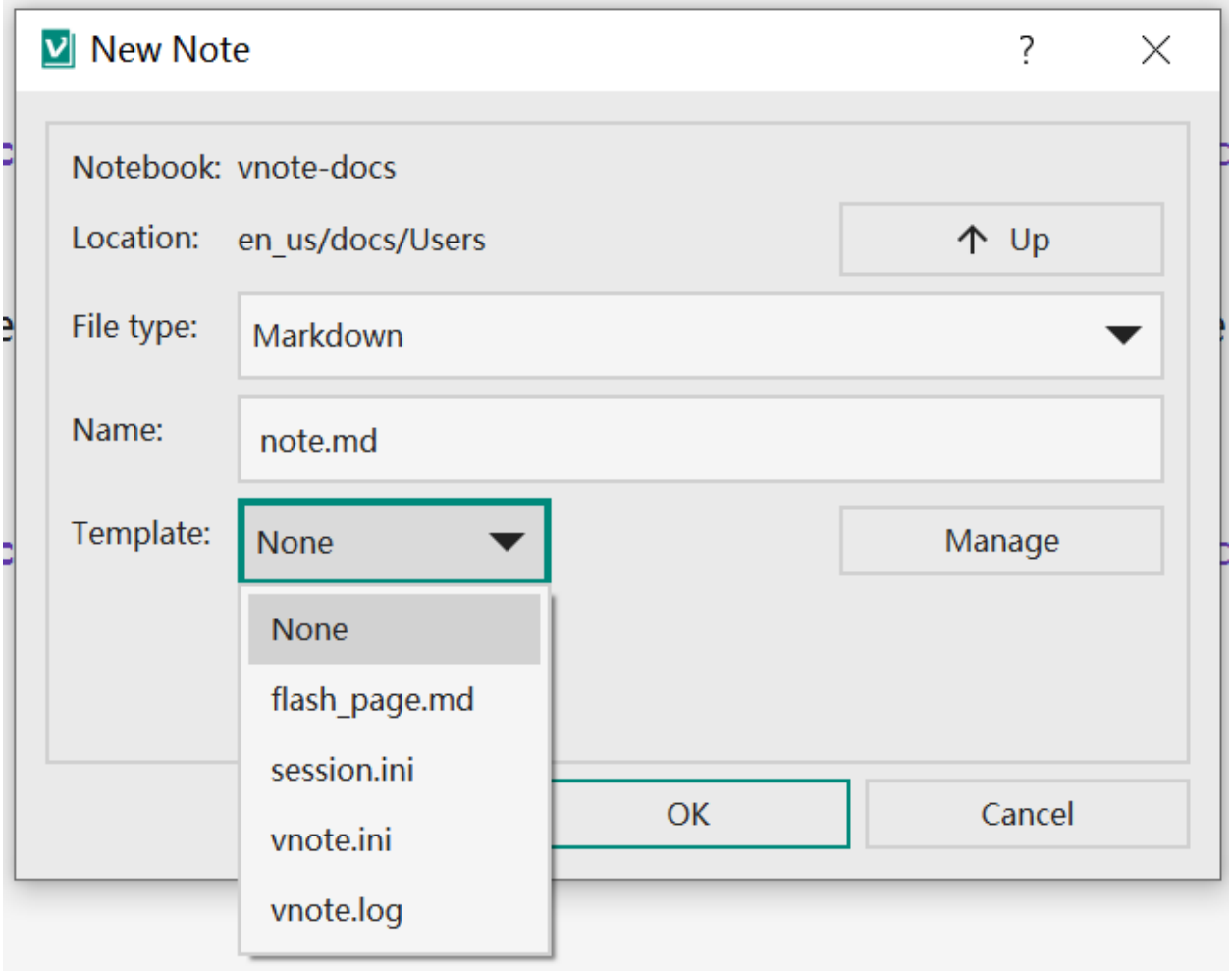
```
<mark>$$@@</mark>
```

CHAPTER 6

Template

VNote supports creating a note from a template.

When creating a note, you could choose one template in the dialog.



VNote stores template files in the `templates` folder. One file corresponds to one template.

You could add or delete template files directly in the template folder via system's file browser.

VNote supports **Snippet** in template. For example, you could write a template like this:

```
# %no%
This is a template using **Snippet** to insert note name as the title automatically.
```

`%no%` is a built-in snippet which will be evaluated to the note name (without suffix). Hence if the note name is `week report.md`, then the new note will look like this:

```
# week report
This is a template using **Snippet** to insert note name as the title automatically.
```

