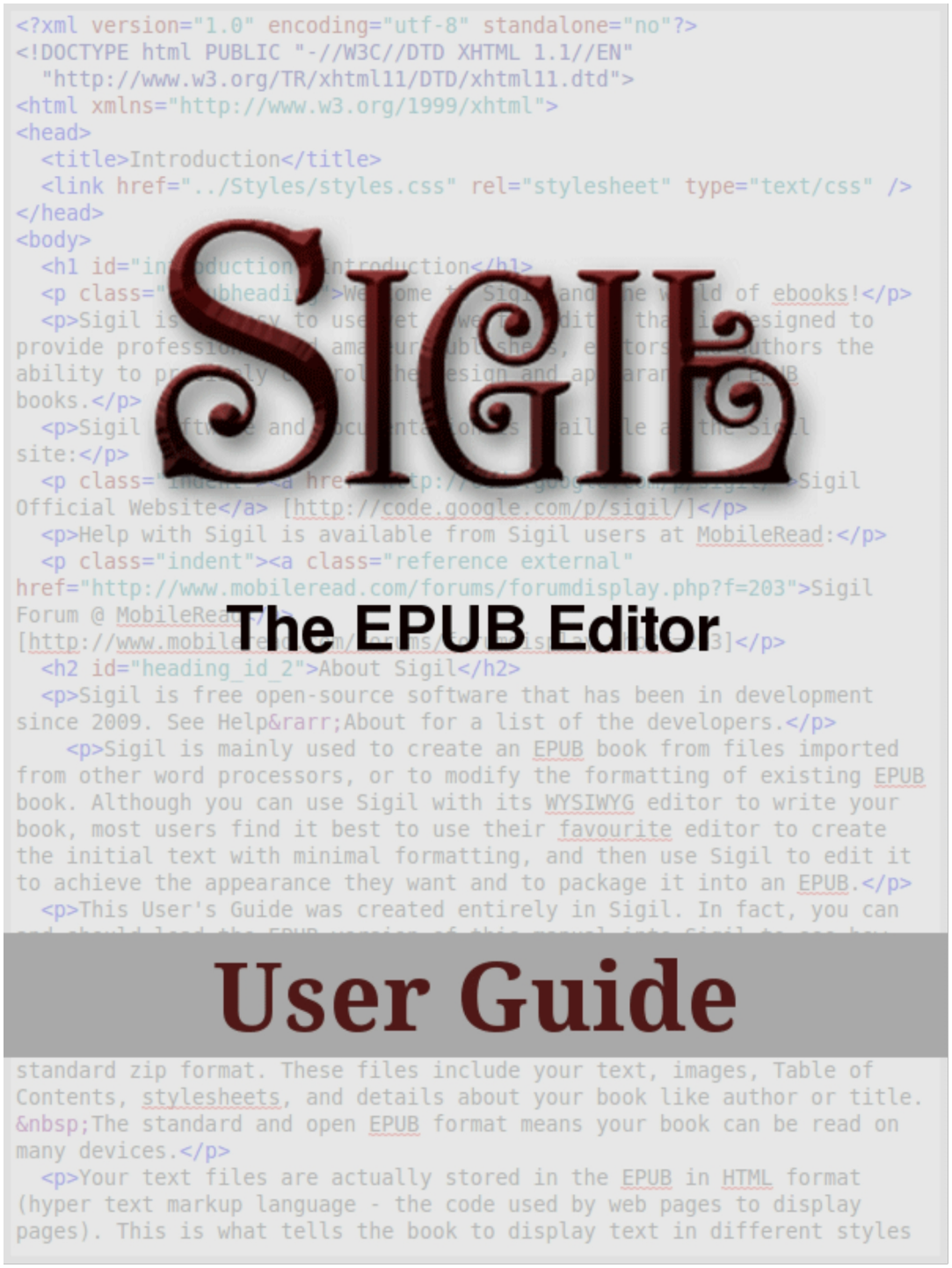

SIGIL
The EPUB Editor
User Guide

Sigil User Guide

For Sigil Versions 1.9.10 and Later

The Sigil MobileRead Documentation Team:

Ashjuk
Banjo
BeckyEbook
DiapDealer
Doitsu
KevinH
odamizu
Tex2002ans

Original Version by: Dave Heiland

Copyright © 2014-2022 Sigil MobileRead Documentation Team

Copyright © 2012-2013 Dave Heiland

All rights reserved.

This work is licensed under a Creative Commons 3.0 Attribution-Share License.

Use of these materials is permitted only in accordance with license rights granted. Materials provided “AS IS”; no representations or warranties provided. User assumes all responsibility for use, and all liability related thereto, and must independently review all materials for accuracy and efficacy.

Table of Contents

Title Page
Copyright
Introduction
    What's New
    About Sigil
    About EPUB
Installation
Features
    User Interface
    Preferences
    Code View
    Preview
    Icon Themes
    Main UI
    Basics
    Security
    Validation
    Advanced
    Opening and Saving Files
    Code View
    Preview
    Book Browser
    Delete
    Rename
    ReGex Rename
    Move
    Merge
    Sort
    Link Stylesheets
    Add Semantics
    Open With
    Save As
    Add Blank HTML File/Stylesheet/SVG Image
    Add Copy
    Add Existing Files
    Select All
    Validate With W3C (CSS stylesheets only)
    Cover Image (image files only)
    Font Obfuscation (font files only)
    Renumber TOC Entries (.ncx files only)
    Metadata
    Add Cover
    Table of Contents
    Validation

- Spellcheck
- Images, Video, Audio
- Special Characters
- Using Checkpoints to Quickly Save/Restore Epub State
- Splitting and Merging
- Find & Replace
- What to Search For
- Saved Searches
- Clips
- Context Menu
- Clip Editor
- Clip Bar
- Clips Window
- Context Menu
- Keyboard Shortcuts
- Main Menu
- Clipboard History
- Links and IDs
- Styles
- Indexes
- Reports
- External Editors
- Preferred External Xhtml Editor
- Using the Page Inspector
- Manage Plugins
- Running Plugins
- Automate Lists
- Creating and Editing Automate Lists
- Running an Automate List
- Customize the User Interface
- Customizing the Toolbars
- Customizing the Panels
- Customizing the Icons

Tutorials

- Getting Started With EPUB
- Prepare Your File For Sigil
- Open Your File With Sigil
- Save Your EPUB File
- Add the Author and Title
- Add a Cover Image
- Create Files For Each Chapter
- Identify Your Chapters
- Generate A Table of Contents
- Create Links and Notes
- A Quick Introduction To Code View
- Check For Spelling Mistakes
- Edit With Find & Replace
- Use Stylesheets

- Include Custom Fonts
- Advanced Find & Replace – Regex
- Regular Expression Reference
- Tips

FAQ
- Common Questions
- Where to Get Help
- Using Sigil
- Formatting and Styles

Contributing to Sigil

Introduction

— Sigil: The EPUB Editor —

Welcome to the Sigil User Guide – updated for Sigil 1.9.10 and later.

Sigil 1.9.10+ is the result of countless hours of volunteer effort to create the fastest, most stable and powerful version of the best free EPUB editing software available.

Jump right to the Tutorials to learn how to use Sigil step by step. Or view What's New for changes in recent releases.

Download Sigil from:

- The Official Sigil Website

Help is available from other users and the developers of Sigil at MobileRead:

- Sigil Forum @ MobileRead

The Sigil User Guide includes documentation that will help you get the best out of editing your EPUB, including:

- **Installation:** Installing Sigil on Windows, MAC and Linux.
- **Features:** A complete description of all of Sigil's features.
- **Tutorials:** Step by step guides for using Sigil to create your book.
- **FAQ:** Frequently asked questions about Sigil.
- **Contributing to Sigil:** How to support Sigil.

You can download the Sigil User Guide ebook in EPUB format.

What's New

The latest versions of Sigil contain hundreds of changes, bug fixes and new features while still retaining the majority of the same easy to use interface of earlier versions. Almost all of Sigil's code has been updated and improved with particular focus on the following areas:

- **Stability:** All known crashes and related bugs from previous versions have been identified and fixed – making Sigil completely stable for updating your EPUBs.

- **Robustness:** Sigil can correct even more common problems when reading files, and will no longer exit when opening invalid EPUBs. Saving has been enhanced to perform more integrity checks. Splits and merges can now be done exactly where you want without potential data loss. The cursor position is correctly updated between Code View and Preview, and Find & Replace works consistently and accurately over one or all files.

- **Performance:** Sigil is now significantly faster in many areas: opening and changing tabs, generating a TOC, merging files, bulk renaming, replacing text, etc.

- **Flexibility:** More preferences, options and features were added to let you use Sigil the way you want.

For a detailed list of all the change made to Sigil, see the Changelog.

Recent Major Changes in Sigil include:

- **Switch from QtWebKit to QtWebEngine** QtWebEngine reduces memory growth and plugs many major memory leaks and has ongoing security and support patches from Qt/Google/Chromium

- **Removal of Book View** because it allowed users to mistakenly create very broken xhtml that was not acceptable in an EPUB. To replace part of its functionality, we created a new external program called PageEdit that is better suited for basic WYSIWYG proofing and simple Edits. Users who

know no xhtml or css will probably want to stay with Sigil 0.9.14 as it was the last version to support BookView. More on PageEdit and integrating it into Sigil can be found in this chapter.

- **Support for Python Plugins** to allow users to customize and integrate how they use Sigil into their own workflow and to add the tools they need or want independently from Sigil. These plugins can be downloaded and installed to add extended support to Sigil. These plugins include epubcheck, Access-Aide, KindleImport, Kindlegen, FlightCrew, ePub3-itizer, DOCX Import plugins, and etc. More about plugins, and where to download and how to install them will be described in a later chapter.

- **Support for Preserving User Specified entities**

- **Sigil embeds its own Python interpreter** (v3.8.3 at the time of this writing to support Python 3 based plugins.

- **Sigil uses a specially modified version of Google's Gumbo Parser** to auto correct xhtml parsing errors according to the html5 and xhtml specs.

- **Sigil now fully supports editing and creating reflowable EPUB2 and EPUB3 EPUBs.**

- **Sigil has removed its longtime restrictions on EPUB layouts.** Custom EPUB layouts are fully supported.

About Sigil

Sigil was designed to make it easy to create great ebooks using the EPUB format. If you are formatting books for your own use, or you are a professional editor publishing books on multiple platforms, then Sigil is for you. You can use Sigil to format and package your books into an EPUB that looks exactly the way you want it to using an advanced set of features that have made Sigil one of most popular EPUB editors available.

This open-source and completely free software is written and supported entirely by volunteers. The current Sigil development team includes Doug Massay and Kevin Hendricks as co-lead developers and maintainers. Previous developers/authors include John Schember, Grant Drake and Dave Heiland. Sigil was originally created by Strahinja Markovic in 2009 and remains completely free to download and use. Support is also provided by other volunteers who contribute code, translations and helpful answers to user's questions. See the **Help=>About** menu in Sigil for a complete list of contributors.

About EPUB

The EPUB (electronic publication) format is one of the most popular file formats for ebooks. It is an open and freely available standard that can be used by anyone.

An EPUB ebook is simply a collection of files each with different functionality stored together using the common zip format. These files include your words, images, Table of Contents, stylesheets, fonts and details about your book like author or title. EPUB's standard format means your book can be read on many e-readers or converted easily to other e-readers not using EPUB.

Your EPUB stores your words and text in the book using the an XML form called XHTML of the HTML format (Hyper Text Markup Language) – the same code used by web pages to display text and images). This just means that the files contain your text along with markup codes and tags (like `<h1>` or `<p>`) which tell e-readers how to format and display the text – some words could be bold or italic, others could be a paragraph or a heading, etc.

In addition to XHTML tags to control or style the layout, EPUB books typically make use of stylesheets to help organise the use of styles used by the XHTML files. A stylesheet is another file in the EPUB that contains a list of XHTML tags, classes, and selectors along with instructions on how to display any text using that matches the tag, class or selector, e.g. `<h1>` (heading level 1) should be larger and centered. This allows you to separate the content of your book from its layout and allows you to easily adjust the look of the EPUB.

Of course, when anyone actually reads an EPUB they don't see all these XHTML files or stylesheets. The e-reader just displays the text and images like a normal book or web browser – as close as it can to the formatting you defined, even if there are thousands of pages or files.

You don't have to understand all the details of the EPUB format or understand XHTML formatting codes to start using Sigil. However, if you spend any time with Sigil you will soon need to become familiar with these areas in order to get the most out of the advanced features of Sigil, and, ultimately, to achieve the design you want.

> Text files are stored in XHTML format (an XML compliant version of HTML), but to make the guide easier to read we'll use HTML or XHTML almost interchangeably.

Sigil is designed to fully work with both the older version 2 of the EPUB standard or the newer EPUB3 standard.

Even though Sigil is designed to work with EPUB files, the EPUB format can easily be converted to other ebook formats like MOBI.

To learn more about HTML you can read various online guides, including:

- W3Schools HTML Guide

Help with on the EPUB format is available from EPUB users at MobileRead:

- EPUB Forum @ MobileRead

Help with using Sigil itself is available from users at MobileRead:

- Sigil Forum @ MobileRead

If you really want to dig into the details of the official EPUB standard on which Sigil is built you can read the original technical specifications:

- Official EPUB 2.0.1 IDPF Standard
  Official EPUB 3.0 IDPF Standard
  Official EPUB 3.2 (Current Standard)
  - **OPS (Open Publication Structure):** The key specification which describes everything in an EPUB file.
  - **OPF (Open Packaging Format):** How the EPUB files relate to each other, metadata details (content.opf).
  - **OCF (OEBPS Container Format):** How files are packaged into a .epub file.

You can download the EPUB version of the Sigil User Guide, which was created entirely in Sigil, and view it in Sigil to see an example of an EPUB ebook.

Installation

— Getting Started —

You can download the latest version of Sigil from the official Sigil web site and install it for your OS.

Windows

Sigil is supported on Windows 10 and later.

- Download Sigil for Windows

Double-click the file to run the installer, e.g.:

- Sigil-1.9.10-Windows-x64-Setup.exe

Mac

Sigil is supported on MAC OS 10.12 x64 bit Intel or later

- Download Sigil for MAC

Unpack the TXZ file and drag the “Sigil.app” file to your Applications folder, e.g.:

- Sigil.app-1.9.10-Mac.txz

For Sigil versions prior to Sigil 1.9, if you want to install Sigil plugins that come with Tcl/Tk based user interfaces, you’ll also need to install ActiveTcl 8.6.x. Sigil version 1.9 and later include this by default as part of Sigil.

Linux

A binary version of Sigil is not available due to the number of versions of Linux.

However, since the source code is freely available, you can build from source for your version of Linux. And due to the effort of many volunteers, there is most likely a pre-packaged version of Sigil built for your Linux distribution available online:

- Building Sigil On Linux

Features

— An In-Depth Guide —

This section gives a quick overview of all of Sigil's features, followed by chapters detailing everything available in Sigil.

You can also read the Tutorials section for step-by-step instructions for common tasks in Sigil, and then refer back here for additional details.

Editing

You edit your book using Code View. The Code View editor allows you to directly edit the XHTML formatting of your files to give you complete control over your book and will allow you to use all of Sigil's advanced features. You can use Sigil's live Preview window with Code View to show you what your page will look like as you type.

NOTE: Book View and its WYSIWYG editing was removed from Sigil in version 0.9.15. See this chapter on integrating PageEdit with Sigil as an alternative WYSIWYG editing mechanism.

Find & Replace

The ability to search for or replace text is one of the most powerful functions in Sigil. You can search or replace in one file or in all the files at once, and you can search for simple text or use Regular Expressions to change or correct text or formatting. You can also use Saved Searches to store your searches for reuse.

Managing Files

The Book Browser organises all the files in your EPUB by type and gives you quick access to open files or to modify them. Double-click a file to open it in the editing window, or right-click on selected files to see options for renaming, moving, deleting, merging, adding, linking stylesheets, etc. The XHTML files are listed in the order they appear in your book – and you can drag and drop them to reorder them. You can also easily add or insert files into your book.

Table of Contents

The Table of Contents (TOC) window shows you exactly what your TOC will look like, and allows you to jump to specific chapters by clicking on chapter names. You can also generate a new TOC based on headings in your book at any time, edit the TOC directly, and create an HTML version of your TOC.

Please note that in EPUB3 books the Table of Contents window will display the contents of nav.xhtml (NAV) instead of toc.ncx to match the EPUB3 specification.

Modifying Book Details

The Metadata Editor provides a quick way to update book details like Title, Author, and Language. You can add any of the hundreds of standard EPUB metadata entries like publication date, co-authors, copyright information, etc.

Preferences

Many settings in Sigil can be customised in the Preferences window. In particular you can set Sigil's user interface language, keyboard shortcuts, which entities you would like to preserve in your code, text appearance, dictionaries, and automatic level of formatting and cleaning along with User Interface options, icon sizing, and plugin management.

Validation

Sigil includes a basic sanity check to make sure your code is at the minimum well-formed for parsing. To do more advanced checking you can install either the FightCrew plugin or the epubcheck plugin for more complete validation of your EPUB. Also included inside Sigil is the ability to easily validate your CSS via an online CSS Validation website service or via the new CSSValidator plugin.

Reports

Reports can be generated to display details about your files including size, number of words, whether images are in use, which styles are used, characters in the book, etc.

Spellchecking

You can check the spelling of your text using the powerful Spellcheck dialog that allows you to see all misspelled words and to save them to a dictionary or correct

them. Multiple user dictionaries can be created and you can use one or more dictionaries simultaneously.

Advanced Tools

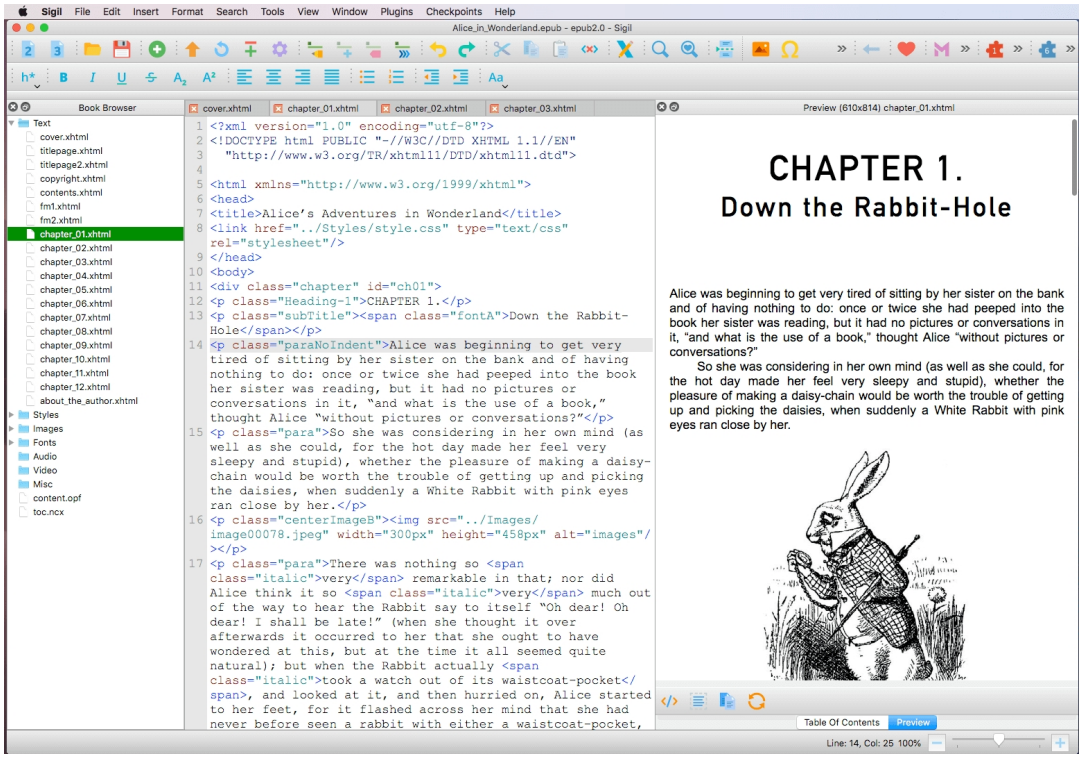
Sigil has a number of custom tools to assist with creating your ebook – including Saved Searches to save your Find & Replace searches, and Clips to save and paste text or styles. Unused images and styles can be removed automatically, files can be edited with external programs, and you can jump directly to styles used by HTML code.

Language Support

Sigil has been translated into over 23 languages and makes it easy to change the user interface language as needed. It is written in C++ for high performance and includes full Unicode UTF-8 support.

User Interface

— Windows, Menus and Toolbars —



Sigil's user interface consists of:

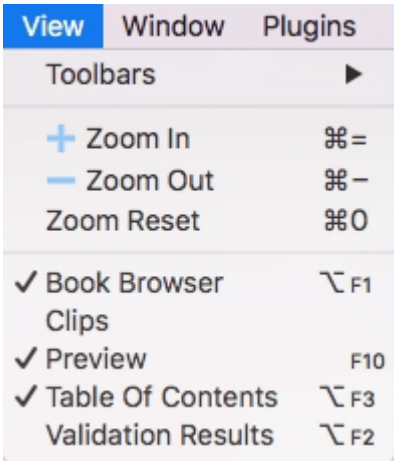
- **Windows:** Displays for information like Code View, Book Browser, Preview and Table of Contents.
- **Main Menu:** Most available commands, like **File=>Open**, can be accessed from the main menu bar at the top of the user interface.
- **Keyboard Shortcuts:** User-definable shortcuts for all menu items, like Ctrl +S to save your file.
- **Toolbar Buttons:** Movable buttons that provide quick access to many menu items.
- **Context Menus:** Commands shown when you right-click on something, like a filename or window.

> The screenshots in the User Guide may be slightly different from what you see on your computer, depending on your operating system.

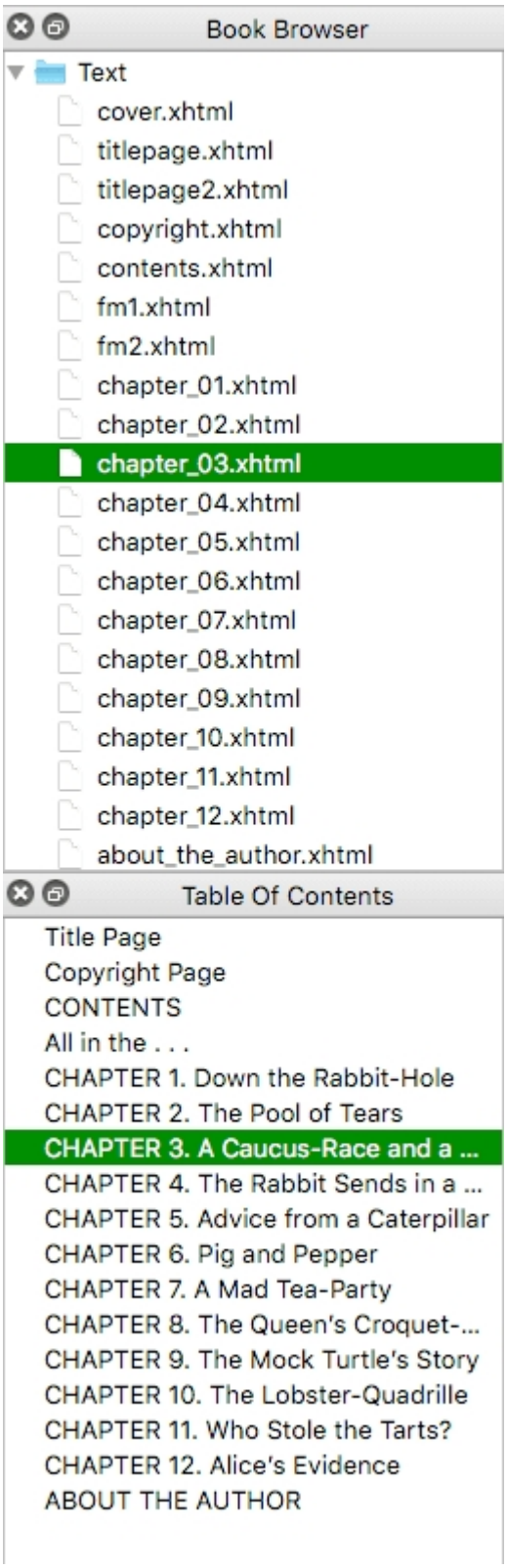
Windows

Sigil uses various windows in the main interface to present information: a window to show the files in the book (Book Browser), to edit your book (Code View), to see a live preview of your book (Preview), to view your table of contents, to view validation results, etc.