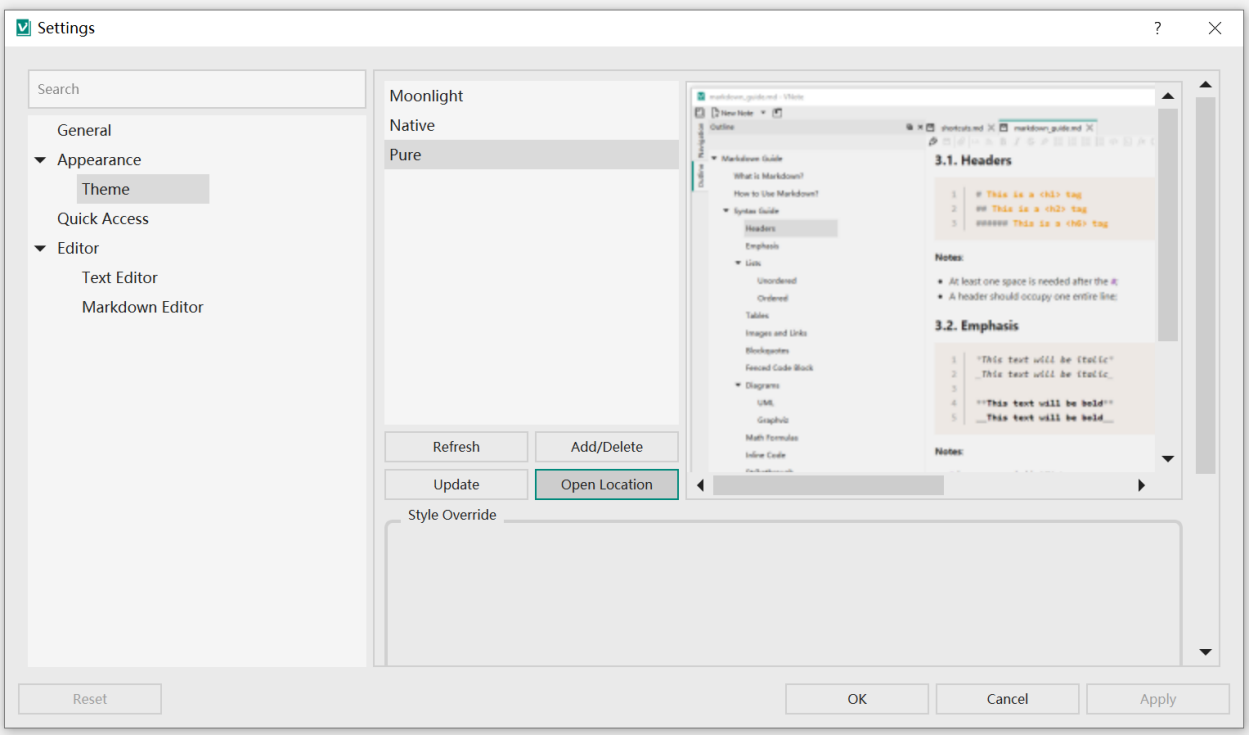
CHAPTER 7

Themes and Styles

7.1 Themes

A **theme** specifies the look of VNote, the style of the editor and read mode, and the syntax highlight style of code block.

A theme corresponds to a folder in the `themes` folder. You could change and manage themes in the `Settings` dialog.



7.1.1 How to Add A Theme

It is a good practice to start a custom theme based on an existing theme. Copy the folder of your favorite theme and paste it into the `themes` folder under **user configuration** folder. Remember to rename the folder.

7.1.2 Components of A Theme

Some key files of a theme:

- `palette.json`: the palette of a theme which defines several colors to be used in the theme;
- `interface.qss`: file for **Qt Style Sheet**, which specifies the look of all the widgets; it will use the colors defined by `palette.json`;
- `text-editor.theme`: theme file of the text editor (as well as Markdown editor);
- `web.css`: style sheet file of the read mode of Markdown;
- `highlight.css`: style sheet file of the read mode of Markdown for code block syntax highlight; VNote uses Prism for syntax highlight in read mode;

7.2 Samples

7.2.1 Custom Fonts

Read Mode

For the font in **read mode**, it is specified in `web.css` by the `font-family` and `font-size`.

Styles of the body:

```
body {
    margin: 0 auto;
    font-family: -apple-system, "Noto Sans", "Helvetica Neue", "Segoe UI", Helvetica,␣
↪sans-serif, Tahoma, Arial, Geneva, Georgia, Palatino, "Times New Roman", "", "YaHei␣
↪Consolas Hybrid", "Microsoft YaHei", "", "Microsoft YaHei UI", "WenQuanYi Micro Hei
↪", "", Dengxian, "", STXihei, "", "Liberation Sans", "Droid Sans", NSimSun, "",␣
↪SimSun, "", "Apple Color Emoji", "Segoe UI Emoji";
    color: #222222;
    line-height: 1.5;
    padding: 16px;
    background-color: #f5f5f5;
    font-size: 16px;
}
```

Styles of the code blocks and inline code:

```
code {
    font-family: "YaHei Consolas Hybrid", Consolas, Monaco, Monospace, Courier;
    color: #8e24aa;
    word-break: break-word;
}

pre code {
    display: block;
```

(continues on next page)

(continued from previous page)

```
    padding-left: 0.5em;
    padding-right: 0.5em;
    color: #222222;
    background-color: #e0e0e0;
    line-height: 1.5;
    font-family: "YaHei Consolas Hybrid", Consolas, Monaco, Monospace, Courier;
    white-space: pre;
    -moz-tab-size: 4;
    -o-tab-size: 4;
    tab-size: 4;
}
```

For code blocks with syntax highlight, styles are defined in `highlight.css`:

```
code[class*="language-"],
pre[class*="language-"] {
    color: black;
    background: none;
    font-family: Consolas, Monaco, 'Andale Mono', 'Ubuntu Mono', monospace;
    font-size: 1em;
    text-align: left;
    white-space: pre;
    word-spacing: normal;
    word-break: normal;
    word-wrap: normal;
    line-height: 1.5;

    -moz-tab-size: 4;
    -o-tab-size: 4;
    tab-size: 4;

    -webkit-hyphens: none;
    -moz-hyphens: none;
    -ms-hyphens: none;
    hyphens: none;
}
```

Text Editor and Markdown Editor

For the font of text editor and Markdown editor, it is specified in `text-editor.theme` file.