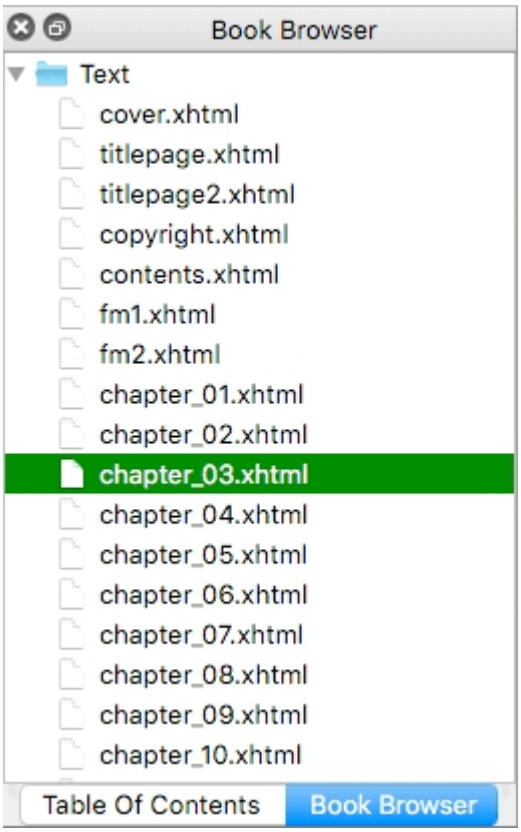
You can select which windows appear on your screen by using the **View** menu on the menubar:



Certain windows can also be moved. To move a window, click on its header bar and drag it to its new location. For instance you can stack the windows. Here's the Table of Contents placed below the Book Browser:

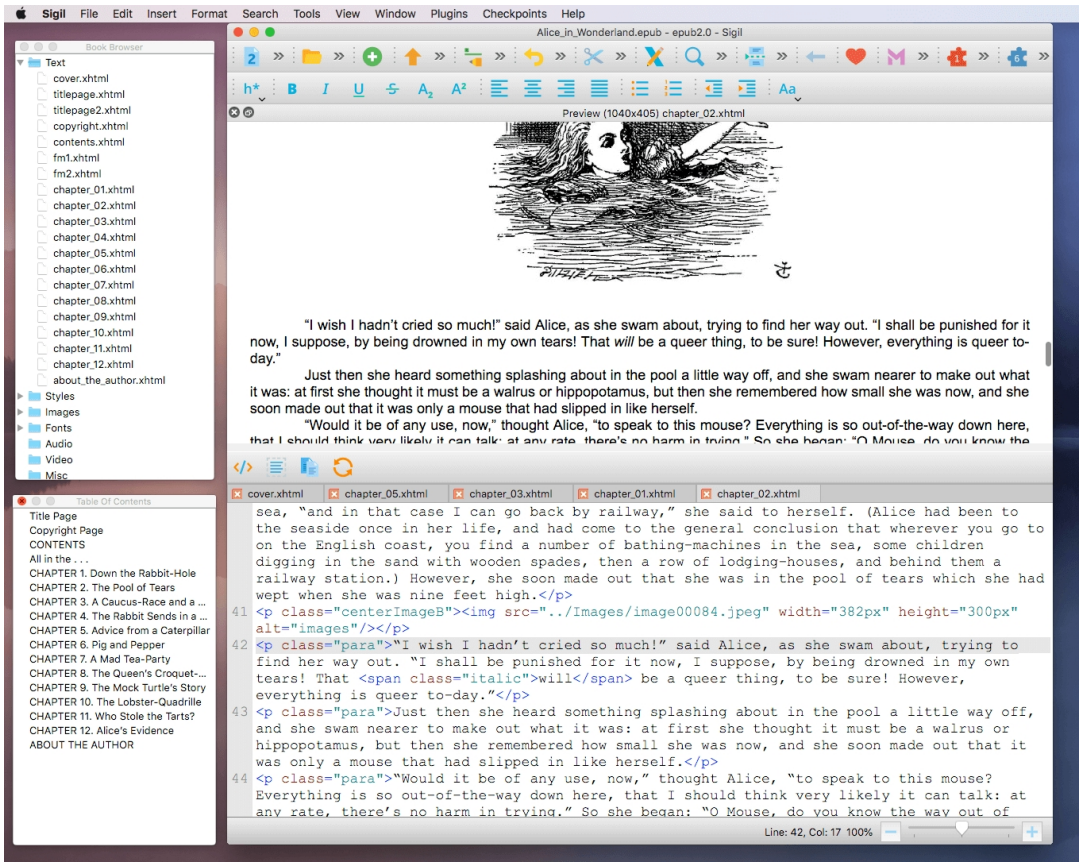


You can also create tabs for the Table of Contents and the Book Browser (and other windows) if you drag and drop it directly on the Book Browser window instead of just above or below it:



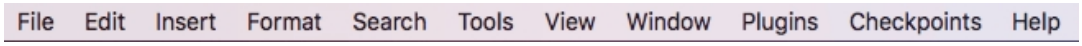
You can even undock windows (separate them from the main Sigil window so that they float on their own). To undock a window, double-click on the window's header bar. In some OS's you can also undock and re-dock the windows by clicking on the little button in the top left or right corner of the window.

Here's an example of the Preview window at the top, Code View at the bottom, and the Book Browser and Table of Contents undocked and floating on the left:



Main Menu

This is the list of menu items that is always at the top of Sigil's window or your screen:



File

- **New:** Create a new, empty ePub2 or ePub3.
- **Open:** Open an EPUB, HTML file, or text file from your computer.

- **Add:**
  - **Existing Files:** Add HTML files, images, etc. from your computer.

  - **Blank HTML File:** Create a new HTML file in the book.
  - **Blank Stylesheet:** Create a new CSS Stylesheet in the book.
  - **Blank SVG Image File:** Create a new SVG image file in the book.

- **Save:** Save your file (use often!).
- **Save As:** Save your file under a different name.

- **Save A Copy:** Save a copy of your file under a different name without renaming your current file (useful for backups).

- **Print Preview:** Preview the printing of the current tab.
- **Print:** Print the current tab.

- List of recently opened files.

- **Quit**: Exit Sigil, asking if you want to save your file first if it hasn't been saved. (On Mac, Quit is under the **Sigil** menu.)

Edit

- **Undo:** Undoes the last action, restoring text that has just been deleted, inserted or replaced.
- **Redo:** Reverse the last Undo.

- **Cut:** Remove the selected text and store it to the Clipboard for paste.
- **Copy:** Copy the selected text and store it to the Clipboard for paste.
- **Paste:** Insert text previously cut or copied to the Clipboard into your document.
- **Edit/Paste from Clipboard History:** View and edit a list of text snippets cut or copied to the Clipboard, and select and paste (insert) a snippet into your document. (To set the number of snippets to save, go to Preferences.)
- **Delete Line:** Delete the current line in Code View.

- **Change Case:**
    - **Lowercase:** Change the selected text to lowercase.
    - **Uppercase:** Change the selected text to uppercase.
    - **Titlecase:** Change the first letter of each selected word to uppercase.
    - **Capitalize:** Change the first letter of the selection to uppercase.

- **Split at Cursor**: Split the current file into two files immediately.
- **Split at Markers:** Split all files at previously inserted split markers.

- **Preferences:** Open the Preferences window to change options. (On Mac, Preferences is under the **Sigil** menu).

Insert

- **Split Marker:** Insert a marker to use for Split at Markers (see **Edit** menu).

- **File:** Insert an image, audio, or video file from your book or your computer.
- **Special Character:** Insert special characters from a chart.

- **ID:** Insert an anchor name that can be used with links.
- **Link:** Insert a link to something in your document or anywhere else.

- **Closing Tag:** Insert a tag to close the last unclosed tag.

- **Clip:** Select a clip to insert.

Format

- **Heading:**
  - **Heading 1:** Format the paragraph as a level 1 heading.
  - **Heading 2:** Format the paragraph as a level 2 heading.
  - **Heading 3:** Format the paragraph as a level 3 heading.
  - **Heading 4:** Format the paragraph as a level 4 heading.
  - **Heading 5:** Format the paragraph as a level 5 heading.
  - **Heading 6:** Format the paragraph as a level 6 heading.
  - **Normal:** Format the paragraph as normal text.
  - **Preserve existing attributes:** Keep attributes like class when formatting.

- **Bold:** Make text bold.
- **Italic:** Make text italic.
- **Underline:** Make text underlined.
- **Strikethrough:** Make text striked-through.
- **Subscript:** Make text subscript.
- **Superscript:** Make text superscript.

- **Align Left:** Align text to the left margin (use Remove Formatting to reset to default).
- **Center:** Align text centered on the screen.
- **Align Right:** Align text to the right margin.
- **Justify:** Align text to left and right margins.

- **Bulleted List:** Change text into a list of items preceded by bullets.
- **Numbered List:** Change text into a list of items preceded by numbers.

- **Decrease Indent:** Decrease the indentation of the text.
- **Increase Indent:** Increase the indentation the text.

- **Text Direction LTR:** Set the direction of text from left to right.
- **Text Direction RTL:** Set the direction of text from right to left.
- **Text Direction Default:** Set the direction of the text to the default.

- **Remove Formatting:** Delete tags within selected text.
- **Remove Tag Pair:** Remove open and close tag (place the cursor inside the tag to select it).

Users can shift double-click in a tag to select just that tag's contents. Or use alt double-click in a tag to select that tag and its contents. This can be useful for using **Remove Formatting**

Search

- **Find & Replace:** Open the Find & Replace window.

- **Find Next:** Search forward for the next matching text.
- **Find Previous:** Search backward for the previous matching text.

- **Replace:** Replace matched text.
- **Replace/Find Next:** Replace matched text and search forward for the next matching text.
- **Replace/Find Previous:** Replace matched text and search backward for the previous matching text.
- **Replace All:** Replace every instance of matching text according to the Find & Replace settings.

- **Count All:** Count every instance of matching text according to the Find & Replace settings.

- **Dry Run Replace All:** Perform a dry-run of the results from a Replace All showing every instance of matching text and its replacement according to the Find & Replace settings. No actual replacements are made.

- **Filter Replacements:** Create a table showing all matching text and potential replacements according to the Find & Replace settings. The user can then delete any undesired replacements before applying any changes.

- **Current File:**
    - **Find Next In File:** Search only in the current file regardless of Find & Replace mode.

    - **Replace Next In File:** Replace then Find Next only in the current file regardless of Find & Replace mode.
    - **Replace All In File:** Replace All only in the current file regardless of Find & Replace mode.

    - **Count All In File:** Count All only in the current file regardless of Find & Replace mode.

- **Bookmark Location:** Bookmark your current location in Code View.

- **Go To Link Or Style:** Go to the selected link, or go to the CSS style for this tag.
- **Back:** Go back to where you set a bookmark, clicked a link or used Go to Link or Style.

- **Mark Selected Text:** Mark the selected text for use with Find & Replace, which will limit the search to the selected text only.
- **Go To Line:** Jump to a specific line number in Code View.

Tools

- **Add Cover:** Select an image to make it the cover.
- **Metadata Editor:** Open the Metadata Editor to edit book details.

- **Table Of Contents:**
  - **Generate Table Of Contents:** Use predefined headings to generate a TOC.
  - **Edit Table Of Contents:** Edit the TOC manually.
  - **Create HTML Table Of Contents:** Create an HTML file version of the TOC.

- **Restructure Epub to Sigil Norm:** Starting with Sigil 1.0.0, EPUBs loaded into Sigil are kept "as-is"; no files or folders are moved or renamed. This is a departure from Sigil 0.9.18 and earlier, which automatically restructured each EPUB into a standard Sigil form (one OEBPS folder, specific names for the OPF and NCX, only one folder for text, images, etc.). Users who prefer the standard Sigil form may use Restructure Epub to Sigil Norm to manually restructure their EPUB to the old Sigil standard.
- **Create a Custom Empty Epub:** Design a custom EPUB layout to use when creating new EPUBs.

- **Index:**
  - **Index Editor:** Manage your index entries.

  - **Add To Index Editor:** Add the selected text to the Index Editor list.
  - **Mark For Index:** Add the selected text to the index when it is created.

  - **Create Index:** Create an HTML file containing the index list.

- **Spellcheck:**
  - **Spellcheck:** Open dialog to spellcheck all words at once.

  - **Highlight Misspelled Words:** Toggle highlighting of misspelled words in Code View.
  - **Next Misspelled Word:** Find the next misspelled word in Code View.

  - **Add Misspelled Word:** Add the misspelled word to the default dictionary.

    - **Ignore Misspelled Word:** Ignore the misspelled word temporarily while Sigil is running. Ignored words are also cleared when you open a new book.

    - **Clear Ignored Words:** Clear the list of temporarily ignored words.

- **Reformat HTML:**
    - **Mend and Prettify all HTML Files:** Fix HTML syntax errors in all HTML files and pretty-print them.
    - **Mend all HTML Files:** Fix HTML syntax errors in all HTML files.

- **Epub3 Tools:**
    - **Update Manifest Properties:** Add or update mandatory manifest properties.
    - **Generate NCX/Guide for epub2 e-readers:** Populate the NCX file with TOC entries from the NAV file (in EPUB3s, the NCX isn't updated automatically since the NCX isn't required; use this tool to manually update the NCX). Also populate the Guide with Landmarks entries from the NAV file.
    - **Remove the NCX and Guide:** Remove the NCX and Guide, both of which are not required in EPUB3, but are sometimes included for backward compatability with EPUB2 reading systems.

- **Well-Formed Check EPUB:** A simple validation tool to check if your EPUB is well-formed.
- **Validate Stylesheets With W3C:** Check if your CSS files are valid using the W3C web service.
- **Reports:** Display various details about HTML files, images, CSS styles, etc.

- **Clip Editor:** Create and edit snippets of text and styles, and paste them into your document.
- **Saved Searches:** Create, edit, and run saved Find & Replace searches.

- **Delete Unused Media Files:** Remove unlinked image, video, and audio files from your book.
- **Delete Unused Stylesheet Selectors:** Remove unreferenced selectors from CSS files.

View

- **Toolbars**
    - **New:** Show or hide the buttons to create New ePub2 and New ePub3.
    - **File:** Show or hide the Open and Save buttons.
    - **Add Existing:** Show or hide the Add Existing Files button.
    - **Checkpoints:** Show or hide the checkpoint buttons: Create Checkpoint for Epub, Create Epub from previous Checkpoint, Compare Epub against Checkpoint, Manage Checkpoint Repositories.

- **Undo/Redo:** Show or hide the Undo and Redo buttons.
- **Edit:** Show or hide the editing buttons: Cut, Copy, Paste, Remove Open and Close Tag Pair.
- **External Editor:** Show or hide the Launch External Xhtml Editor button. (To set your preferred External Xhtml Editor, go to **Preferences=>General Settings**.)
- **Find:** Show or hide the Find & Replace and Saved Searches buttons.
- **Insert:** Show or hide buttons to: Split at Cursor, Insert File, Insert Special Character, Insert ID, Insert Link.
- **Back:** Show or hide the Back button.
- **Donate:** Show or hide the Donate button.
- **Tools:** Show or hide the tool buttons: Metadata Editor, Generate Table of Contents, Edit Table of Contents, Spellcheck.
- **Plugins Set 1:** Show or hide 5 Plugin buttons. (To choose which plugins appear here, go to **Preferences=>Plugins=>Shortcuts**).
- **Plugins Set 2:** Show or hide 5 additional Plugin buttons. (To choose which plugins appear here, go to **Preferences=>Plugins=>Shortcuts**).
- **Heading:** Show or hide the heading buttons: h1, h2, etc.
- **Format:** Show or hide the formatting buttons: Bold, Italic, Underline, Strikethrough, Subscript, Superscript.
- **Align:** Show or hide the text alignment buttons: Align Left, Center, Align Right, Justify.
- **List:** Show or hide the list buttons: Bullets and Numbering.
- **Indent:** Show or hide the Decrease Indent and Increase Indent buttons.
- **Change Case:** Show or hide the change case buttons: Lowercase, Uppercase, Titlecase, Capitalize.
- **Text Direction:** Show or hide the text direction buttons: Left-to-Right, Right-to-Left, Default.
- **Clip Bar:** Show or hide the clip bar. (To create and manage clips, go to **Tools=>Clip Editor**.)
- **Clip Bar2:** Show or hide a second clip bar. (To create and manage clips, go to **Tools=>Clip Editor**.)
- **Index:** Show or hide the indexing buttons: Index Editor, Add the selected text to the Index Editor, Mark the selected text for inclusion in the Index, Generate a new Index HTML file.

- **Zoom In:** Increase the size of text and images in Code View or Preview (whichever window is selected).
- **Zoom Out:** Decrease the size of text and images in Code View or Preview (whichever window is selected).
- **Zoom Reset:** Reset the window zoom to 100%.

- **Book Browser:** Show or hide the list of files in the EPUB.
- **Clips:** Show or hide a selectable list of all text clippings.

- **Preview:** Show or hide a live preview window.
- **Table of Contents:** Show or hide a clickable TOC window.
- **Validation Results:** Show or hide the validation results window.

Window

- **Next Tab:** Change view to the next tab in Code View.
- **Previous Tab:** Change view to the preceding tab in Code View.
- **Close Tab:** Close the current tab in Code View.
- **Close Other Tabs:** Close all tabs other than the current tab in Code View.

- **Previous File:** Open the file above the selected file in the Book Browser.
- **Next File:** Open the file below the selected file in the Book Browser.

Plugins

- **Manage Plugins:** Add or remove plugins, and select which plugins appear in the toolbar.

Checkpoints

- **Create Checkpoint for Epub:** Save the current version of the EPUB as a checkpoint (stored via a local git repository).
- **Create Epub from previous Checkpoint:** Create a new EPUB from an earlier checkpoint version.
- **Compare Epub against Checkpoint:** Compare the current version of the EPUB with an earlier checkpoint version.
- **Manage Checkpoint Repositories:** View a list of all checkpoints, view logs, and remove selected checkpoint repositories.

Help

- **User Guide**: Official documentation for Sigil.

- **Donate**: How to donate to ongoing development.
- **Sigil Website**: Announcements and updates about development.

KEYBOARD SHORTCUTS

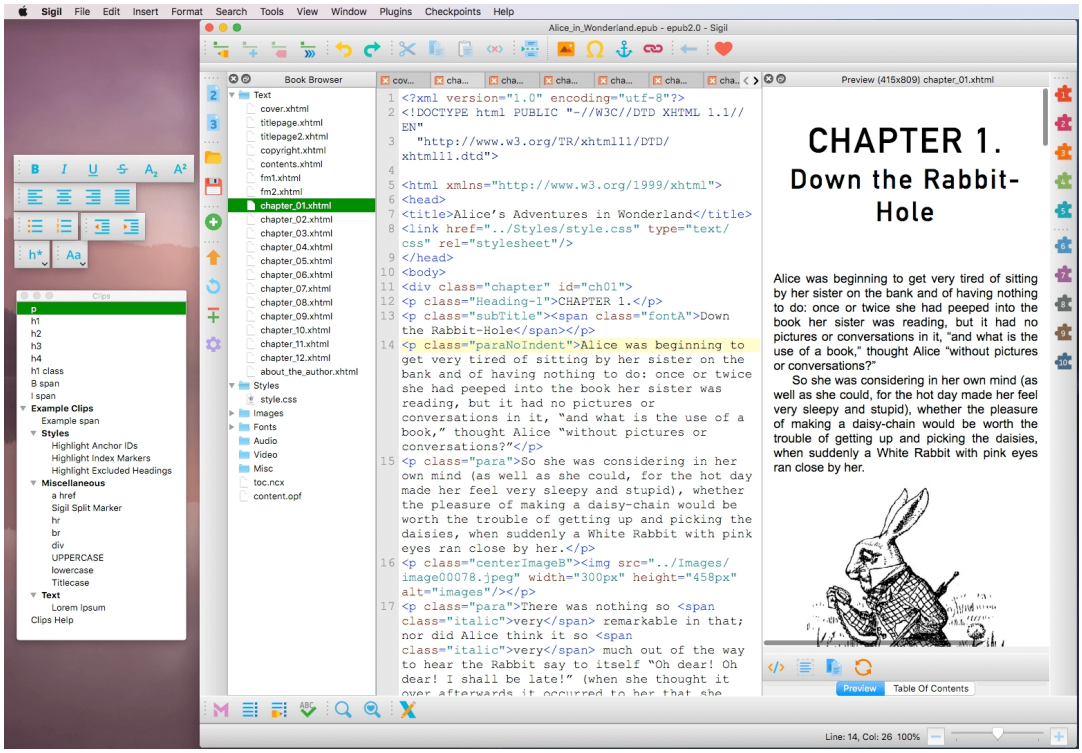
Keyboard shortcuts (e.g. Ctrl+S for Save) are shown next to menu entries and in **Edit=>Preferences: Keyboard Shortcuts** (for Mac: **Sigil=>Preferences: Keyboard Shortcuts**).

You can create or change keyboard shortcuts in Preferences.

TOOLBARS

Toolbars are rows of buttons that provide shortcuts to commonly used commands. Toolbars are positioned at the top of the window by default, but you can click and drag groups of buttons to run along the left, right or bottom of the window, if you like, or undock them as floating palettes.

Here's an example of toolbar buttons running along all four sides of the window, plus more buttons floating undocked, and the Clips window floating undocked:



To move toolbars, click and hold on the small bar to the left of each button group then drag it to where you want it.

Right-click on the toolbar to easily select which buttons or windows to show, or use the **View=>Toolbars** menu.

Hover over a button to see tooltips explaining what the button does.

Main Toolbar

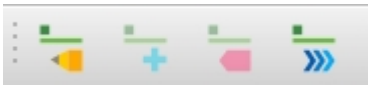
The main toolbar contains buttons used for shortcuts to Sigil's main features. They are shown below with their equivalent menu entry.

- **New ePub2: File=>New=>ePub2**
- **New ePub3: File=>New=>ePub3**

- **Open: File=>Open**
- **Save: File=>Save**

- **Add Existing Files: File=>Add=>Existing File**

- **Create Checkpoint for Epub: Checkpoints=>Create Checkpoint for Epub**
- **Create Epub from previous Checkpoint: Checkpoints=>Create Epub from previous Checkpoint**
- **Compare Epub against Checkpoint: Checkpoints=>Compare Epub against Checkpoint**
- **Manage Checkpoint Repositories: Checkpoints=>Manage Checkpoint Repositories**



- **Index Editor Tools=>Index=>Index Editor**
- **Add the selected text to the Index Editor: Tools=>Index=>Add To Index Editor**
- **Mark the selected text for inclusion in the Index: Tools=>Index=>Mark For Index**
- **Generate a new Index HTML file: Tools=>Index=>Create Index**

- Undo: **Edit=>Undo**
- Redo: **Edit=>Redo**

- Cut: **Edit=>Cut**
- Copy: **Edit=>Copy**
- Paste: **Edit=>Paste**

- Remove Open and Close Tag Pair: **Format=>Remove Tag Pair**

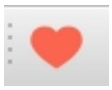
- Launch External Xhtml Editor: **(no equivalent menu entry)**

- Find & Replace: **Search=>Find & Replace**
- Saved Searches: **Tools=>Saved Searches**

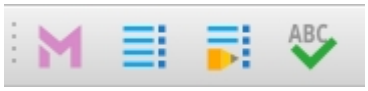
- Split At Cursor: **Edit=>Split At Cursor**

- Insert File: **Insert=>File**
- Insert Special Character: **Insert=>Special Character**
- Insert ID: **Insert=>ID**
- Insert Link: **Insert=>Link**

- Back: **Search=>Back**



- **Donate:** **Help=>Donate**



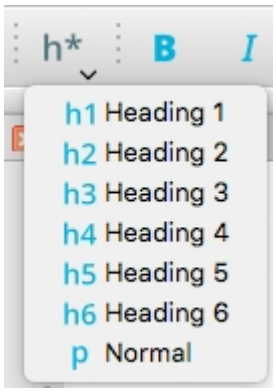
- **Metadata Editor:** **Tools=>Metadata Editor**
- **Generate Table of Contents:** **Tools=>Table Of Contents=>Generate Table Of Contents**
- **Edit Table of Contents:** **Tools=>Table Of Contents=>Edit Table Of Contents**
- **Spellcheck:** **Tools=>Spellcheck=>Spellcheck**

- **Run Plugin 1:** **(no equivalent menu entry)**
- **Run Plugin 2:** **(no equivalent menu entry)**
- **Run Plugin 3:** **(no equivalent menu entry)**
- **Run Plugin 4:** **(no equivalent menu entry)**
- **Run Plugin 5:** **(no equivalent menu entry)**

- **Run Plugin 6:** **(no equivalent menu entry)**
- **Run Plugin 7:** **(no equivalent menu entry)**
- **Run Plugin 8:** **(no equivalent menu entry)**
- **Run Plugin 9:** **(no equivalent menu entry)**
- **Run Plugin 10:** **(no equivalent menu entry)**

Formatting Toolbar

The formatting toolbar contains buttons used for formatting text. They are shown below with their equivalent menu entry.



- **Heading 1:** Format=>Heading=>Heading 1
- **Heading 2:** Format=>Heading=>Heading 2
- **Heading 3:** Format=>Heading=>Heading 3
- **Heading 4:** Format=>Heading=>Heading 4
- **Heading 5:** Format=>Heading=>Heading 5
- **Heading 6:** Format=>Heading=>Heading 6
- **Normal:** Format=>Heading=>Normal

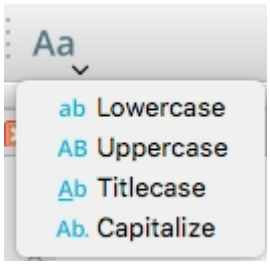
- **Bold:** Format=>Bold
- **Italic:** Format=>Italic
- **Underline:** Format=>Underline
- **Strikethrough:** Format=>Strikethrough
- **Subscript:** Format=>Subscript
- **Superscript:** Format=>Superscript

- **Align Left:** Format=>Align Left
- **Center:** Format=>Center
- **Align Right:** Format=>Align Right
- **Justify:** Format=>Justify

- **Bulleted List:** Format=>Bulleted List

- **Numbered List:** **Format=>Numbered List**

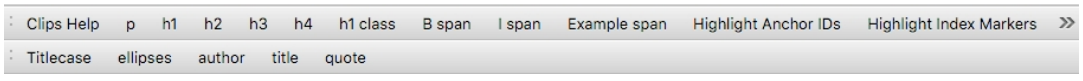
- **Decrease Indent:** **Format=>Decrease Indent**
- **Increase Indent:** **Format=>Increase Indent**



- **Lowercase:** **Edit=>Change Case=>Lowercase**
- **Uppercase:** **Edit=>Change Case=>Uppercase**
- **Titlecase:** **Edit=>Change Case=>Titlecase**
- **Capitalize:** **Edit=>Change Case=>Capitalize**

- **Left-to-Right:** **Format=>Text Direction LTR**
- **Right-to-Left:** **Format=>Text Direction RTL**
- **Default:** **Format=>Text Direction Default**

Clip Bars

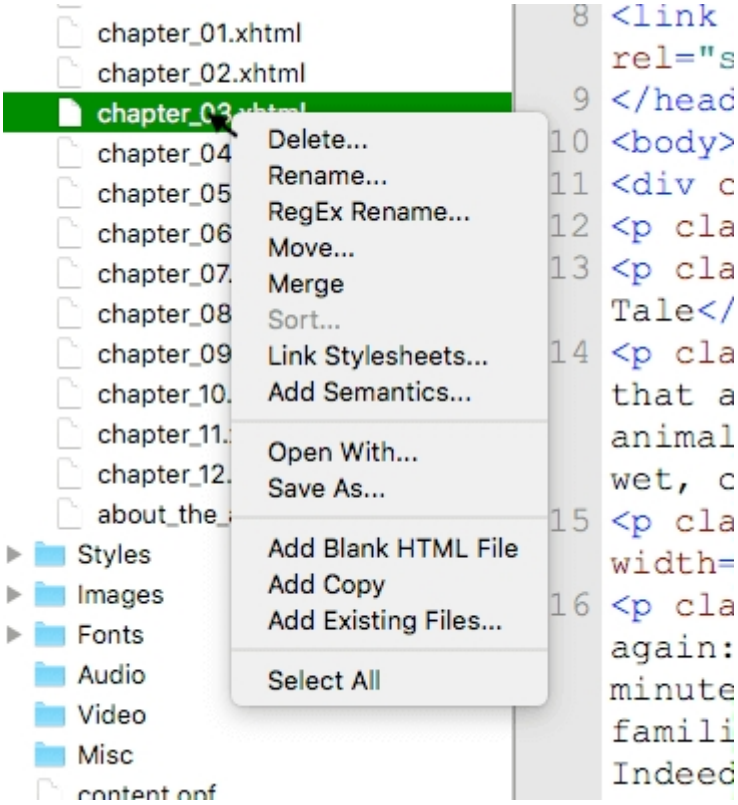


There are two clip bars: **Clip Bar** and **Clip Bar2**. Buttons are created for the first 40 clips not in a group. The first clip bar will contain the first 20 clips, the second clip bar will contain the next 20 clips. To create, edit, and manage clips, go to: **Tools=>Clip Editor**

Text direction, Clip Bar, and Clip Bar2 are not visible by default. Use **View=>Toolbars** to hide/unhide these button groups.

Context Menus

Context Menus are menus that appear when you **right-click** on text, a label, a window, a tab, a header, a text box, etc., giving you quick access to a variety of actions. Context menu items are discussed in related chapters.



When in doubt, right-click on something to see if there are additional actions available!

Zoom

On the lower right of the main window is a zoom control that lets you shrink or enlarge the text and images in the Code View and Preview windows independently. Click on the window you want to zoom, then click on the + or − icons to increase or decrease the zoom level. The percentage indicates the current zoom level. (Note:

Changing the zoom level on Preview will only work when the Preview window is docked.)

Status Bar

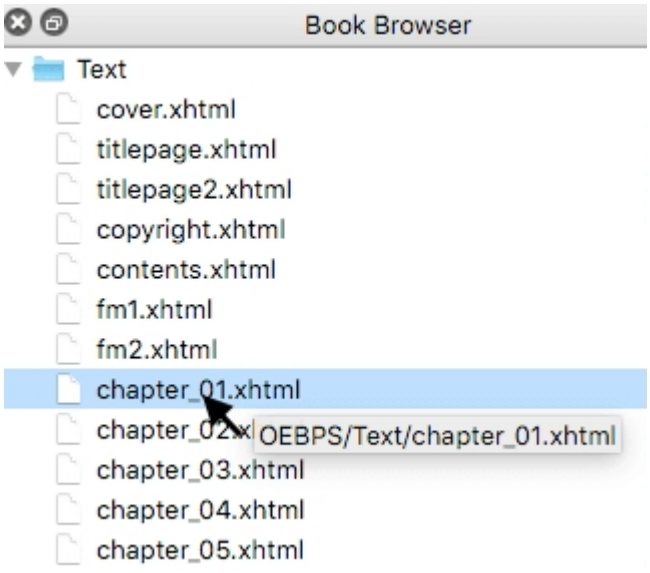
The bottom left corner of the main window will occasionally show status information when an action is completed, such as “EPUB saved” or “Table of Contents edited”.

Toward the right corner of the main window (before the zoom slider), you will find information about the currently opened file, such as the Line and Column in Code View.

Tooltips

Many of the buttons, dropdown boxes, labels and fields have tooltips that provide information on how they are used. To see a tooltip just hover your mouse over the button or other item and the help text should be displayed:

Hovering over a filename in the Book Browser will reveal the full filepath:



There are many tooltips available in Sigil. Most dialogs and buttons will provide some help text.

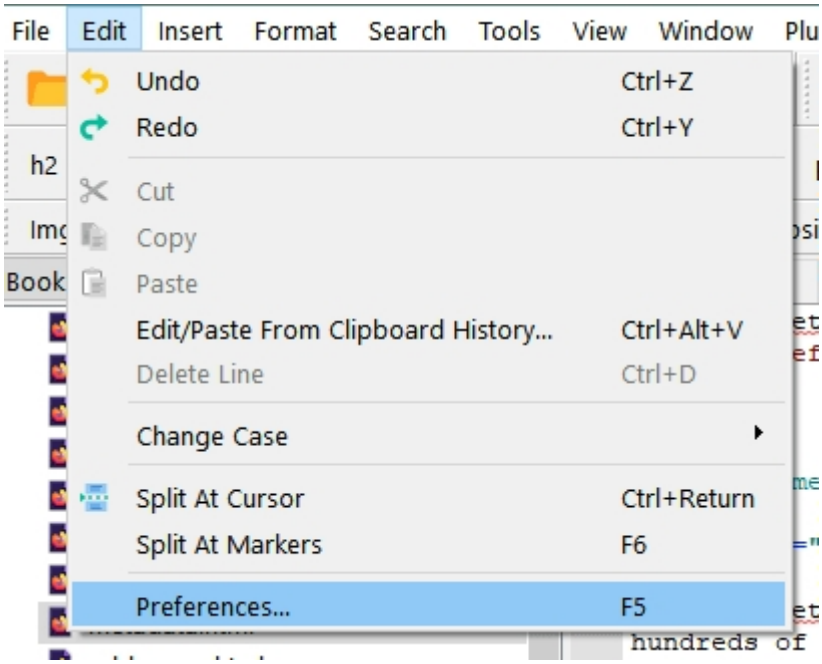
Preferences

— Customize Sigil —

The Preferences dialog gives you control over many settings in Sigil and is divided into seven sections:

- Appearance
- General Settings
- Keyboard Shortcuts
- Language
- Spellcheck Dictionaries
- Preserve Entities
- Plugins

To open Preferences, go to the main menu bar at the top of the screen and select **Edit=>Preferences** (on Mac: **Sigil=>Preferences**):



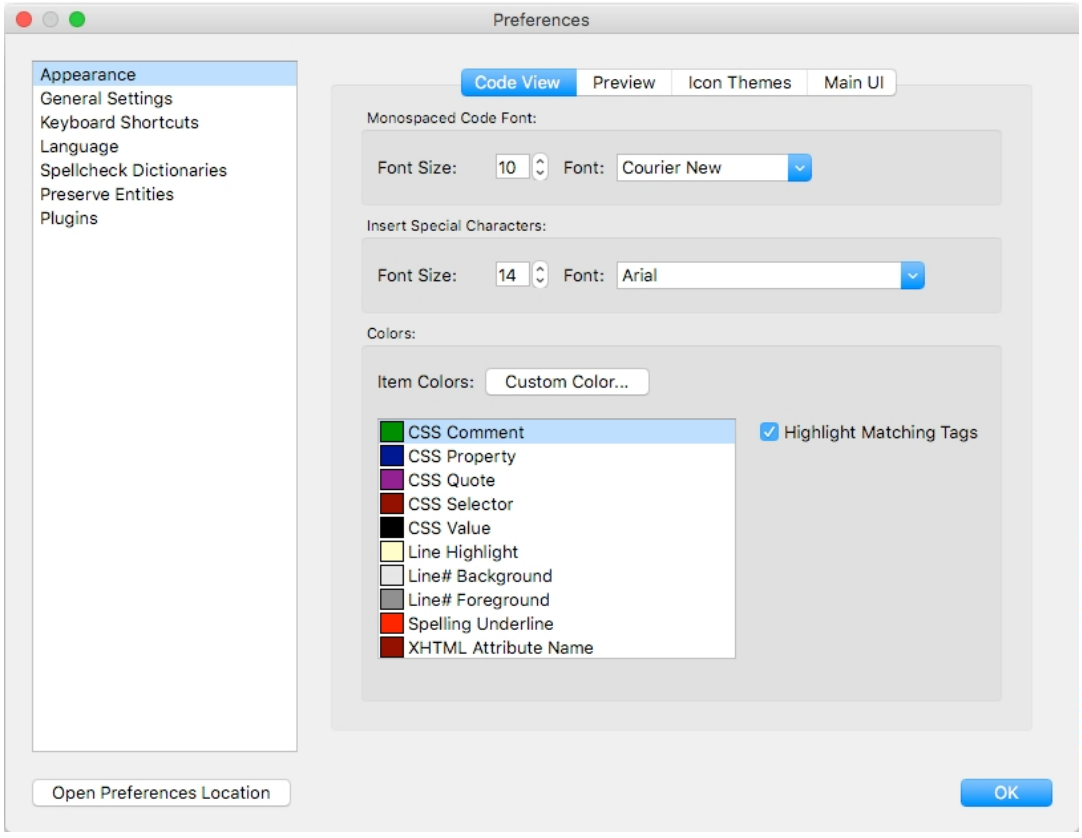
Once the Preferences window is open, select the section you want from the list on the left. Some of the sections are further divided into tabs that run across the top. Once you are satisfied with your settings, click **OK** or close the window to save the current settings.

Hover your mouse over the various preference options to see if they reveal a helpful tooltip.

Appearance

Sigil offers many options for how things appear on your screen. Appearance preferences are divided into four clickable tabs running across the top of the dialog window: Code View, Preview, Icon Themes, and Main UI.

Code View

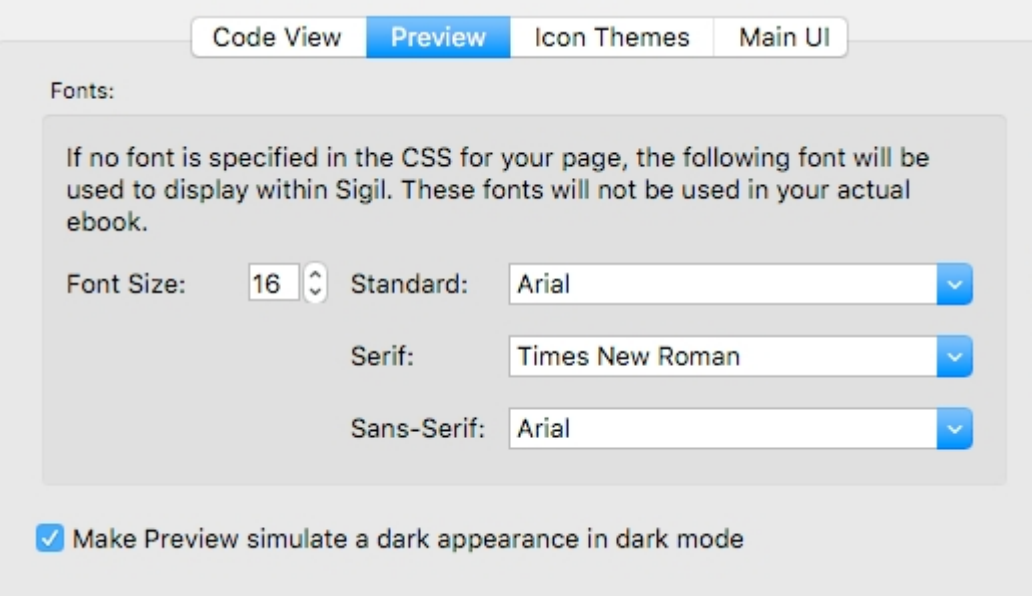


- **Monospaced Code Font**: Select the font size and font family used for text in the Code View window.
    - Sigil's Code View uses monospaced system fonts installed on your computer. If there is a monospaced font you would like to use that is not offered in the dropdown window, make sure it is installed on your computer.

- **Insert Special Characters**: Select the font size and font family used in the Insert Special Characters window.

- **Colors**:
  - Select the color of HTML and CSS items.
  - Select the color for highlighting text, the underline for misspelled words, etc.
  - Choose whether matching tags are highlighted when the cursor is inside a tag.

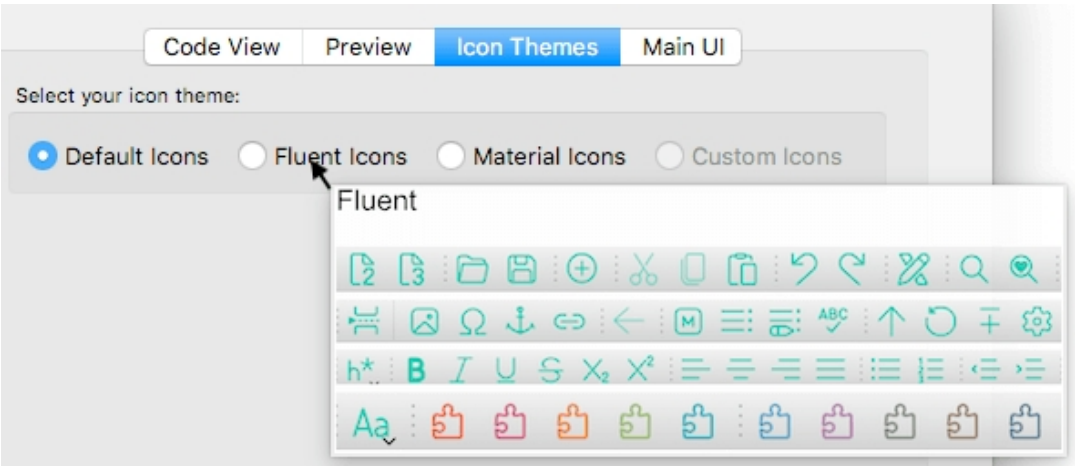
Preview



- **Fonts**: Select the default font size and font family for text shown in the Preview window when none is specified in the CSS stylesheet. ("Standard" sets the overall default font.)

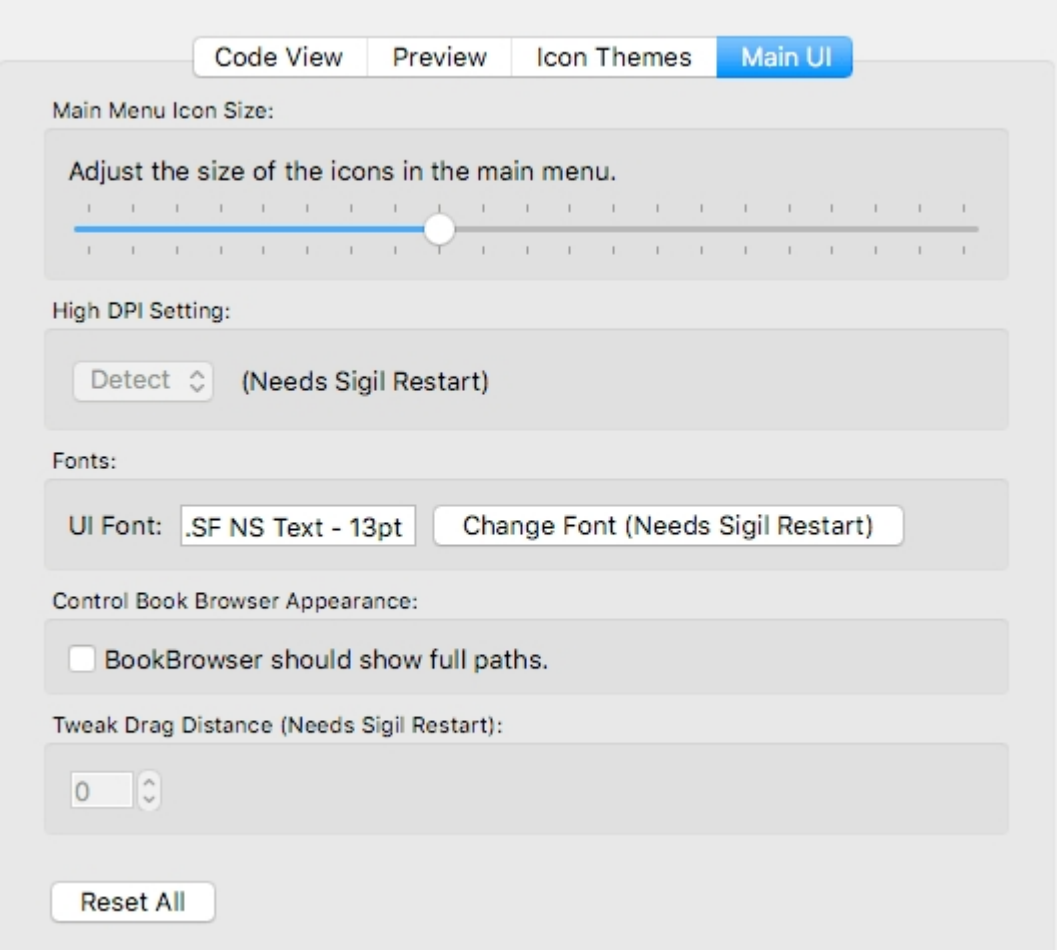
- **Make Preview simulate a dark appearance in dark mode**: Choose whether the Preview window simulates a dark appearance when your computer is in dark mode.

Icon Themes



- **Select your icon theme**: Sigil offers three different icon themes: Default, Fluent, and Material. Hover your mouse over the theme name to see a sample.
    - Sigil also supports custom user-supplied icon themes via a custom .rcc file. Check out https://github.com/Sigil-Ebook/IconThemes for more information.
    - Restart Sigil after changing your icon theme for them to take effect.

Main UI



- **Main Menu Icon Size**: Adjust the size of the toolbar buttons using a slider.

- **High DPI Setting** (Windows only):
  - **Detect**: Detect whether any high dpi scaling should take place. This setting defers to any Qt environment variables that are set to control high dpi behavior.
  - **On**: Turn on high dpi scaling and ignore any Qt environment variables that are set controlling high dpi behavior.
  - **Off**: Turn off high dpi scaling regardless of if any Qt environment variables controlling high dpi behavior are set.

- **Fonts**: Select the font family and font size used for Sigil's menus, buttons, etc.

- **Control Book Browser Appearance**: Choose whether the Book Browser shows full filepaths or abbreviated ones.