Text editor:

```
{
    "editor-styles" : {
        "Text" : {
            "//comment" : "Support a list of fonts separated by ,",
            "font-family" : "YaHei Consolas Hybrid, Consolas, Monaco, Andale Mono,␣
↪Monospace, Courier New",
            "font-size" : 12,
            "text-color" : "#222222",
            "background-color" : "#f5f5f5",
            "selected-text-color" : "#f5f5f5",
            "selected-background-color" : "#1976d2"
        }
```

(continues on next page)

(continued from previous page)

```
    }
}
```

Markdown editor:

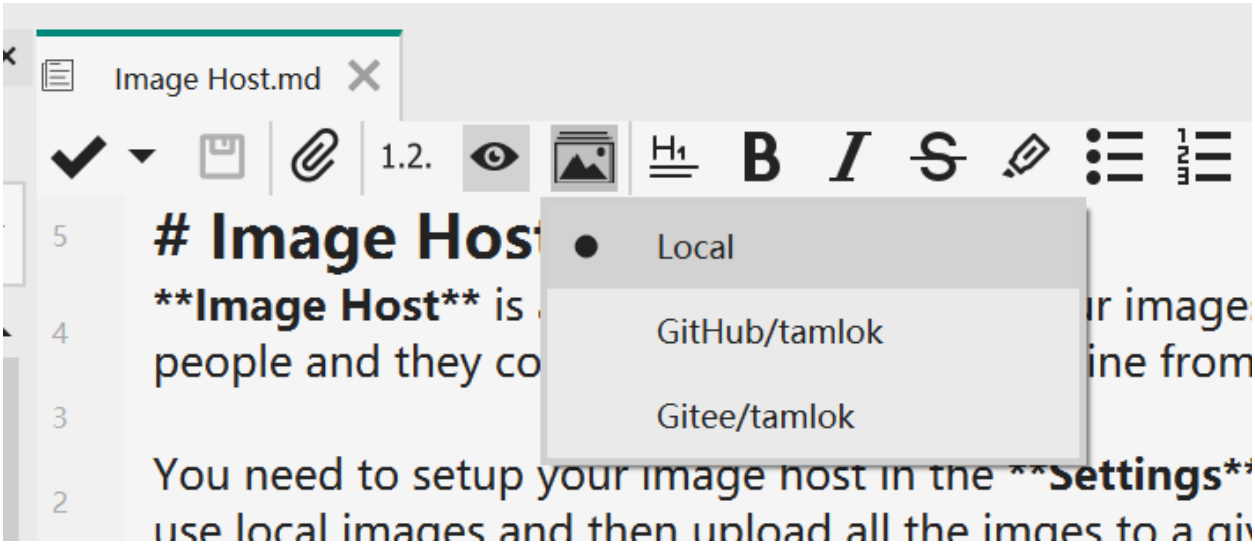
```
{
    "markdown-editor-styles" : {
        "Text" : {
            "//comment" : "Support a list of fonts separated by ,",
            "font-family" : ", YaHei Consolas Hybrid, Microsoft YaHei, , Microsoft␣
↪YaHei UI, WenQuanYi Micro Hei, , Dengxian, , STXihei, , Liberation Sans, Droid Sans,
↪ NSimSun, , SimSun, , Verdana, Helvetica, sans-serif, Tahoma, Arial, Geneva,␣
↪Georgia, Times New Roman",
            "font-size" : 12,
            "text-color" : "#222222",
            "background-color" : "#f5f5f5",
            "selected-text-color" : "#f5f5f5",
            "selected-background-color" : "#1976d2"
        }
    }
}
```