- **Tweak Drag Distance** (Windows only): Adjust the distance (in pixels) necessary to drag an item before a move event is triggered. This is useful if you are accidentally moving tabs with a twitch of the mouse.

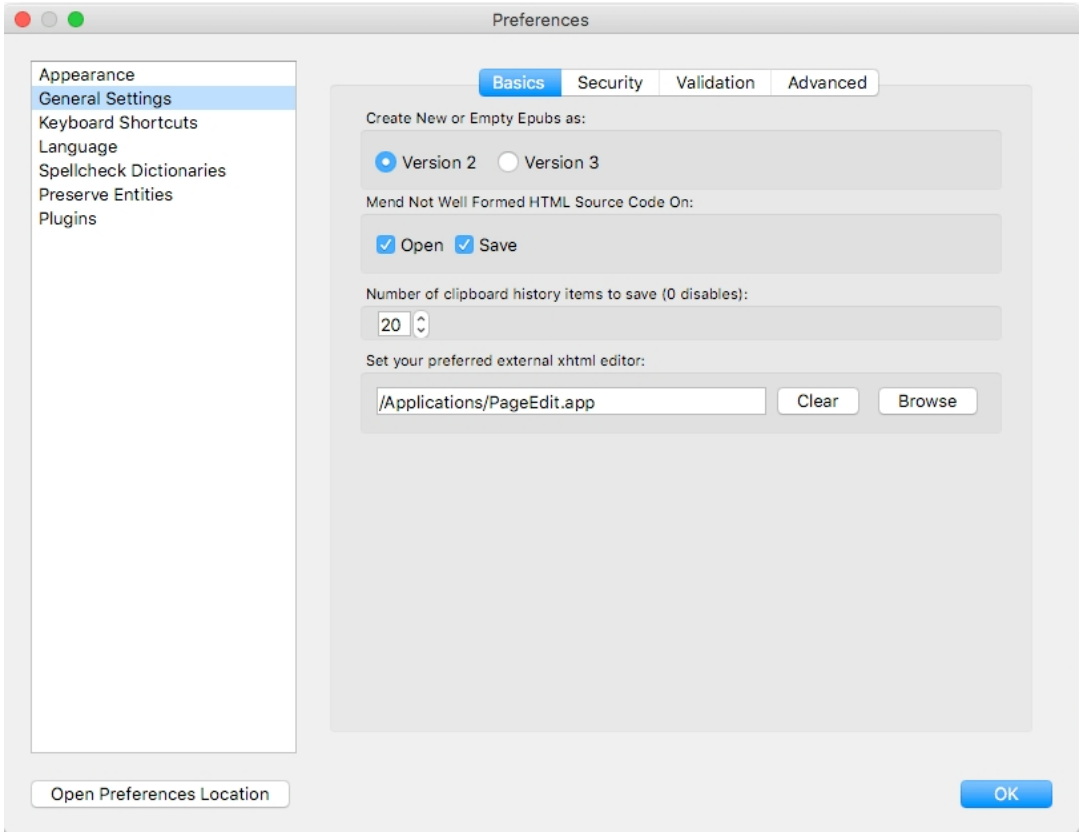
- **Reset All**: Reset all Main UI Appearance preferences to default.

> Appearance settings affect how your book is displayed in Sigil. They do NOT affect how your book is displayed on e-readers!

General Settings

General Settings are divided into four clickable tabs running across the top of the window: Basics, Security, Validation, and Advanced.

Basics



- **Create New or Empty Epubs as**: Choose the version of EPUB (2 or 3) that will be used when new books are created in Sigil.

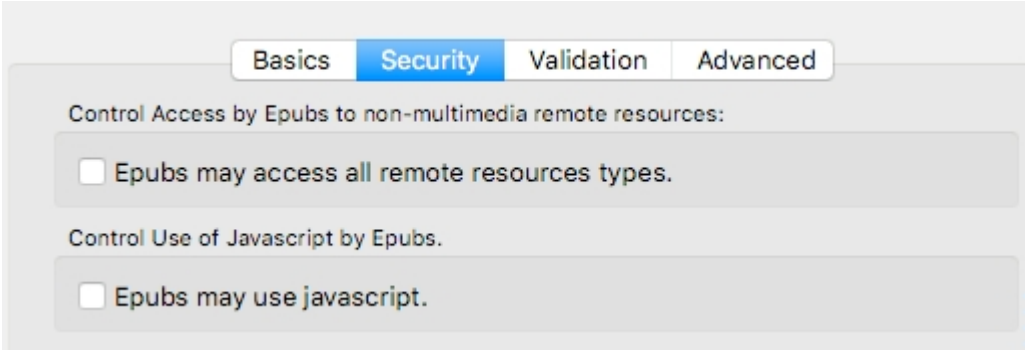
- **Mend Not Well Formed HTML Source Code On**: When checked, Sigil will automatically mend HTML source code that is not well formed or are missing a DOCTYPE, html, head, or body element whenever you Open or Save a book. Mending is done using Google's gumbo auto-repair parser. If

ill-formed code is detected, a confirmation window will appear asking you to confirm whether you want to proceed with mending.

- **Number of clipboard history items to save**: Specify the number of clipboard snippets that will be saved to a list for later recall; or specify 0 to disable clipboard history.

- **Set your preferred external xhtml editor**: If you set a preferred external xhtml editor, it will launch whenever you click the Launch External Xhtml Editor toolbar button . This allows you to quickly and easily open and edit selected html files using an external editor (such PageEdit, notepad++, or BBEdit) with the click of a button.

Security



- **Control Access by Epubs to non-multimedia remote resources**: Choose whether to allow EPUBs to access non-multimedia remote resources

- **Control Use of Javascript by Epubs**: Choose whether to allow EPUBs to use javascript.

Validation

- **EPUB2 W3C Stylesheet Validation Level**: Choose the CSS level that the W3C stylesheet validation tool will use for EPUB2 books.
- **EPUB3 W3C Stylesheet Validation Level**: Choose the CSS level that the W3C stylesheet validation tool will use for EPUB3 books.

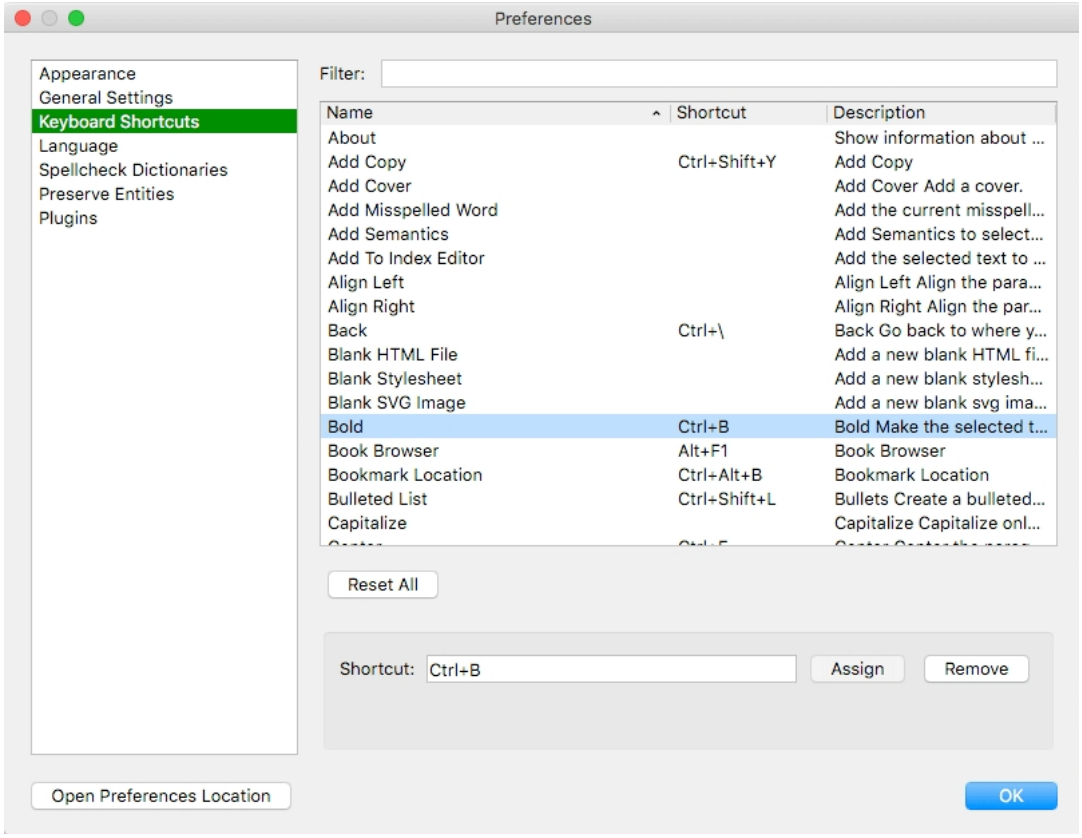
Advanced

- Set the temp folder that Sigil will use for your temporary EPUB files.
    - If you're using the macOS version of Sigil, you should select a folder other than **/temp** to avoid potential data loss issues that might occur in some macOS versions.

Keyboard Shortcuts

Sigil offers a variety of keyboard shortcuts, which you can customize to your liking. For example, if you want F12 to always open Find & Replace, you can change it here.

> You can assign or re-assign a shortcut temporarily if you need a function to be easier to access for a short time.



To assign or remove a shortcut:

- Select an entry from the list of commands.
- Click in the Shortcut text box.
- Press the key combination you want to use – do not spell it out using letters, just press the key combination.
- Click **Assign** to set the shortcut. If the key combination is already being used for another command, an alert message showing the conflict will appear.
- To remove a shortcut, select an entry and click **Remove**.
- Use **Reset All** to change all shortcuts back to their defaults.

Use the Filter text box to easily search for shortcuts.

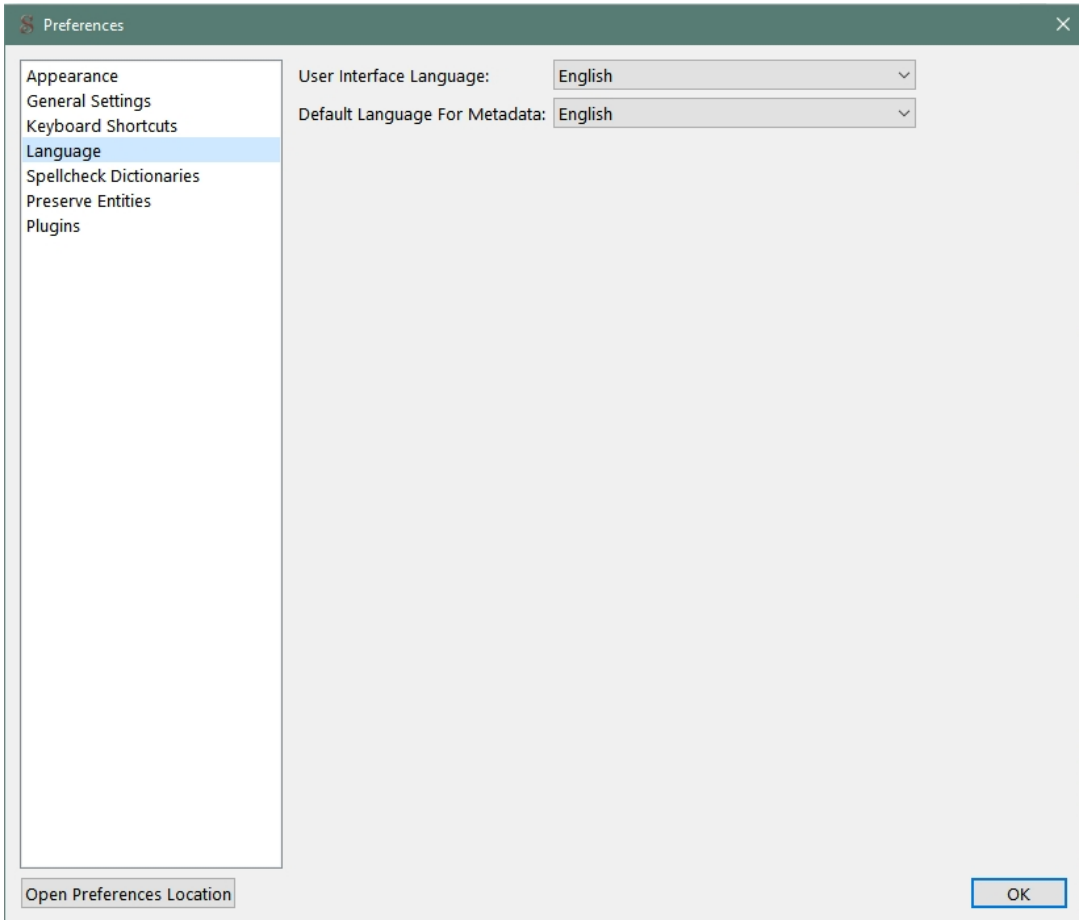
Note to Mac users: References to the Ctrl key in Sigil correspond to the Cmd key on the Macintosh keyboard, and references to the Meta key correspond to the Ctrl key on Mac.

This is due to Qt's automatic remapping, which allows each platform's most common shortcut key sequences to work as expected. For example, when Windows/Linux users press Ctrl+C, they get the Copy command, as they expect;

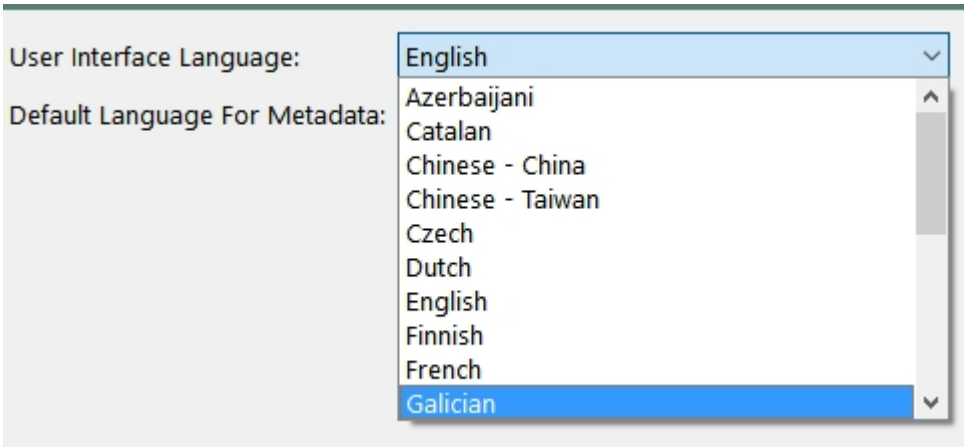
and thanks to Qt's automatic remapping of Ctrl to Cmd on macOS, when Mac users press Cmd+C, they get the Copy command, as they expect.

Language

Sigil has been translated into many languages.



- **User Interface Language**: Choose your preferred language for Sigil's user interface. If a translation is not available for specific words, they will be shown in English.

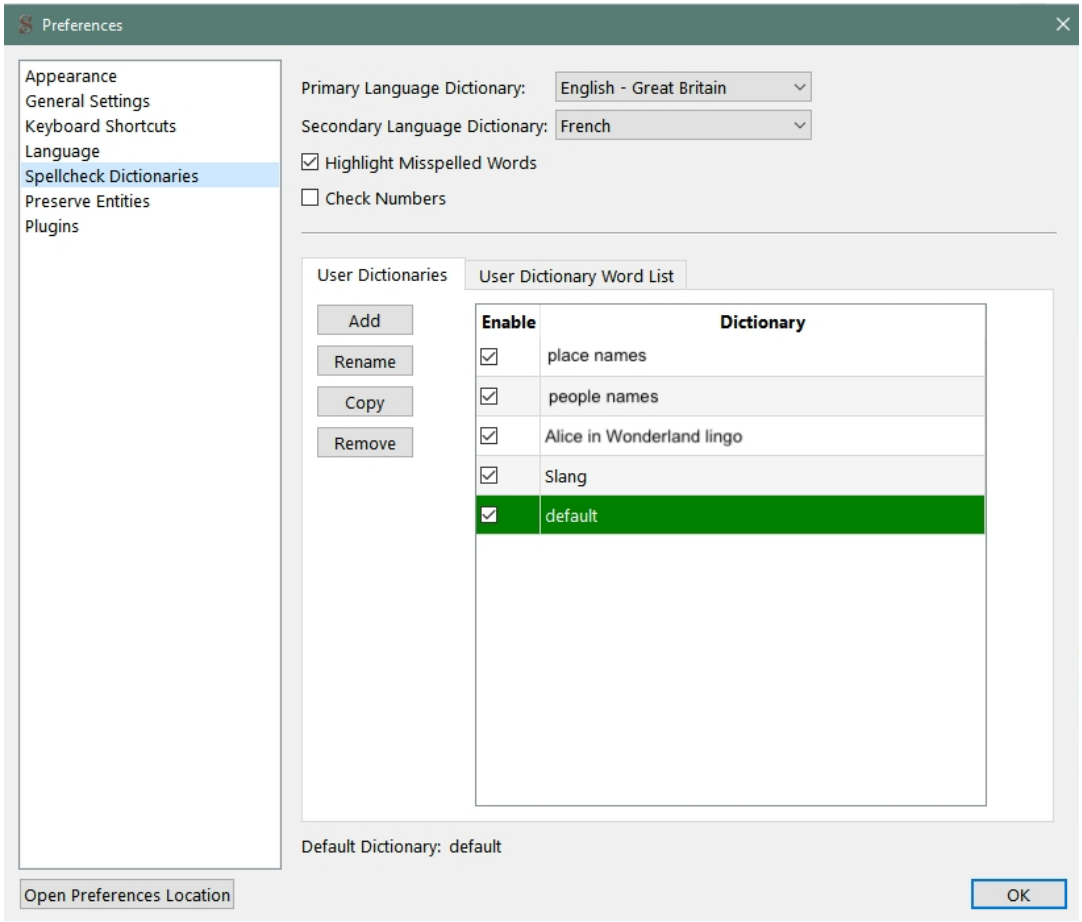


- **Default Language For Metadata**: Set the default language that will be used by the Metadata Editor for new books or books that do not already have a language set.

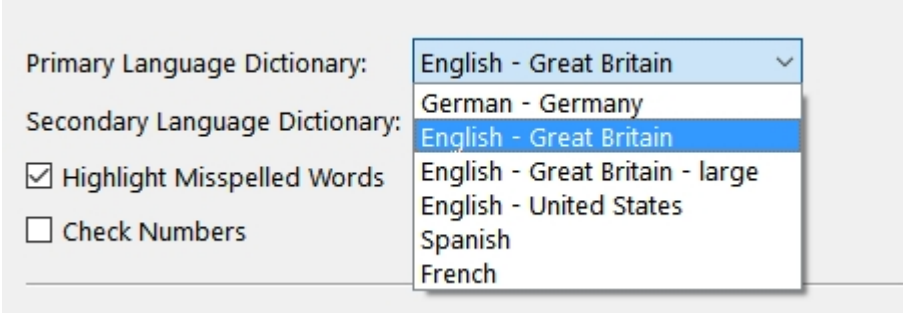
> You must restart Sigil in order for a new user interface language to take effect.
>
> Try setting the interface to another language and restarting Sigil to see the translations volunteers have created.

Spellcheck Dictionaries

Sigil supports spellchecking and offers a variety of spellchecking options, including a choice of standard dictionaries and the use of personal user dictionaries.



- **Primary Language Dictionary**: Choose your primary language dictionary from a dropdown menu.



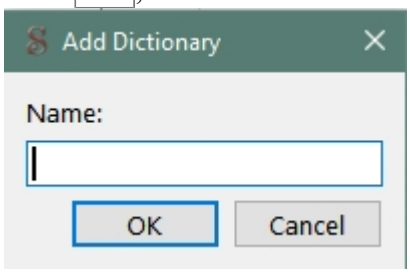
  - To add new standard dictionaries (which must be in the hunspell format), click **Open Preferences Location** and go into the "hunspell_dictionaries" directory. Put your new dictionary files into the "hunspell_dictionaries" directory then restart Sigil.
  - One source for additional dictionaries is the OpenOffice Dictionaries site. To add these dictionaries, download, extract, and copy the files (e.g. en_GB.aff, en_GB.dic, hyph_en_GB.dic) to the "hunspell_dictionaries" directory location and restart Sigil.

- **Secondary Language Dictionary**: Choose your secondary language dictionary.

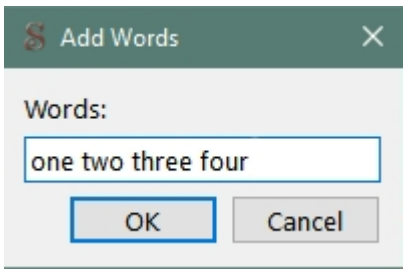
- **Highlight Misspelled Words**: Choose whether misspelled words are highlighted in Code View with a wavy red underline. You will then be able to right-click on the word to select a suggested correction or add it to a dictionary.

- **Check Numbers**: Enabling this option will allow words that have numbers in them, such as 32nd or HTML5, to be checked for spelling.

- **User Dictionaries**:
    - You can create your own personal user dictionaries, and even use different ones for different books and different types of words (e.g., places, names, etc.).
    - If you have not created any user dictionaries, Sigil will create and enable a "default" user dictionary for you, which you can rename or delete.
    - To create a new user dictionary, click on the User Dictionaries tab, click **Add**, then enter a name for your new dictionary:



    - You can see the words in each User Dictionary by selecting the dictionary under the User Dictionaries tab, then clicking on the User Dictionary Word List tab. You will then see all the words in that dictionary, which you can add to, edit, or remove.
    - To add words directly to a user dictionary, select the dictionary from the User Dictionaries tab, click on the User Dictionary Word List, then click **Add**. Type in as many words as you want, or paste them in from a file. Words can be separated by commas, spaces, or listed one per line.



    - You can also add words to a dictionary using Spellcheck, or by right-clicking on misspelled words in Code View and selecting one of the **Add To Dictionary** options.
    - You can copy a complete dictionary using **Copy**, and then rename it as needed.