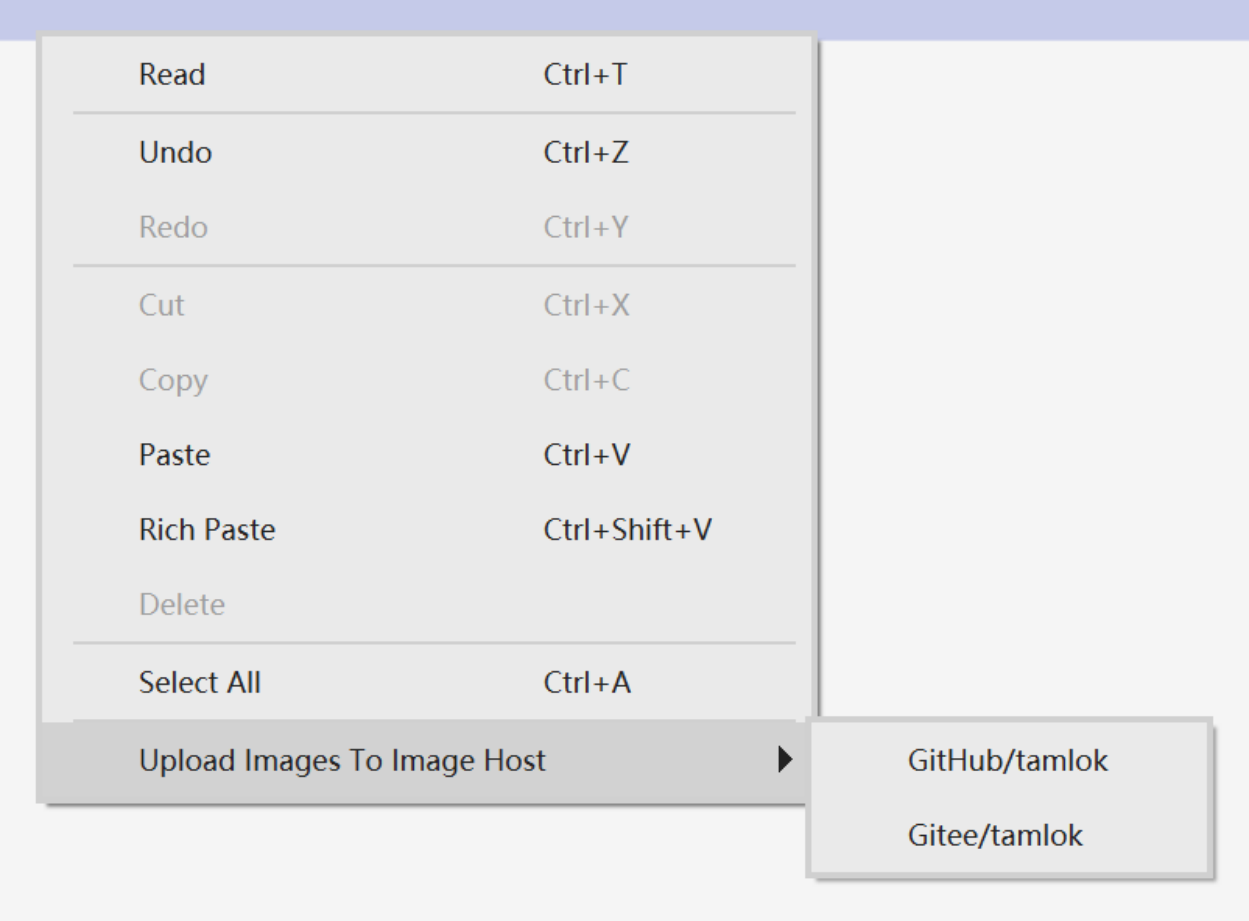
CHAPTER 8

Image Host

Image Host is a online service to hold your images. Different from local images, with image host, you could just share your plain text Markdown file wihtout any image files to people and they could access your images online from anywhere.

You need to setup your image host in the **Settings** dialog at first. Then you could choose to use local images or image host at the editor. If network is poor, you could also first use local images and then upload all the imges to a given image host at the end.



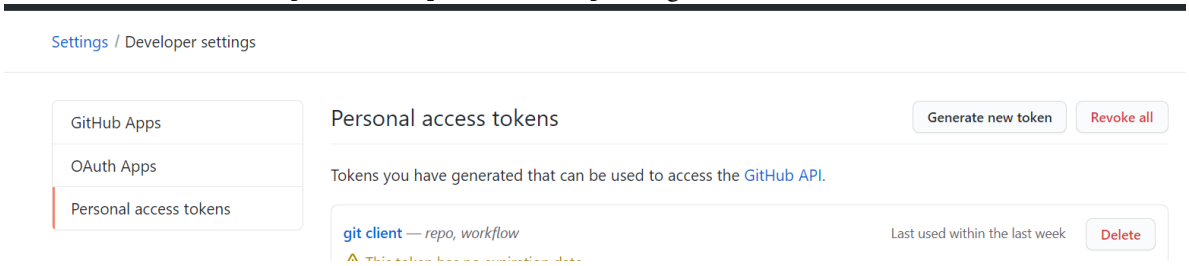


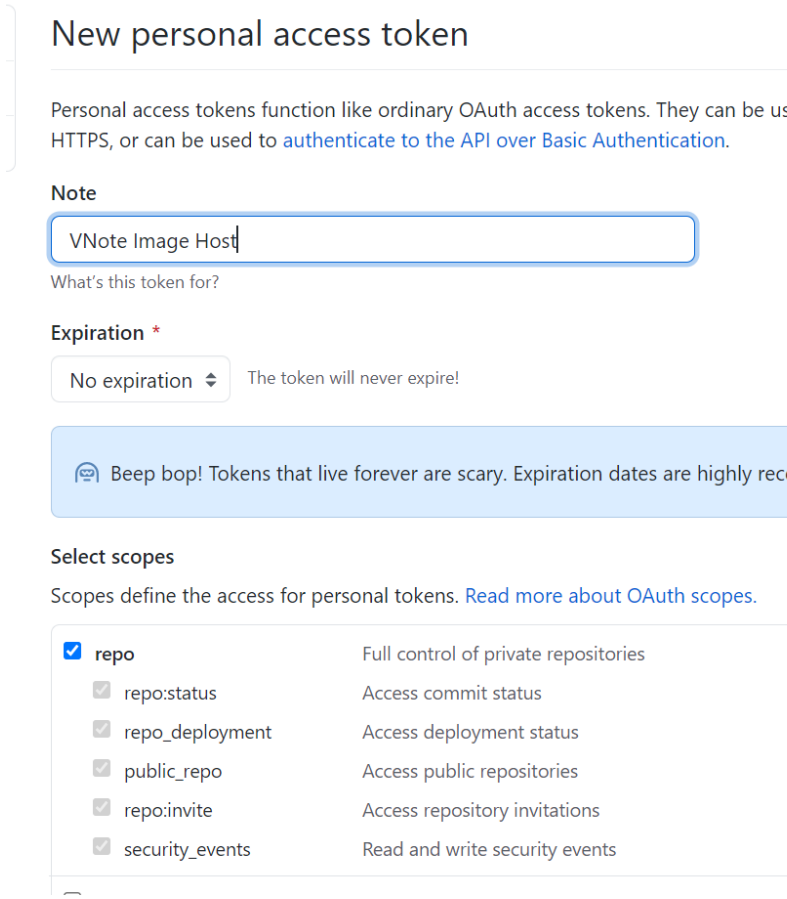
8.1 Configuration

8.1.1 GitHub/Gitee

Gitee shares the similar process with GitHub. Let's take GitHub as an example.

1. Go to the GitHub `Settings`, `Developer settings` and generate a new `Personal access tokens`.





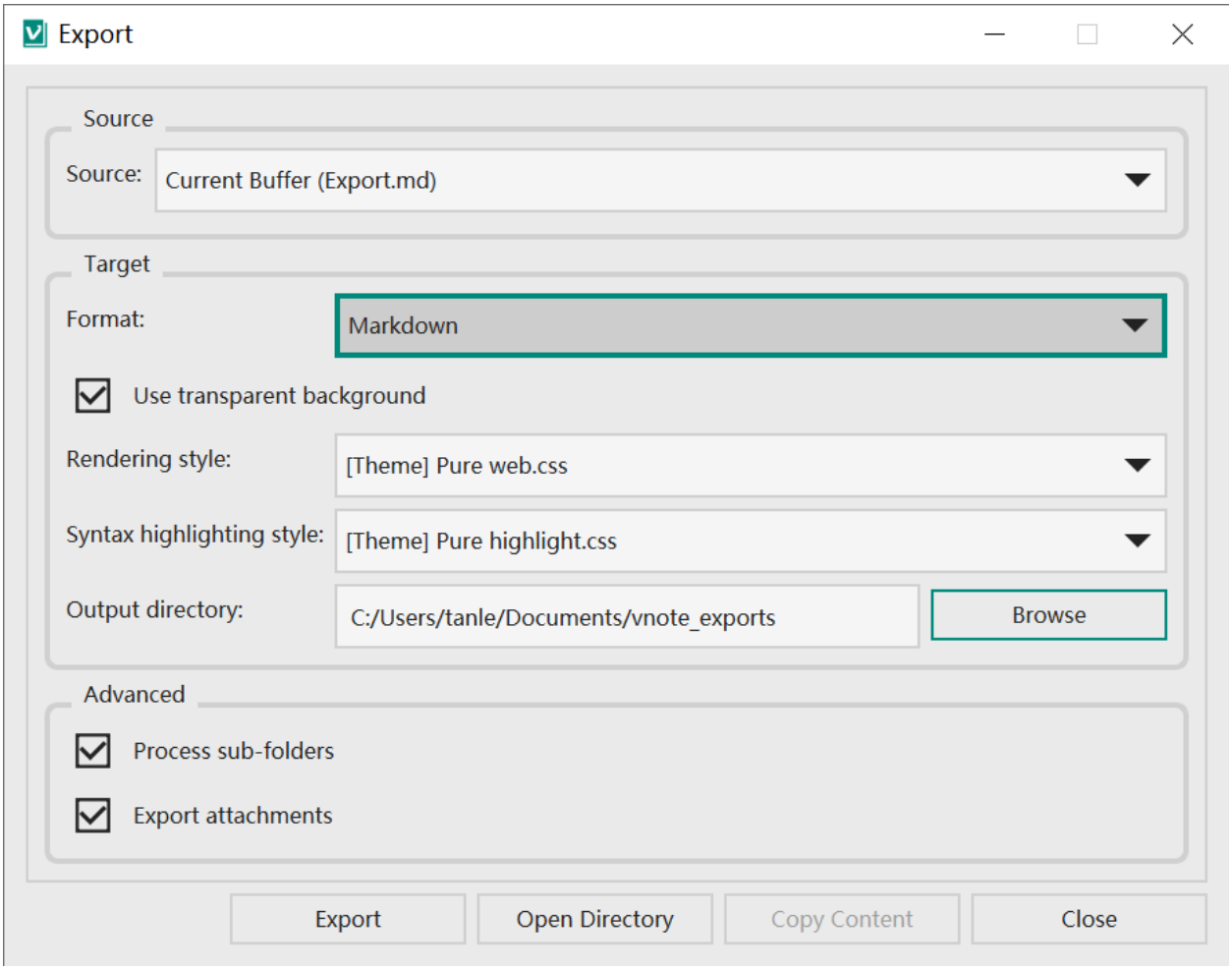
2. Select the `repo` scopes and generate the token. Copy it.
3. Create a **public** repository to hold the images. Remember to generate the default `README` file to create the first commit.
4. New a image host in VNote and fill in the **Personal Access Token**, **User Name** and the **Repository Name**.

CHAPTER 9

Export

VNote supports exporting notes to multiple formats, such as `Markdown`, `PDF`, and `HTML`. With Pandoc, more target formats are supported.