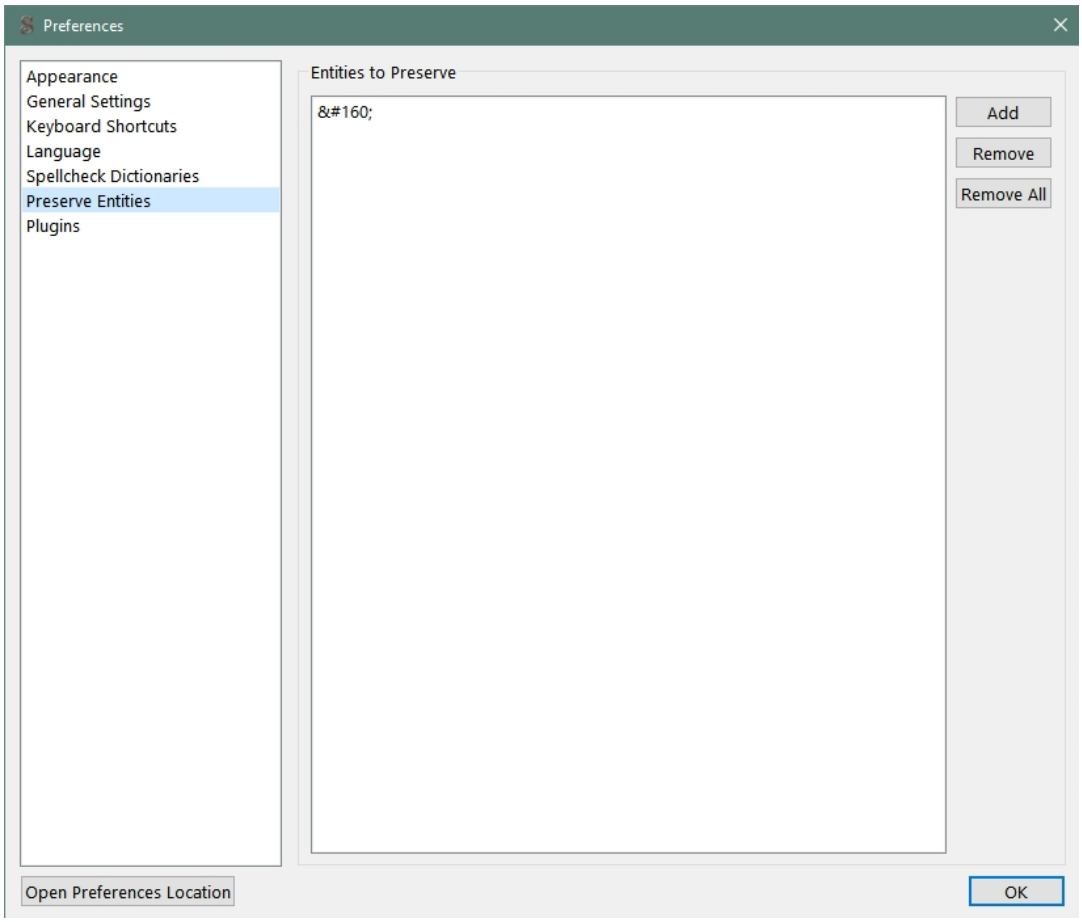
- You can select which user dictionaries are enabled by marking the checkbox next to the dictionary's name. Multiple dictionaries can be enabled at the same time. All enabled dictionaries will be used when checking words to see if they are misspelled.

The **default dictionary** is the one that is selected and highlighted. It is also identified at the bottom left corner of User Dictionaries section of the Preferences window. The default dictionary is used when you add words using the Add Misspelled Word shortcut or the context menu item **Add To Default Dictionary**.

An **enabled dictionary** is marked with a checkmark to the left. While only one user dictionary can be selected and set as the *default* at any one time, mulitiple dictionaries can be *enabled*. All enabled dictionaries are used when spellchecking is run.

Preserve Entities

When you run Mend (**Tools=>Reformat HTML=>Mend All HTML Files**), Sigil automatically converts all character entities to Unicode characters *unless* they are included in the Entities to Preserve list.



- To preserve specific entities and prevent Mend from converting them to Unicode characters, click **Add**, and add them to the Entities to Preserve list. Multiple entities can be input at one time when separated by spaces, line breaks, or commas.

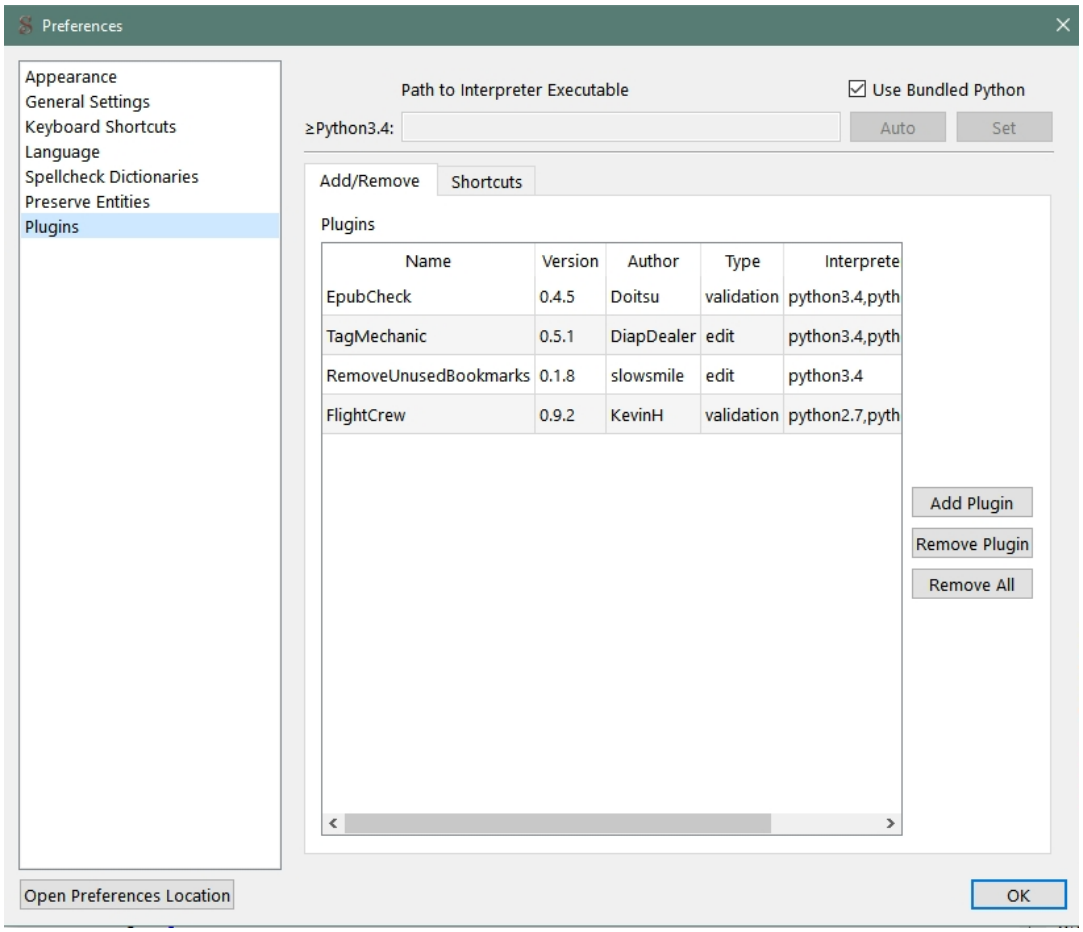
- Use **Remove** or **Remove All** to remove entities from the list. Entities not on the list will be converted to Unicode characters on Mend.

Please note that you can't use named entities (e.g. &nsbp;) in EPUB3 books; you must use numerical entities (e.g.,) instead.

If you have Mend Not Well Formed HTML Source Code turned on for Open or Save, and the source code is well formed, entities will ***not*** be automatically converted (because entities are perfectly valid code). To convert entities in html files with well formed code, you must manually invoke **Tools=>Reformat HTML=>Mend All HTML Files**.

PLUGINS

Sigil supports a wide variety of useful plugins that augment Sigil's editing power.



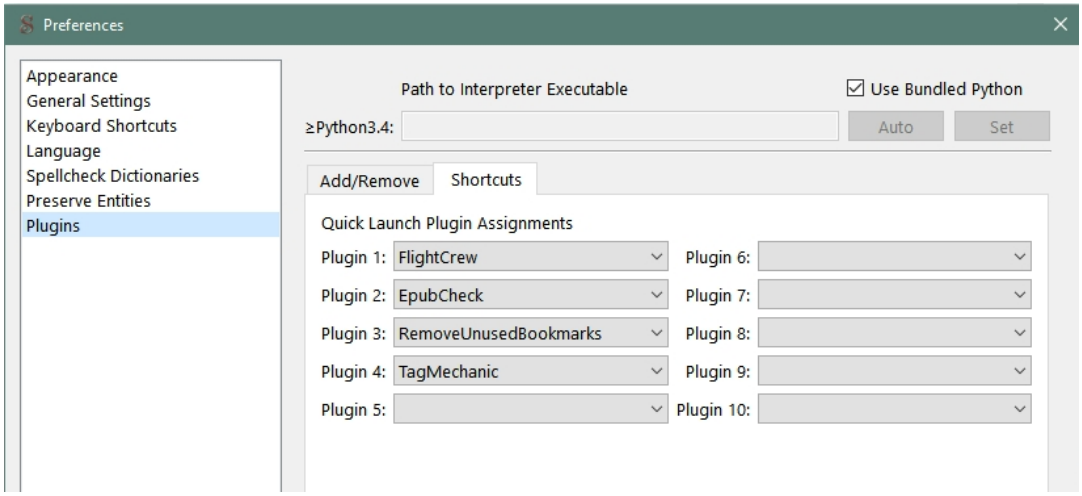
- **Use Bundled Python:** Enable this option to select the Python interpreter bundled with Sigil (recommended). Please note that if you disable this option, you will need to select an external Python 3 interpreter.

- **Path to Interpreter Executable:** This option is intended for plugin developers; you'll also need to set it if Use Bundled Python is disabled (or can't be selected). Click **Auto** or **Set** to choose an external Python 3 interpreter.

- **Add/Remove**
  - **Add Plugin:** Click this button to install a plugin.
  - **Remove Plugin:** Click this button to uninstall the currently selected plugin.
  - **Remove All Plugins:** Click this button to uninstall all plugins.

- **Shortcuts**
  - Assign plugins to the 10 plugin buttons on the toolbar. When you click on a plugin button, the assigned plugin will launch automatically.

Toolbar buttons:



Open Preferences Location

The **Open Preferences Location** button at the bottom left corner of all Preferences screens will open the directory containing all of your preferences files. You can copy the files for backup or for sharing with others.

The preferences location will likely be in the following locations:

Windows:

```
…\Local Settings\Application Data\sigil-ebook\sigil
```

Mac:

```
/Users/[Username]/Library/Application\ Support/sigil-ebook/sigil/
```

Linux:

```
$HOME/.local/share/sigil-ebook/sigil
```

The files/folders you may see include:

- **custom_preview_style.css**: Optional user CSS for Sigil's Preview window.
- **hunspell_dictionaries**: Where any standard dictionaries you add are stored.
- **local-devtools**: Where data for Preview's Inspector is stored.
- **local-storage**: Where JavaScript local storage data is stored.
- **plugins**: Where any plugins you install are stored.
- **plugins_prefs**: Where plugin preferences files are stored.

- **qt_styles.qss**: Optional custom Qt styles for Sigil GUI elements.
- **QtWebEngine**: Where data for Preview is stored.
- **repo**: Where checkpoint versions are stored.
- **sgc-toc.css**: Optional custom TOC stylesheet for EPUB2 books.
- **sgc-nav.css**: Optional custom TOC stylesheet for EPUB3 books.
- **sigil_clips.ini**: Your saved Clips.
- **sigil_index.ini**: Your Index entries.
- **sigil_searches.ini**: Your Saved Searches (Sigil 1.5.0 and earlier).
- **sigil_searches_v2.ini**: Your Saved Searches (Sigil 1.6.0 and later).
- **sigil.ini**: General configuration information for Sigil.
- **user_dictionaries**: Where any personal user dictionaries are stored.

Change the Font Size of GUI Controls

Since Sigil is a Qt app, you can customize the font size of some GUI controls by creating a **qt_styles.qss** stylesheet file and placing it in the Sigil preferences folder. (**qt_styles.qss** needs to contain valid Qt style sheet declarations.)

For example, the following styles will increase the font size of the fields in the Find & Replace window:

```
QLabel { font-size: 16pt }
QLineEdit { font-size: 16pt }
QComboBox { font-size: 16pt }
```

Customizing the Background of Preview

Since Preview is technically a browser showing rendered (x)html/CSS, the only way to alter font and/or background colors is by specifying that info via styles: either inline or via CSS from within the EPUB.

The only exception to the above is that a custom user-specified CSS file will be applied to Preview (no matter what EPUB is open) if a file named **custom_preview_style.css** exists in the Sigil preferences folder and contains valid CSS.

For example, the following styles will give Preview a light gray background with blue text:

```
body {
  background-color: #D3D3D3;
  color: blue;
}
```

Note: Applying such CSS can possibly adversely affect the visibility of some styled elements within an EPUB, depending on what CSS has already been applied via the EPUB and how.

Opening and Saving Files

— Load and Save Your Books —

Sigil supports opening EPUB files and reading HTML and text files. When Sigil saves your book it saves in the EPUB format: the EPUB file will contain all of your HTML files, images, stylesheets, fonts, etc. zipped up into one file.

> Save early, save often, and make backup copies!
>
> Always close Sigil before you open your EPUB in another program – or the other program will lock the file and prevent you from saving in Sigil.

Opening and Adding Documents

Open Files

To open a file, click the Open button , or select the menu item **File=>Open** or its keyboard shortcut then choose the file you want to open:

EPUB Files

- Everything in the EPUB file is loaded automatically: HTML files, images, stylesheets, metadata, etc..
- Sigil will unpack your files into the specified folders into exact paths shown in the Book Browser via tooltips. The Book Browser itself groups files according to their type or function in the EPUB.
- If the EPUB contains common errors Sigil will try to correct them.
- Depending on your preferences, Sigil will format or clean the HTML files.
- If the EPUB still contains invalid HTML files after cleaning, Sigil will still load the file and report the errors that need to be fixed.

XHTML Files

- The XHTML file and any linked files and stylesheets are imported if possible and placed into their folders.
- The metadata in the XHTML file will be loaded into the metadata for the book.
- Depending on your preferences, Sigil will format or clean the files.
- If the XHTML file is still invalid after cleaning, Sigil will report the error but may not load the file. You may have to fix the errors outside of Sigil.

Text Files

- The text file will be converted to XHTML. Paragraphs must be separated by blank lines to have them split into separate paragraphs.

Add Files

Add files to your EPUB by clicking the Add Existing Files button , selecting the menu item **File=>Add=>Existing Files**, or right-clicking on a file in the Book Browser and choose **Add Existing Files**:

XHTML Files

- Metadata is not loaded from the file (unlike when you use Open).
- Sigil will attempt to load files that are referenced.
- Files will be placed into their appropriate folder.

Image Files

- Files will be placed into the the current images folder.

Video Files

- Files will be placed into the current video folder.

Audio Files

- Files will be placed into the current audio folder.

Stylesheet Files

- Files will be placed into the current styles folder.

Font Files

- Files will be placed into the current fonts folder.

Other Files

- All other files will be shown under the Book Browser's Misc group as they are not usually used in an EPUB, although certain text files in the Misc group can be edited in Sigil.

Add Blank Files

You can also create new empty files in Sigil. Select the menu item **File=>Add** and choose from blank HTML page, blank CSS stylesheet, or blank SVG image (a special image format). You can also do this using this Book Browser, or even add a keyboard shortcut in Preferences.

Saving Your EPUB

Save your book by clicking the Save button , or selecting the menu item **File=>Save** or its shortcut. You should always save your document before doing major edits in case you want to undo them.

Your saved EPUB file contains all the files that make up your book – your text, images, stylesheets, embedded fonts, table of contents, metadata, etc.

> If you edit any text in your document, an asterisk (*) is added to the file's name in the titlebar.

When you save your book Sigil will check it for certain errors and warn you if you need to fix anything. But you should always run one of the Validators available as plugins for Sigil, such as FlightCrew for EPUB2 and EpubCheck for EPUB2 and 3, to check for typical EPUB issues.

> Sigil will not always be able to save an EPUB that contains errors, so frequent saves will make it easier to identify if there are errors that you need to correct.
>
> Sigil will also clean your source code according to your preferences when saving the file.

Save a Copy

You can select **File=>Save A Copy** to save a copy of your EPUB to a different filename. The difference between Save or Save As is that the current book you are working on is not renamed. So you can use this feature to save your EPUB without changing the name of your book or the directory you save files to.

> Use **File=>Save A Copy** for quick backups.

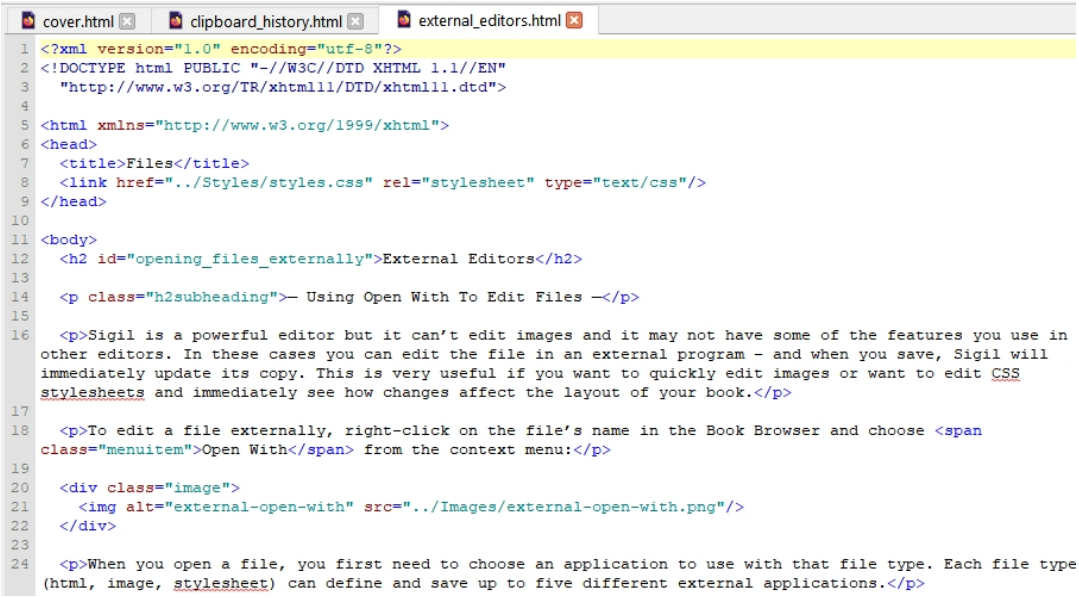
Saving Individual Files

You can also right-click on a file name in the Book Browser and select **Save As** from the context menu. You can then name and save the specific file to a directory on your computer. If you select more than one file, all files will be saved to your chosen directory. This is a quick way to share a specific file with someone without sharing the EPUB, or to create a template file you can use for other books.

Code View

— Direct XHTML Editing —

Code View allows you to directly edit the XHTML and CSS code included in your EPUB and is the most common way to edit files for advanced users.



> Use the Preview window to display a read-only book view version of the page you are editing in Code View.

Code View is very powerful, but you must be careful when editing it to ensure you enter valid XHTML or CSS code.

In addition to displaying your XHTML code exactly as it is in your EPUB, Code View highlights tags and elements in colors to make it easier to read and identify issues.

> You can change the colors used for highlighting in Code View using Preferences.

Formatting

All of the buttons on the formatting toolbar are available for you to use.

Context Menu

Right-click on text to see a context menu which includes the following options when editing XHTML:

- **Clips:** Select from any text Clips you have defined.
- **Add To Clips:** Save the selected text to your Clips.
- **Go To Link Or Style:** Same as **Search=>Go to Link or Style**.
- **Mark Selected Text:** Same as **Search=>Mark Selected Text**. May also show Unmark Marked Text.
- **Reformat XHTML:**
  - **Mend and Prettify Code:** Mends and nicely formats the current file.
  - **Mend And Prettify – All HTML Files:** Mends and Prettifies all HTML files.
  - **Mend Code:** Modifies the current file to make it valid XHTML if it isn't already.
  - **Mend Code – All HTML Files:** Modify all HTML files to make them valid XHTML if they aren't already.

Right-click on a misspelled word to see these options:

- ***suggestions*:** Select from suggested alternatives.
- **Add To Default Dictionary:** Save the word to the default dictionary selected in Preferences.
- **Add To Dictionary:** Save the word to the dictionary you select from the submenu.
- **Ignore:** Skip spell checking this word until Sigil is restarted.

Right-click on an image to see these options:

- **View Image:** Displays the image in a resizable window. You can also Ctrl Click on the filename or use the Go To Link Or Style shortcut. View Image will resize the image to fit the window, and can remain open while you edit.
- **Open Tab For Image:** Opens the image file in a tab at full resolution.

When editing CSS, the context menu includes the following options:

- **Rename Selected Class:** When a class names has been selected, you can rename it in this stylesheet and all impacted XHTML files as well.
- **Reformat CSS:** Reformat the css to either single or multiple lines.

Plus the usual options:

- Undo
- Redo

- Cut
- Copy
- Paste
- Delete
- Select All

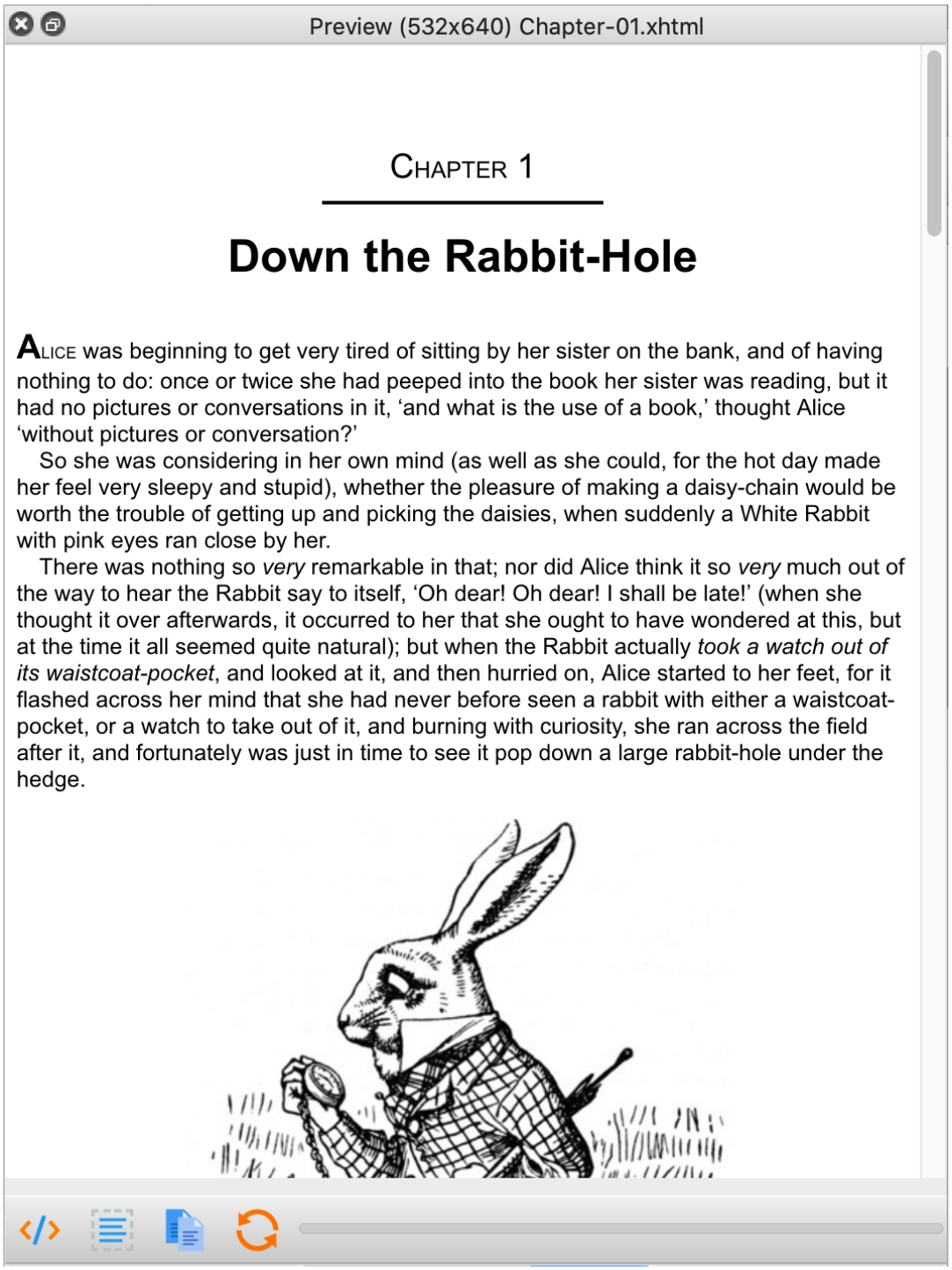
Preview

— Show Code View Changes —

Preview is a movable window that gives you a dynamically updated read-only view of what appears in Code View. You can also inspect your code and styles by clicking the Inspect button (</> icon) in the toolbar at the bottom of Preview.