9.1 General Settings

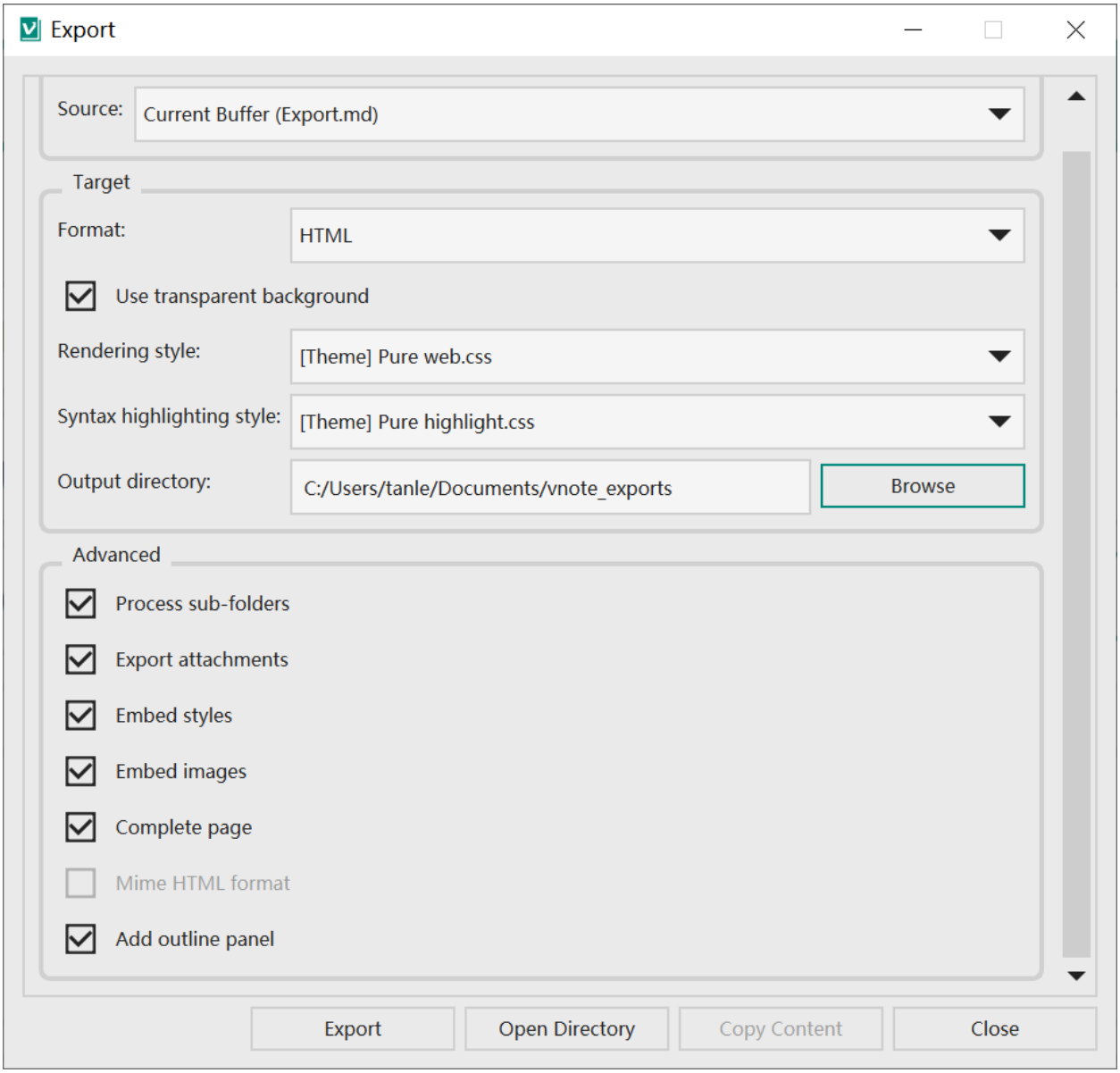


- VNote allows exporting current note, notes in current folder, and notes in current notebook.
- Target formats are `Markdown`, `HTML`, `PDF`, and `Custom`.
- VNote allows specifying different rendering styles in export.