You can use Preview to show the results of changes in your code (including immediate feedback of errors), and you can also open a CSS stylesheet to make changes that are shown in Preview.

Activate the Preview window by selecting the menu item **View=>Preview** or its keyboard shortcut:



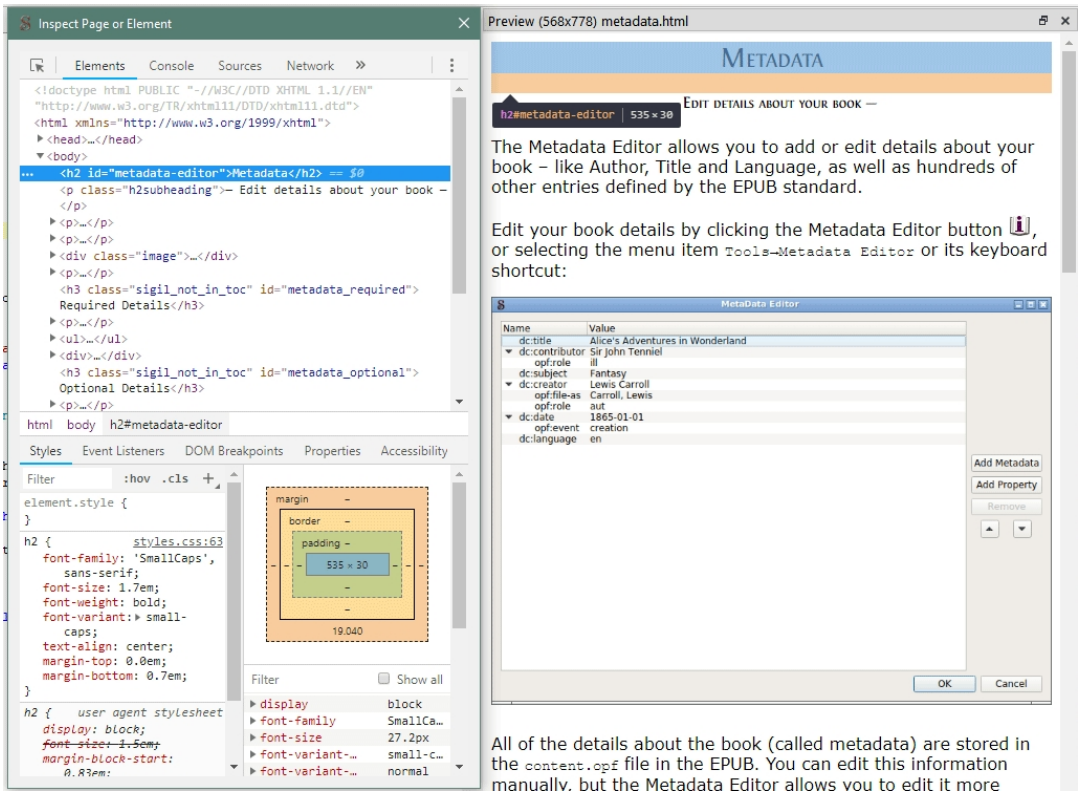
You can toggle the Preview window open or closed with its shortcut, and even move the window to a second monitor if you have one.

The Preview window will be updated immediately when you click somewhere in Code View. It will also be updated after you stop typing for a short time.

If you click in Preview the Code View tab will sync to the same location.

INSPECTOR

The Inspector Widget is opened/closed by clicking the "</>" button on the toolbar at the bottom of Preview. You can resize and reposition it how you like, and Sigil will remember your settings



The Inspector window is an HTML developer's tool that allows detailed examination of the code behind the book's appearance. It is intended to allow you to see why specific sections of your book are displayed the way they are, and to make temporary changes to styles and text to see how they affect your text.

Any changes you make in the Inspector window are only temporary and are not saved!

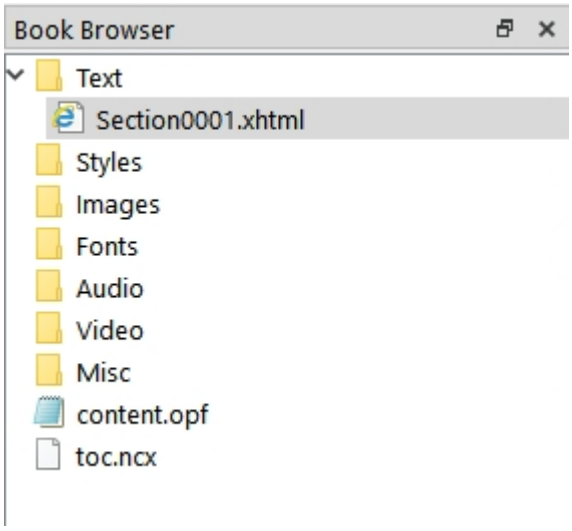
For more information about using the Inspector built into Preview and specifically how to use it to determine the exact CSS being used for any particular tag, see this tutorial.

Book Browser

— Managing Files in Your EPUB —

All of the files that make up your EPUB can be viewed and accessed using the Book Browser.

You will find the Book Browser docked on the left side of the default user interface. You can control whether it is visible or hidden via menu item **View=>Book Browser**. You can also drag it to a different location if you prefer.



Click on the little arrow or plus sign to the left of the folder's name, or double-click on the folder, to expand or collapse it.

Double-click on a file to open it in a tab in the Code View window. If it was already open, the view will switch to the tab containing the file. The Book Browser will always highlight the active file/tab displayed in the Code View window.

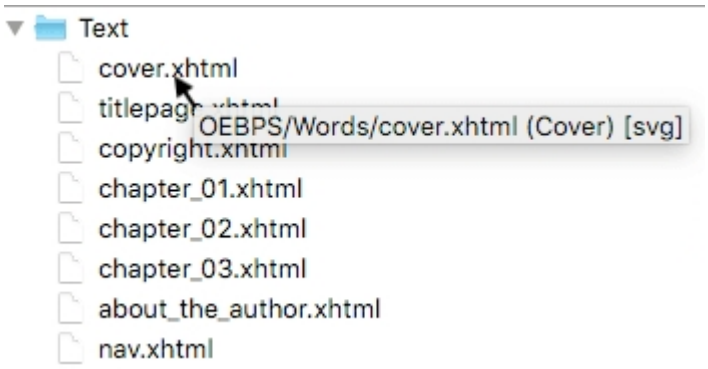
Right-click on a filename or folder to display a context menu of options.

Folders & Files

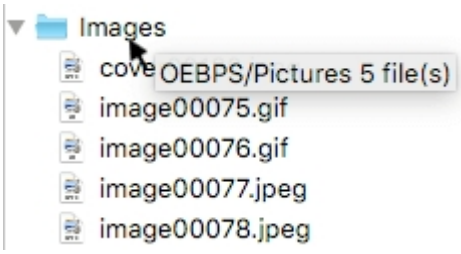
The Book Browser organizes your files into several virtual folders by file type (text, stylesheets, images, fonts, etc.). It is important to keep in mind that starting with Sigil 1.0, folders and their contents are displayed in this manner for ease of editing/multi-selection. *The Folder text/name does not reflect the actual folder name of the actual folders inside your EPUB.* For example, a nav.xhtml file will be

displayed in the *virtual* Text folder with all the other XHTML files, even if it's in a different *actual* folder inside the EPUB.

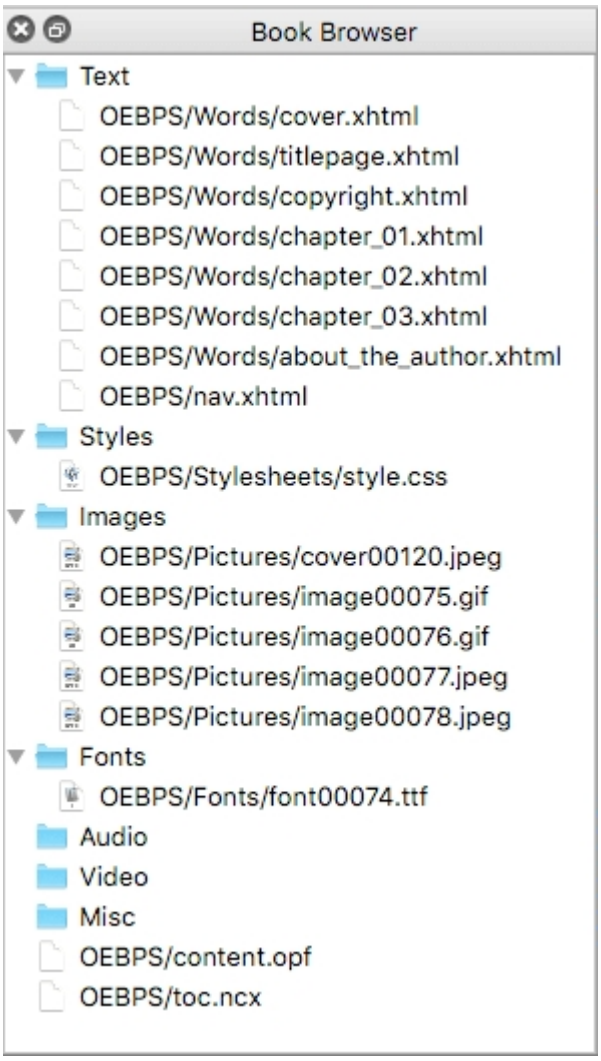
To see the full filepath and actual folder names, hover your mouse over a file. If the file has a guide semantic type or manifest property, it will also be revealed.



Hovering your mouse over a folder shows the actual folder name, full filepath, and number of files in the virtual folder.



To always show full filepaths in the Book Browser, go to **Edit=>Preferences=>Appearance=>Main UI** and select "Book Browser should show full paths".



The virtual folders and the file types they contain are:

- **Text:** The XHTML, HTML, and, HTM files that make up the text of the book.

- **Styles:** The CSS stylesheets that control the formatting of the book.

- **Images:** The PNG, JPEG, GIF, SVG, and all image files in the book.

- **Fonts:** The TTF, OTF and any other embedded fonts that might be in the book.

- **Audio:** The MP3, AAC, MPG, MPEG, M4A and any other embedded audio files that might be in the book.

- **Video:** The MP4, OGG, WEBM, M4V and any other embedded video files that might be in the book.

- **Misc:** Other files that are not typically used by EPUB readers but can be stored with the book. This includes things like javascript (JS) files under EPUB3.

- **No virtual folder:** The NCX (.ncx) and OPF (.opf) files are not placed in a virtual folder. The NCX contains a an EPUB2 readable table of contents of the book (required for EPUB2, optional for EPUB3). The OPF contains a list of files in the book (manifest). It also holds the book metadata, and determines chapter display order (spine), and some semantic data. The OPF is basically the controlling document for the EPUB.

Prior to version 1.0, Sigil automatically restructured each EPUB into a standard Sigil form. As a result, the names of the folders in the Book Browser always matched the actual folders used. Sigil 1.0 no longer does this. Sigil 1.0 loads EPUBs as they are without moving files or renaming folders, and allows you to structure your EPUBs and name your folders as you please. However, if you prefer the old standard Sigil form, you can restructure your EPUB using menu item **Tools=>Restructure Epub to Sigil Norm**.

Selecting Multiple Files

Select multiple consecutive files in a folder by clicking on the first file that you want, then hold down the Shift key while clicking on the last file. All files between and including the first and last clicked will be selected. To select (or add) non-consecutive files, hold down the Ctrl key (on Mac: Cmd key) while clicking individual files. You can also use right-click to call up the context menu and choose **Select All** to grab all of the files in a folder in one go. (Note: You can only select multiple items within the same folder (ie. of the same type).

Once you have selected your files, you can use the right-click context menu to perform actions on the selected files.

Re-Ordering Files

The order of the files in the Text folder is important since it is the order in which the files will be listed in the OPF (.opf), which is the order they will be displayed to readers.

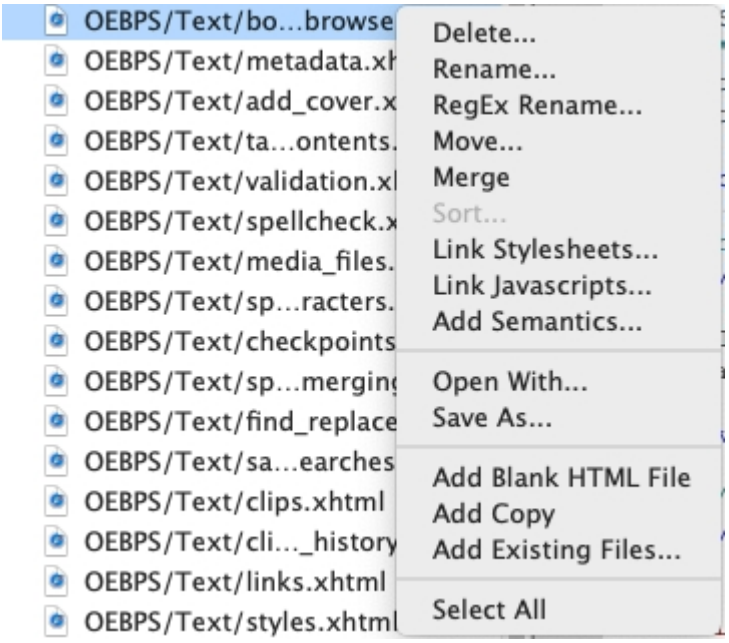
Files in the Text folder can be dragged up or down, allowing you to re-order files quickly and easily. When you change the order of these files in the Book Browser, the OPF is automatically updated. However, the NCX and/or EPUB3 NAV (nav.xhtml) are *not* updated automatically. For more on updating the table of contents files, see the Table of Contents chapter.

> You can also use the context menu to rename the files.

Files in all other folders are automatically sorted alphabetically and cannot be rearranged. The order of their files does not affect how your book is displayed.

Context Menus

Right-clicking on one or more items in the Book Browser brings up a context menu with relevant actions. Different context menus will appear depending on whether you're clicking on a text file, stylesheet, image, folder, etc.

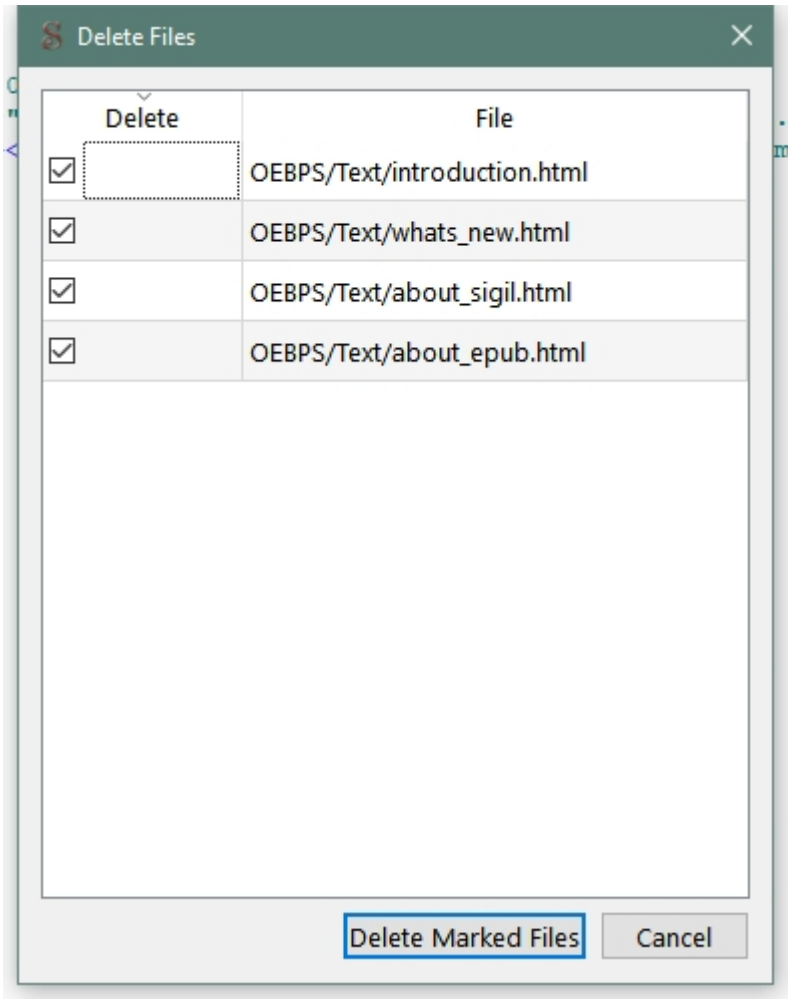


Available context menu commands include:

Delete

Delete the selected file(s). (You cannot delete the last remaining HTML file.)

You will be prompted to confirm that you want to delete the files since the action cannot be undone.

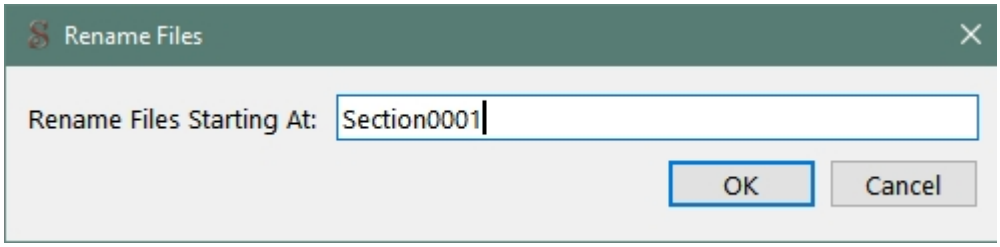


Remember to update or regenerate the Table of Contents if your file was included in the TOC.

Rename

If you select a single file to rename, the name of the file will be highlighted so that you can type in a new name. By default the extension is not highlighted to make renaming easier, but you can also change the extension if necessary.

If you select multiple files, a dialog is displayed asking you how you want to rename the files:



Enter a new name for the first file, including a starting number at the end of the name. Sigil will rename each of your files sequentially using the name you provided and starting at the number you provided. A file will not be renamed if it would result in the book having duplicate file names. You can also rename the extension.

For example, if you start with the original names: Section0001.html, Section002.html, Section0003.html … then:

chapter_01
: will rename the files to: chapter_01, chapter_02, chapter_03 … and keep the .html file extension.

Section_0099
: will rename the files to: Section_0099, Section_0100, Section_0101 … and keep the .html file extension.

Part2-Chapter08
: will rename the files to: Part2-Chapter08, Part2-Chapter09, Part2-Chapter10 … and keep the .html file extension.

.xhtml
: will rename only the file extension of all files to .xhtml without changing the base filename.

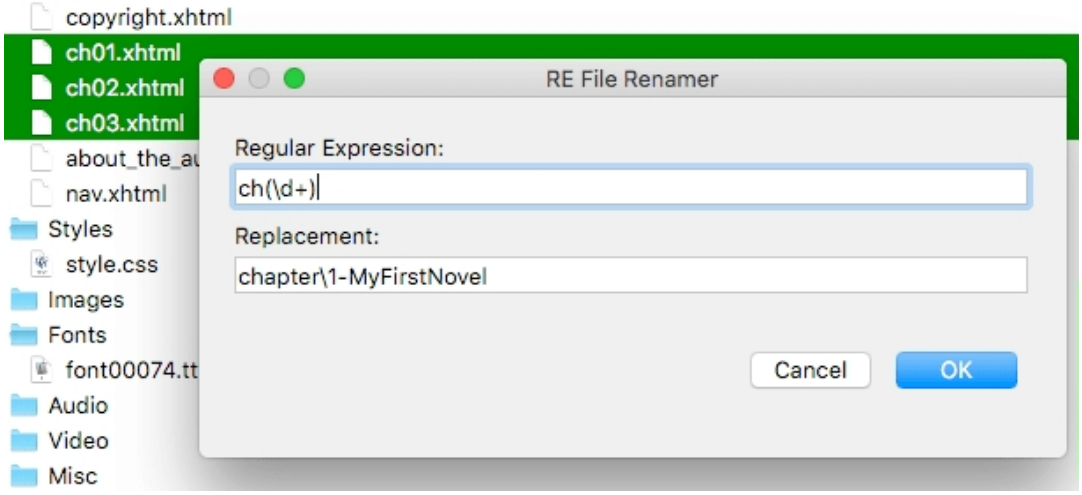
chap01.xhtml
: will rename the files to: chap01.xhtml, chap02.xhtml, chap03.xhtml ….

> When you rename a file, any links pointing to the file in your EPUB will be automatically updated.

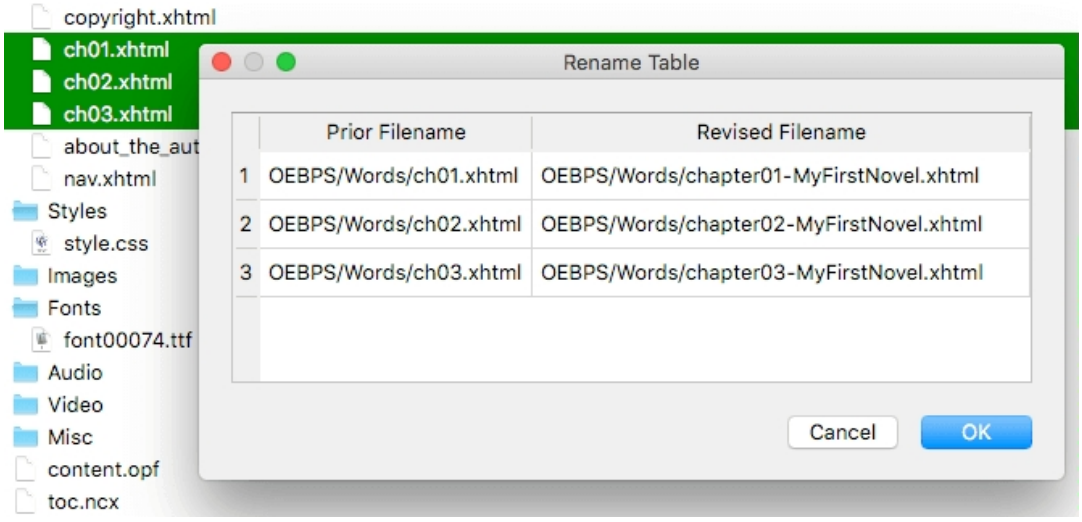
ReGex Rename

A more powerful renaming tool, ReGex Rename allows you to use regular expressions to find and replace text in filenames.

Select the files you want to rename, right-click to choose **ReGex Rename**, and the RE File Renamer dialog box will appear. Input the regular expression you are searching for and the text you want to replace it with, then click **OK**.



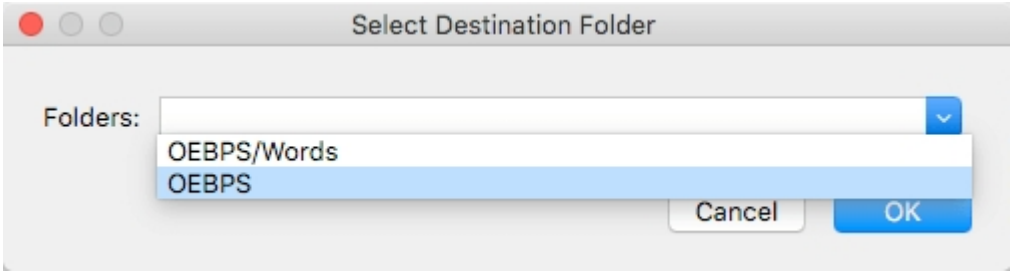
The Rename Table will show you the old and new filenames. If the new names are acceptable, click **OK** to rename your files.



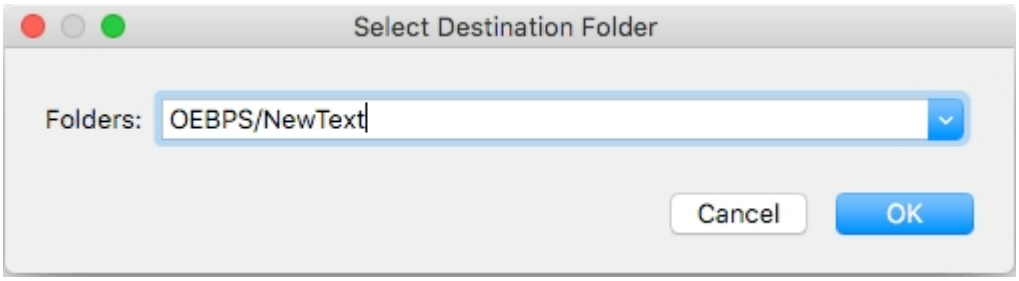
Move

Move the selected files to a different folder.

When you invoke **Move**, a dialog box opens and offers a dropdown menu of existing folders to choose from.



You can also type a new name directly in the text box to create a new folder to move your files into.



Press **OK** to execute the move.

Merge

If you select a single HTML file, then the file will be merged into the previous HTML file in the list.

If you select multiple HTML files, then all files will be merged into the first HTML file you selected.

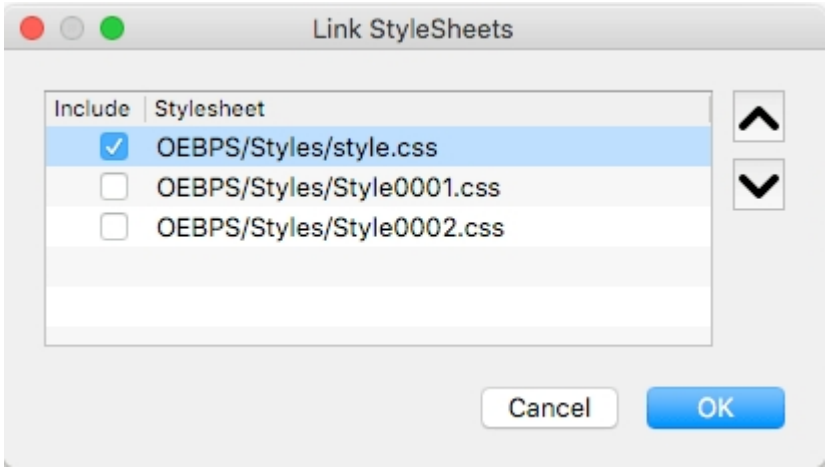
You can also use the Merge keyboard shortcut. See the Splitting and Merging chapter for more details.

Sort

Sort the selected HTML files alpha-numerically. This means filenames containing numbers will be sorted in their expected numerical order, e.g. “1, 3, 2, 10” will be resorted as “1, 2, 3, 10” rather than “1, 10, 2, 3”.

Link Stylesheets

Link stylesheets to the selected HTML files. Choose the stylesheets from the dialog window. Only stylesheets marked with a checkmark will be linked; unchecked stylesheets will be unlinked.



When the dialog is first opened a checkmark is shown against stylesheets only if all selected files are linked to that stylesheet.

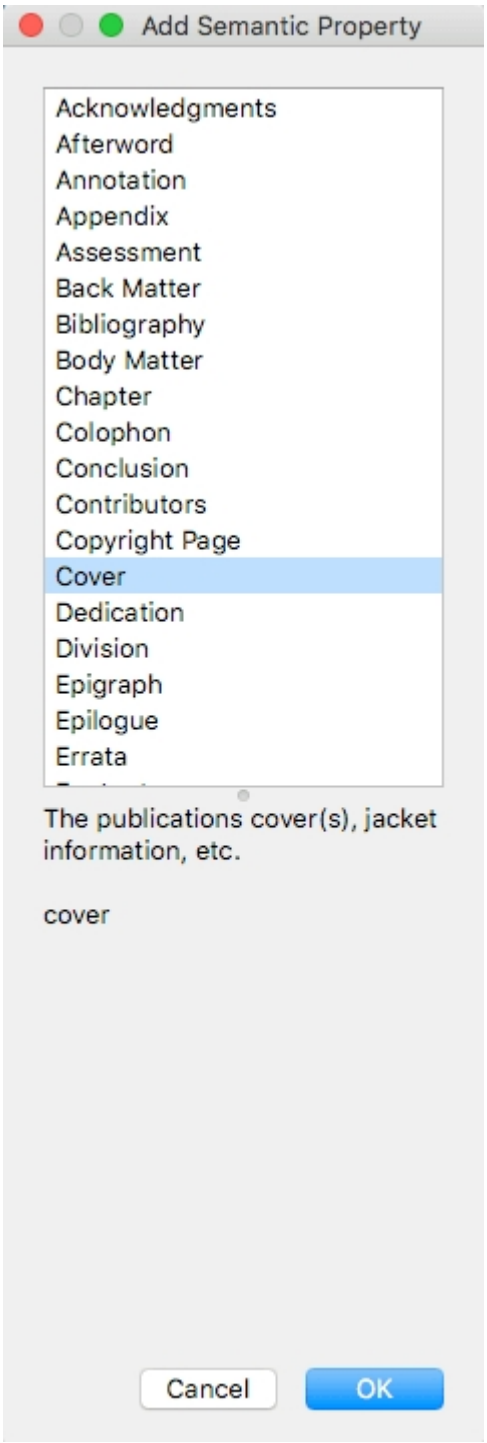
Stylesheets are applied in the order listed when the document is rendered.

Link Javascripts

Link existing javascript files to the selected HTML files (EPUB3 Only). Choose the javascript files from the dialog window. Only javascripts marked with a checkmark will be linked; unchecked javascript files will be unlinked.

Add Semantics

Add Semantics allows you to designate the semantic type for selected files from a scrolling list of choices. EPUB ereaders may use this information (stored in the .opf file in EPUB2 and .opf and nav files in EPUB3) to guide them in how to present these files to readers. Each file can have only one semantic type.



Text (EPUB2) or **Body Matter** (EPUB3) should be used to indicate the page of the book that should be shown when the book is first opened, though not all ereaders follow this convention.

Open With

Open the file in an external application, such as a photo editing program or your favorite HTML editor. For more information, see the External Editors chapter.

Save As

Save the selected files to your computer. This is useful if you need to access the files outside of the EPUB.

Add Blank HTML File/Stylesheet/SVG Image

Add a blank HTML document, CSS stylesheet, or SVG image file to the appropriate folder. For HTML it will add the file just below the one you right-clicked on, or at the bottom of the folder if you right-clicked on the folder name. For other folders it will add the file in alphabetical order.

The new file will be numbered with the lowest available number starting from 0001 (e.g. Section0001.html). You can then rename it.

This context menu command is the same as the main menu item **File=>Add**.

Add Copy

Create a copy of the HTML or CSS file you clicked on and add it to the folder.

Add Existing Files

Add a file that resides on your computer outside of the EPUB.

A dialog window will open allowing you to choose one or more files from your computer to import into your EPUB. Files will be added to the appropriate virtual folder based on their file extension, regardless of what file or folder you right-clicked on.

This is the same as the main menu item **File=>Add=>Existing Files** and the **Add Existing Files** toolbar button.

Select All

A quick way to select all of the files in the current folder so you can rename, link, etc. all of them at once.

Validate With W3C (CSS stylesheets only)

If you right-click on a CSS stylesheet and select **Validate With W3C**, Sigil will open a browser window and send your stylesheet data to the W3C Validation Service, which will then show validation results for the selected stylesheet. (If an HTML editor opens instead, open the resulting HTML file in a browser. This may happen if you've set HTML files to open in an editor instead of a browser.)

Cover Image (image files only)

Add the "cover" semantic designation to the selected image.

Font Obfuscation (font files only)

If you are using custom fonts in your ebook, right-clicking on a font file will allow you to define how the embedded fonts are stored in the EPUB. Normally when fonts are added to an EPUB the fonts are stored in a readable form. In some cases you may not want to make the fonts available for readers to access directly. In these cases you can obfuscate or scramble the fonts in the ebook so that readers cannot access the fonts, but the fonts will still be useable by EPUB ereaders. The details of obfuscation and its implications are beyond the scope of this guide, but there are many references to it on the web.

The obfuscation options are:

- None
- Use Adobe's Method
- Use IDPF's Method

The method will have a checkmark next to it if all selected fonts use that method.

Renumber TOC Entries (.ncx files only)

Renumber the playOrder and navPoint id numbers of the Table of Content entries in the .ncx file sequentially, starting from 1. This feature is intended to clean up the TOC numbering if you have to manually add or delete an entry in the TOC.

You can also use the menu item **Tools=>Table of Contents=>Edit Table of Contents** or **Edit Table of Contents** button. Simply saving the TOC will automatically renumber the ids.

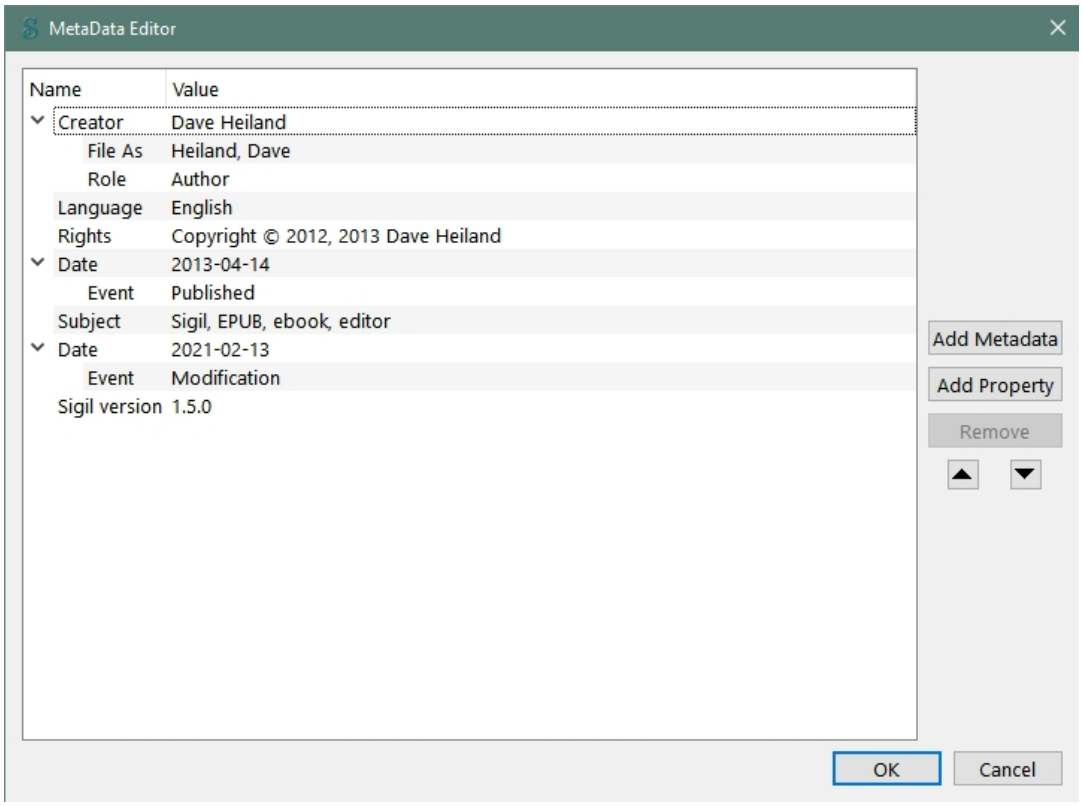
Any navList or pageList items in the file are not saved – only the Table of Contents entries are saved after renumbering.

Metadata

— Edit details about your book —

The Metadata Editor allows you to add or edit details about your book – like Author, Title and Language, as well as many of other entries defined by the EPUB standard.

Edit your book details by clicking the Metadata Editor button , or selecting the menu item **Tools=>Metadata Editor** or its keyboard shortcut:



All of the details about the book (called metadata) are stored in the content.opf file in the EPUB. You can edit this information manually, but the Metadata Editor allows you to edit it more safely and easily.

Required Metadata

There are 3 required metadata elements that you must fill in to create a valid EPUB:

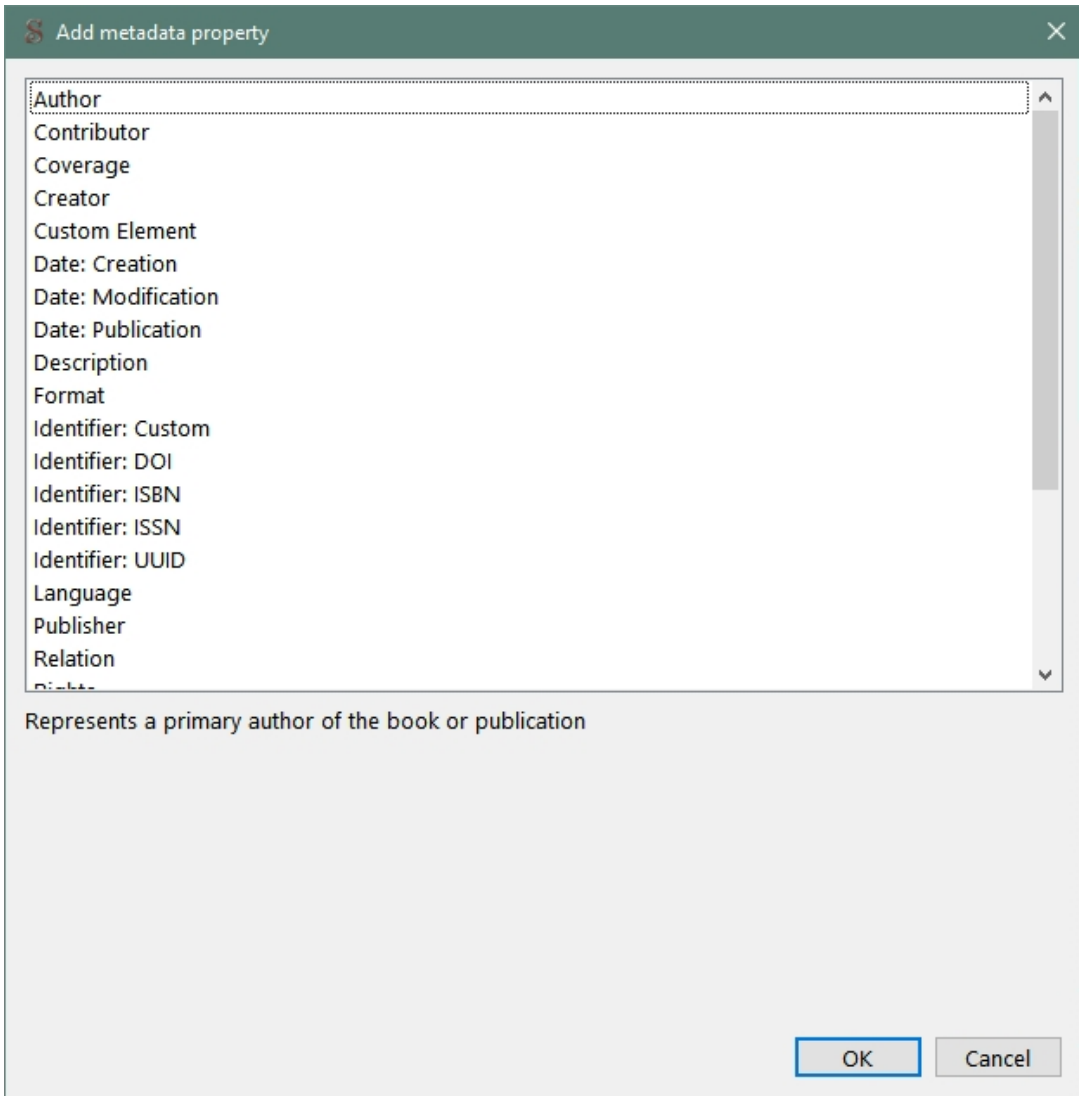
- **Title:** The main title of the book.

- **Author:** The primary author of the book, usually in the form of "Firstname Lastname". You can add more authors using the fields below.
- **Language:** The language the book is written in.

You can also add the optional **File-As** property to provide the author's name for sorting purposes, usually of the form "Lastname, Firstname".

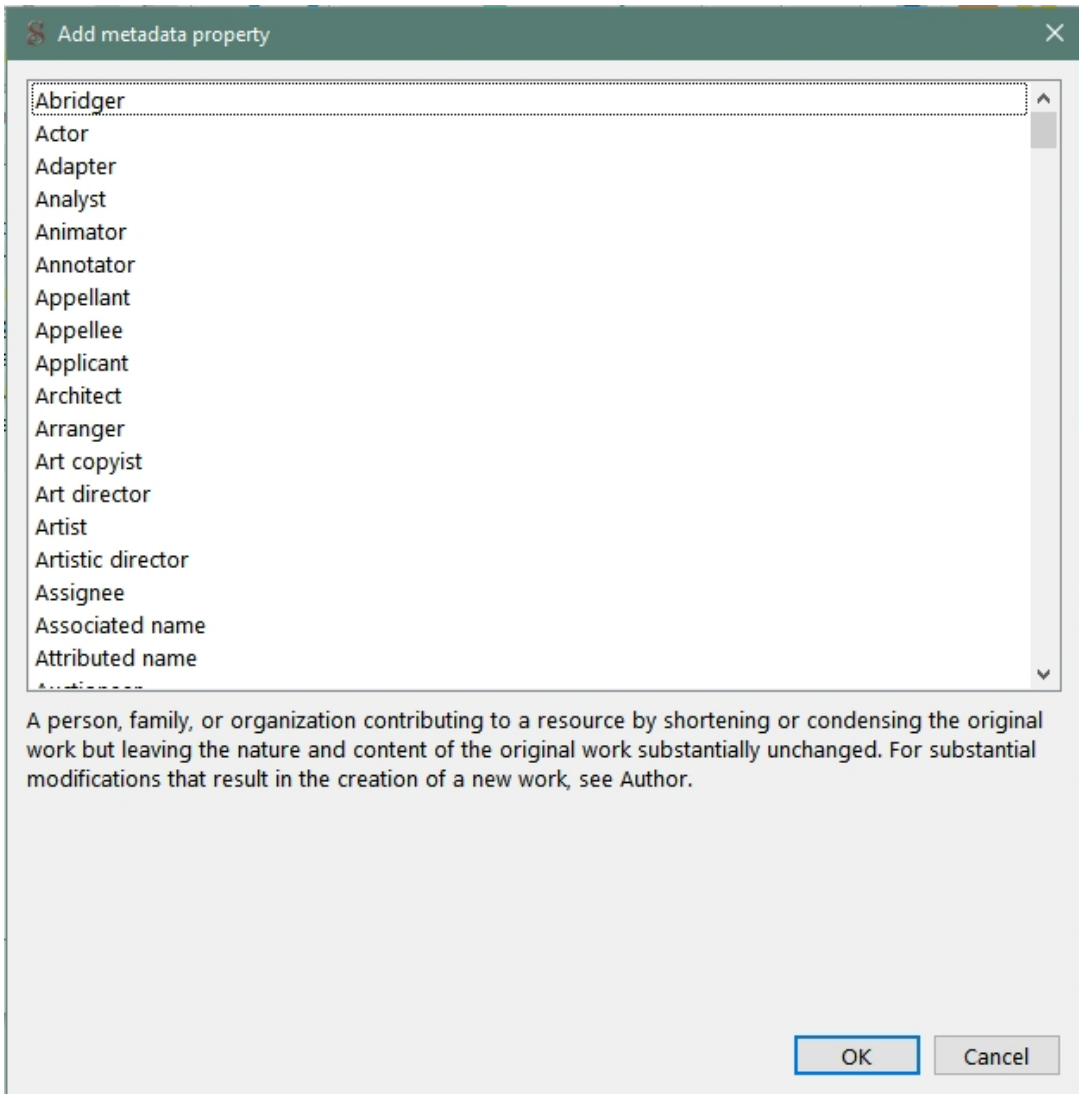
Additional Metadata

You can add additional metadata using the **Add Metadata** button to add metadata elements or **Add Property** button to add additional properties to existing metadata elements.



If you click **Add Metadata** or **Add Property**, Sigil will display EPUB-type specific metadata/property dialog items.

For example, the Role metadata property list contains hundreds of possible activities that people may have performed in the creation of this book such as Authors, Editor, Proofreader, etc.:



You can add the same type of field multiple times.

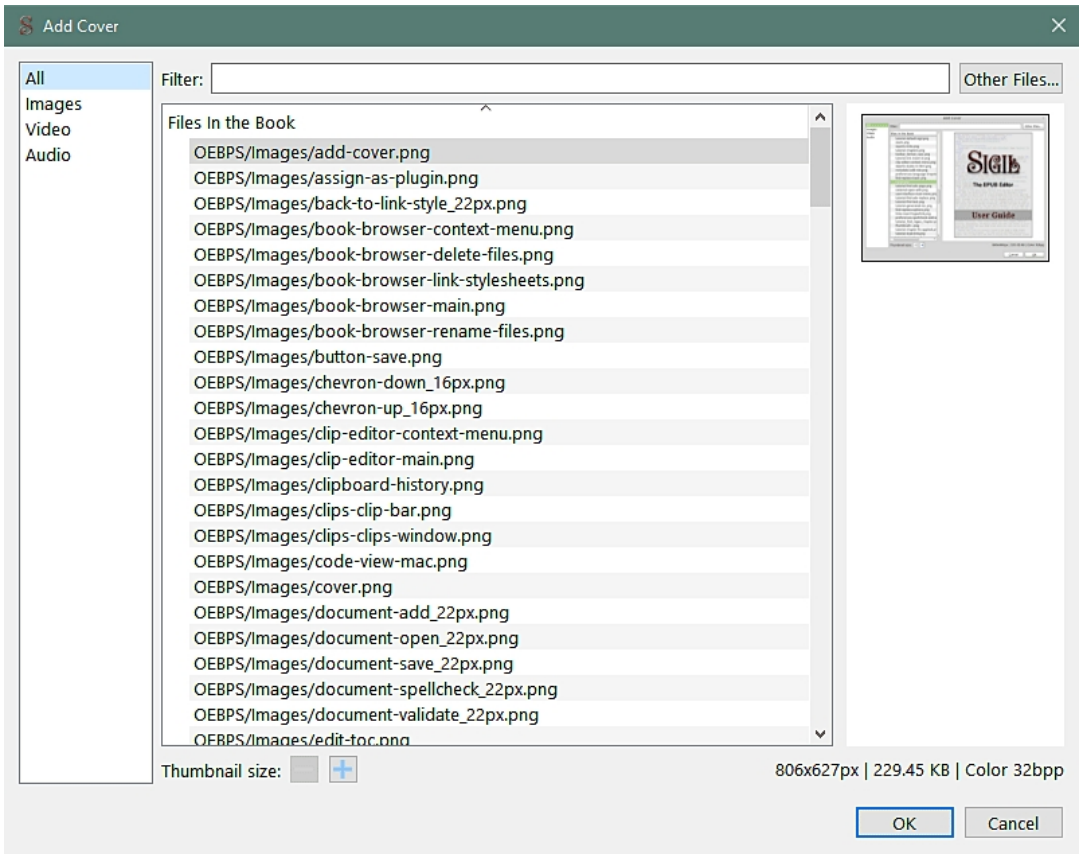
You can re-arrange the order of the entries (which may affect how e-readers display the information) by selecting the entries and clicking on the **Up** or **Down** buttons.