9.2 Markdown

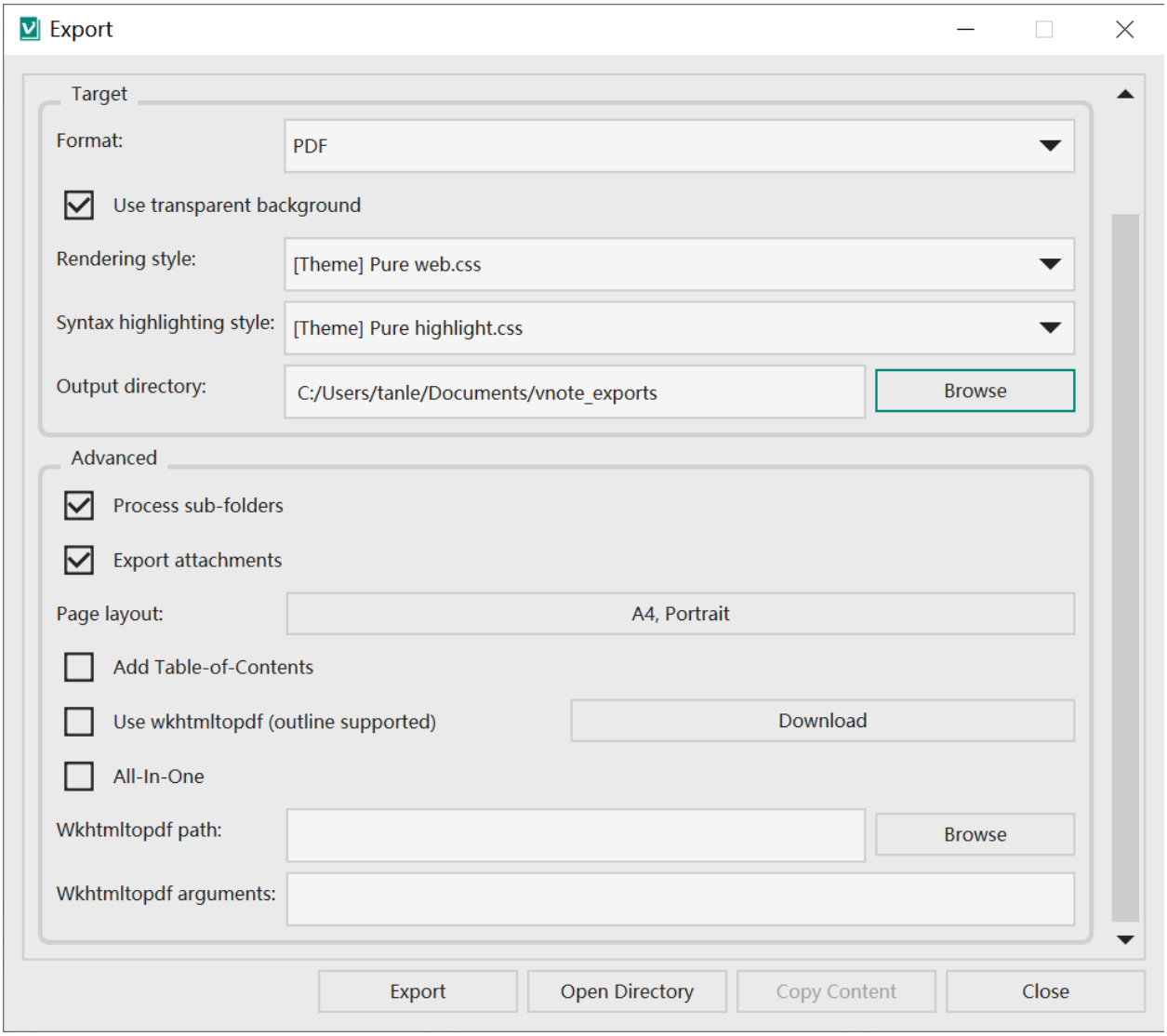
VNote will export the file into `Markdown` format within one single folder, with its images and attachments.

9.3 HTML



VNote allows to export file into one single `HTML` page, with its styles and images embedded, so it is easier to share it to others.

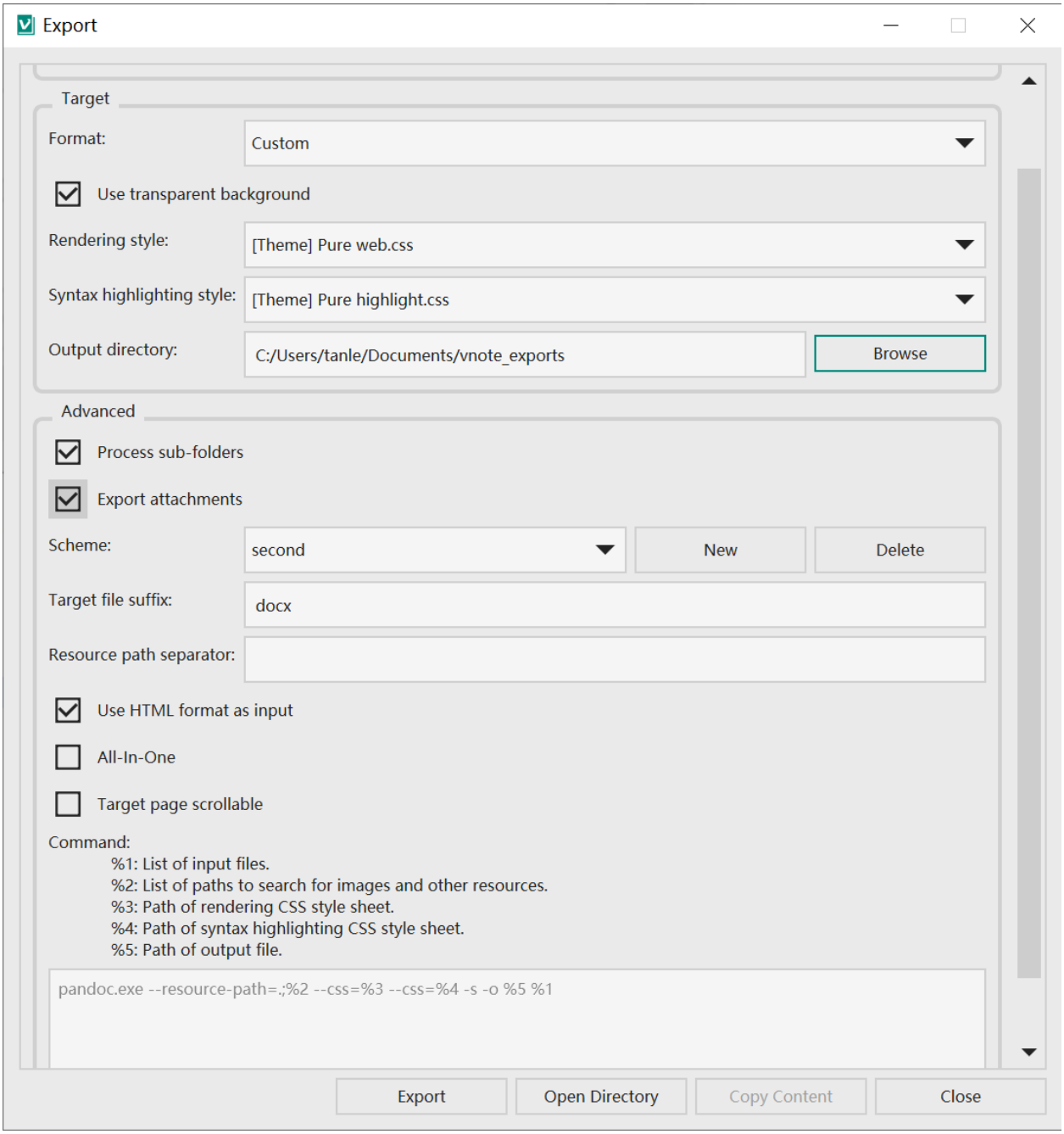
9.4 PDF



VNote supports exporting to `PDF` directly or using `wkhtmltopdf` tool to export. With `wkhtmltopdf`, outline is supported.

`All-In-One` option allows to export and combine multiple source files into one target file.

9.5 Custom



VNote allows to export files using customized command. It is common to use `pandoc` tool to convert to other formats. Please refer to the documentation of `pandoc` for details. You could even use a script to handle it in your own way.

The following command on Windows could export to nearly every format via `pandoc`. Just change the `Target file suffix` into `docx`, `epub`, and so on.

```
"c:\your\path\to\the\downloaded\executable\pandoc.exe" --resource-path=.;%2 --css=%3 -
↪-css=%4 -s -o %5 %1
```

On non-Windows platforms, please change the separator from `;` to `:`.

CHAPTER 10

Frequently Asked Questions

10.1 How to Specify Custom MathJax Script?

VNote uses MathJax 3 to render math formulas. To specify the MathJax script to use, do the following steps:

1. Open the default configuration folder, and copy `web/js/mathjax.js` to the user configuration folder as `web/js/mathjax.js`, then you could modify the copied `mathjax.js` file here. VNote will use this file instead of the default one.
2. Edit the `mathjax.js` file. The MathJax script to use is given by:

```
        this.mathJaxScript = 'https://cdn.jsdelivr.net/npm/mathjax@3/es5/tex-svg.
↪js';
```

You could download the MathJax (howto) and specify a local copy of the MathJax to use:

```
this.mathJaxScript = 'file://c:/Users/foo/mathjax/tex-svg.js';
```

10.2 Crash After Version Update

This often happens if the update crosses multiple versions. Open user configuration folder and delete `vnotex.json`.

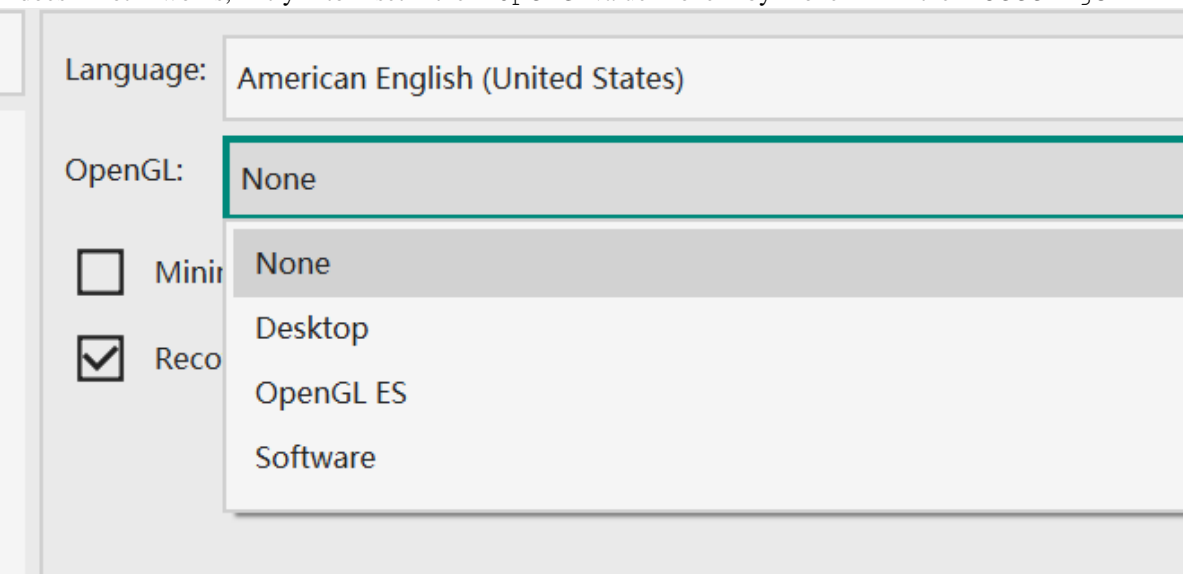
10.3 Interface Is Freezed

10.4 Cursor Is Not Visible in Edit Mode

10.5 Crash When Opening A Note

All the above 3 issues may be caused by the driver of the display card only on Windows. Steps to solve it:

1. Try to update the display card driver;
2. Try to schedule VNote to run with integrated display card;
3. If it does not works, try to set the `OpenGL`value one by one in the `Settings` dialog.



Restart of VNote is needed after each step to check if it takes effect.