Add Cover

— Pick a cover for your book —

The Add Cover feature allows you to pick an image for your cover and have Sigil automatically make the necessary updates to mark the image as your cover.

Add a cover by selecting the menu item **Tools=>Add Cover** or its keyboard shortcut:



You can pick an image in your EPUB, or use **Other Files** to pick one from your computer.

Details

Sigil makes the following changes when you add a cover:

- Creates a cover.xhtml file if one does not exist. If it does exist, it will use the file. If another file with a semantic of Cover exists, it will use that file.
- Inserts the necessary HTML code to display and resize your image into the HTML file.
- Marks the HTML cover file with the semantic of Cover.

- Marks the image file with the semantic of Cover.

Custom HTML Cover File

Sigil inserts code in the HTML cover file that allows your cover to be re-sized and displayed on most e-readers, but you may want to use different code. You can edit the code after the file is created, or provide your own code for Sigil to use.

This is the HTML code Sigil inserts into the cover file by default:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Cover</title>
</head>

<body>
  <div style="text-align: center; padding: 0pt; margin: 0pt;">
    <svg xmlns="http://www.w3.org/2000/svg" height="100%"
    preserveAspectRatio="xMidYMid meet" version="1.1"
    viewBox="0 0 SGC_IMAGE_WIDTH SGC_IMAGE_HEIGHT" width="100%"
    xmlns:xlink="http://www.w3.org/1999/xlink">
      <image width="SGC_IMAGE_WIDTH" height="SGC_IMAGE_HEIGHT"
      xlink:href="SGC_IMAGE_FILENAME"/>
    </svg>
  </div>
</body>
</html>
```

To modify the template Sigil uses for the HTML cover file, create a file in your Preferences directory called cover.xhtml

You can use certain placeholders in your text and Sigil will replace them with the appropriate values:

- **SGC_IMAGE_FILENAME:** this will be replaced by the path to the image in your EPUB.
- **SGC_IMAGE_WIDTH:** this will be replaced by the width in pixels of your image.
- **SGC_IMAGE_HEIGHT:** this will be replaced by the height in pixels of your image.

Table of Contents

— A Map Of Your Book —

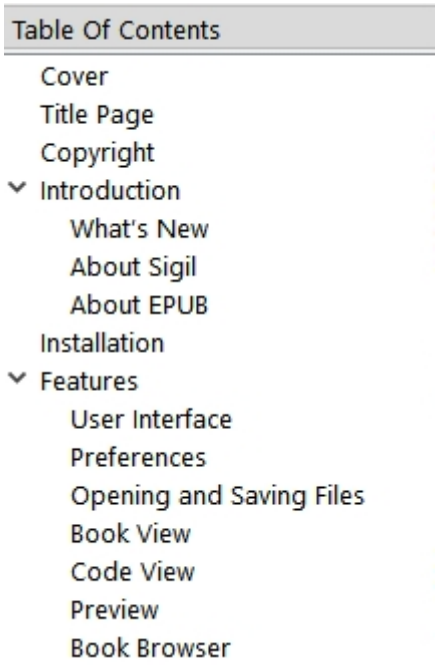
EPUB supports a Table of Contents (abbreviated as TOC) to help readers navigate their way around your book. Typically EPUB e-readers provide a button or menu item to open the TOC.

Sigil provides several ways to work with a TOC. The Table of Contents window in Sigil gives you a dynamic view of your TOC – you can view exactly what your TOC looks like, and select entries to jump to. The Generate TOC function can create a new TOC based on the headings in your document. The Edit TOC function allows you to update an existing TOC when you can't or don't want to use Generate TOC. And Create HTML TOC can create a new HTML page in your document showing what is in the EPUB TOC.

The EPUB TOC is stored in a special XML file called the NCX (typically named toc.ncx) under EPUB2 and in an XHTML document called the NAV (typically named nav.xhtml) under EPUB3. These files are designed to be machine readable so that both e-readers and accessibility software can use them to allow the user to quickly and efficiently navigate throughout the EPUB.

View the TOC

To open the TOC window if it isn't already open select the menu item **View=>Table of Contents**:



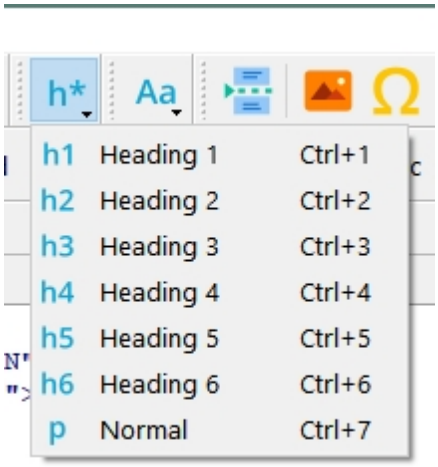
You can single-click on an entry to jump to its location in the book, and expand or collapse entries by clicking on the icons to the left of the chapter names if there are sub-chapters.

> Use the right-click context menu to select **Collapse All** or **Expand All** to quickly show or hide sub-chapters.

Generate Table of Contents

In order to create a Table of Contents automatically you first need to define the headings or chapters in your book.

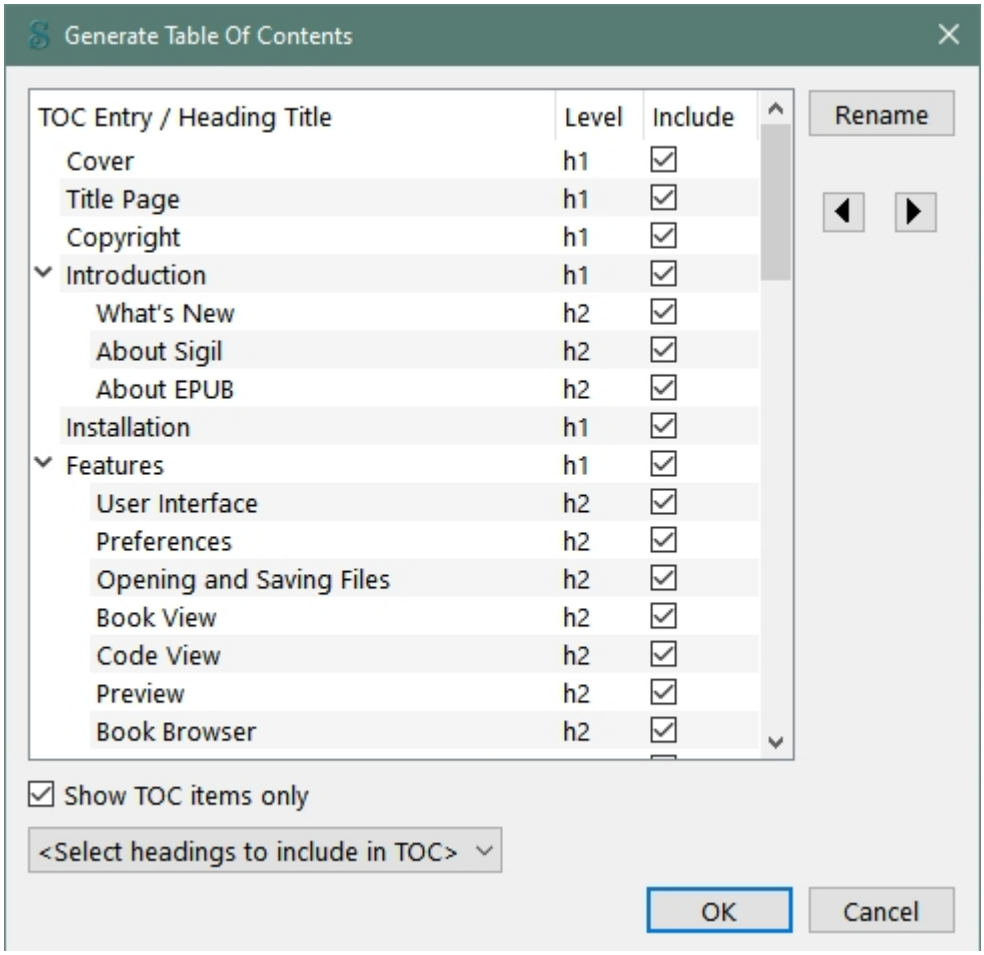
In Code View you can create the your headings by selecting the text you want to make into a heading, and then selecting the level of the heading you want from the **Format=>Heading** menu. Or more easily by using the formatting buttons on the toolbar.



> If your chapter names do not use heading tags (`h1`, `h2`, etc.) as described above, then you will not be able to generate a TOC automatically. As heading tags are very important for Accessibility reasons you should strongly consider adding them.

Start your TOC generation by clicking the Generate Table of Contents button , or selecting the menu item **Tools=>Table Of Contents=>Generate**

Table Of Contents then select the headings you want included in the book's Table of Contents:



Check any name that you want to be included in the Table of Contents, and uncheck any entry that you don't want included.

> When you use Generate Table Of Contents Sigil will update not only your TOC, but also your HTML documents – to make sure that the information you enter can be used again when you next generate the TOC.

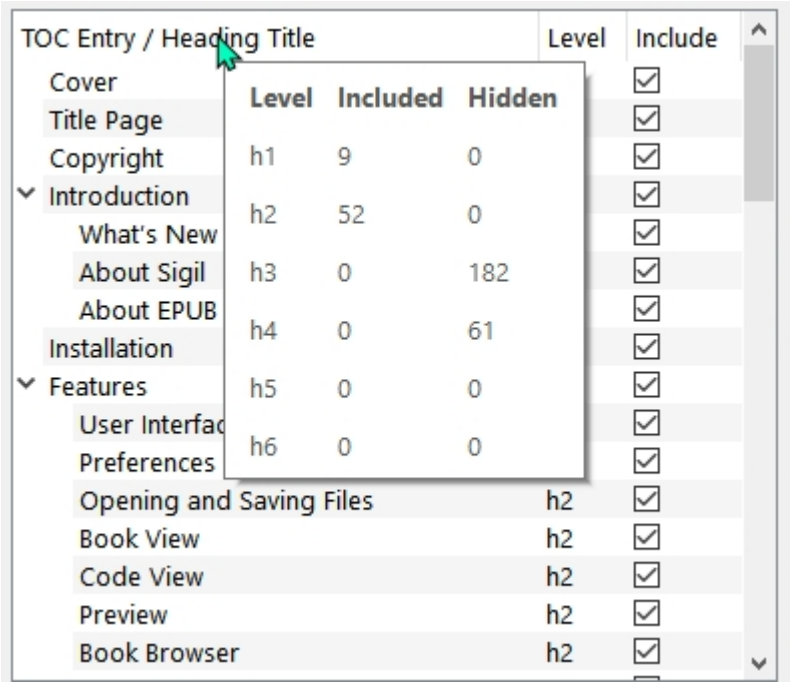
You can use the **Show TOC Items Only** checkbox to show or hide entries that will not be included in the TOC. This does not affect your TOC, only what you see in the dialog. You can also use the **Select headings to include in TOC** drop-down menu to quickly select or unselect headings of a certain type, and then fine tune the selections manually.

You can rename the TOC entry by clicking **Rename**. This will also change your HTML code to set the "title" attribute of your heading.

You can change the level of your Headings using the left and right arrow keys. This will also change the level of the headings in your HTML code.

> The tooltip for each title (just move your mouse over the title and leave it for a few seconds) will show you which file the heading is in and its HTML code.

The tooltip for the headings of table will show you the number of headings of each type:

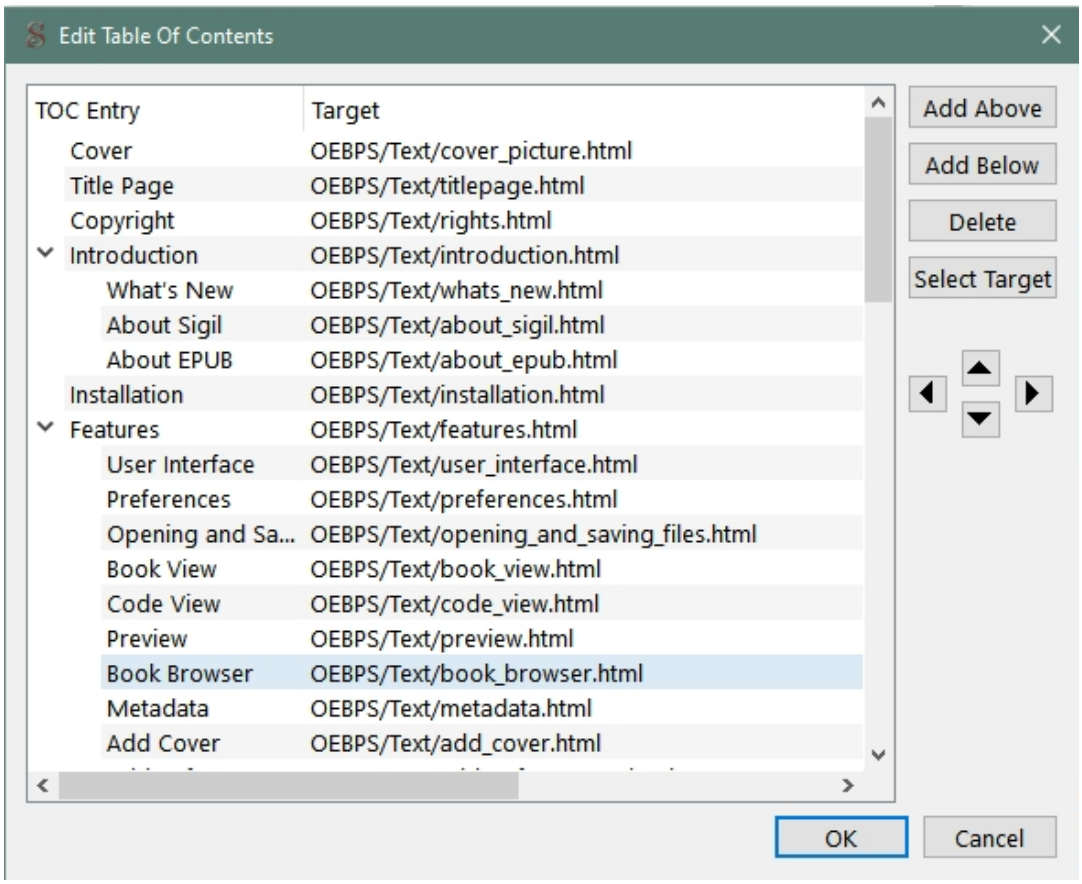


When you select **OK**, Sigil will add or update automatically generated IDs to any headings you have selected if they don't have regular IDs already. It will then create a new NCX file on eEPUB2 and Update the NAV file on EPUB3 that contain links to your chapters. If your heading is "near" the beginning of your HTML file, Sigil will link directly to the file and not to the heading ID. This is because on some e-readers linking directly to an element slows down the display of the TOC.

> Generating the TOC automatically will overwrite any existing TOC in the toc.ncx file of an EPUB2 or the nav.xhtml file of an EPUB3.

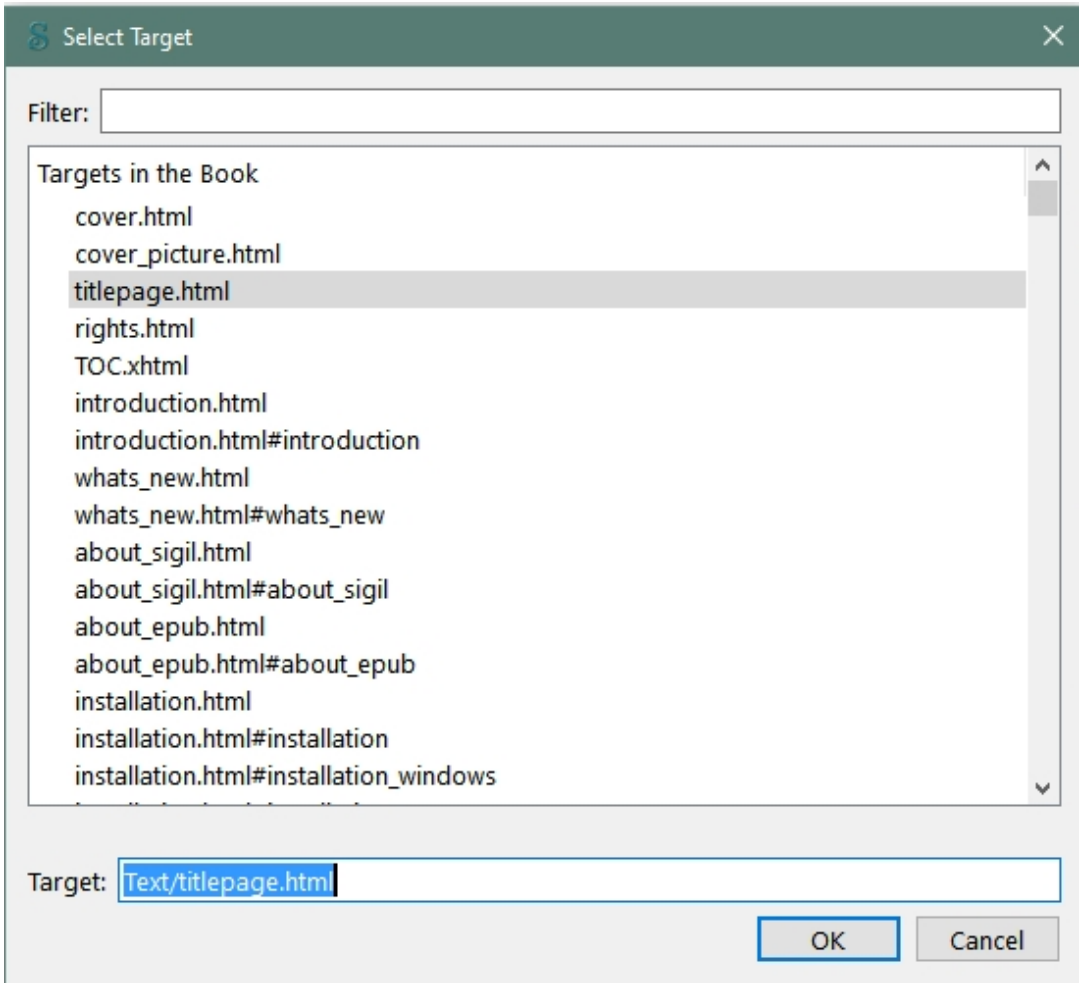
Edit Table of Contents

If you can't or don't want to generate the TOC from the headings in the document, you can use Edit Table Of Contents to make changes to the existing TOC. Any changes made to the TOC will not affect your HTML files, and all changes would be lost if you use Generate TOC.



You can edit an entry by double-clicking on it. You can add new entries or delete existing ones. Use the arrows to increase or decrease the heading level of an entry.

To change what a TOC entry points to you can type in the target destination directly, or use the **Insert Link** button. This will open the Select Target dialog where you can choose which location to point the TOC entry to:



The right-click context menu allows you to delete or rename entries, or expand or collapse the list.

> TOC entries must point to a valid target. If you just want a heading in the TOC, point it to the next valid entry.

Generate HTML Table of Contents

You can use the menu **Tools=>Table Of Contents=>Generate HTML Table of Contents** to create an Inline HTML TOC for your EPUB. This type of table of contents is meant to be seen and used by human readers and need not be machine readable unlike the NCX or NAV.

An inline TOC is just another HTML page in the EPUB that contains the Table of Contents in text format with links that can be used to jump to specific chapters.

Another reason for creating an HTML Table of Contents is that these can be converted to other ebook formats that do not support the NCX of EPUB2 or NAV of EPUB3.

The Generate HTML Table of Contents menu item will overwrite any existing HTML TOC file (if it has a semantic type of TOC). If no such file exists it will create an HTML file called toc.html and mark its semantic type as TOC. If toc.html already exists it will be replaced.

You can change the contents of the inline TOC once you've created it – but remember to do this each time you create the TOC.

Custom HTML TOC Stylesheet

The generated HTML TOC page uses a separate CSS file (sgc-toc.css) to format the TOC. If the file already exists in your book it will not be overwritten (so you can customise it without losing your changes). But if it does not exist, it will look for sgc-toc.css in your preferences directory and use that as the default file. If there is no file in your preferences directory, then Sigil will create a default stylesheet for the TOC.

If you're generating an EPUB3 TOC, Sigil will look for sgc-nav.css in your preferences directory and use that as the default file.

Update The NCX TOC In EPUB3 Books

If you create an EPUB3 book with Sigil or open an EPUB3 book without an NCX file, no NCX will exist. If you would like to add an NCX to make your EPUB3 more backwards compatible after editing is finished, use **Tools=>Epub3 Tools=>Generate NCX/Guide for epub2 e-readers** menu item to generate and add a NCX that older EPUB2-only devices should be able to use.

Validation

— Making sure your EPUB is Error Free —

In order to be considered a valid EPUB, your ebook must meet the standards defined for EPUBs. This means it must include certain files and information and not contain any errors as defined by the EPUB specification. To help you check if your EPUB is valid you can use the following tools which are available for free as plugins for Sigil.

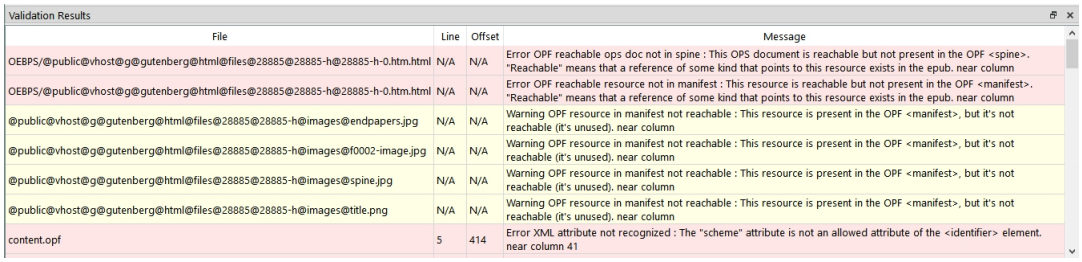
EPUB Validation With FlightCrew (EPUB2 only)

Please note that FlightCrew is no longer included in Sigil and has been converted to an plugin. (If you press F7, Sigil will only run a very basic and limited well-formedness check.)

Before you can use FlightCrew, you'll have to download the FlightCrew plugin .zip file for your OS and install it.

To run FlightCrew select Plugins > Validation > FlightCrew.

This will display a new window underneath your editing window listing any issues that it finds. Once you have run FlightCrew to produce a report, look at each of the issues listed and then edit your document to correct them.



> You can double-click on most entries to be taken to the relevant part of the book.

You should validate your document each time you change it to verify there are no more errors.

Common Validation Issues:

Note that errors are highlighted in pink, and warnings are highlighted in yellow.

- This resource is present in the OPF <manifest>, but it's not reachable (it's unused).
    - This warning means you have included the listed file in your EPUB, but you have not used it anywhere in your book. If you intended to use it, you should link to it, otherwise you can delete the file. Try to search all your files or toc.ncx for the filename – you should leave off the pathname when searching, e.g. if the error is in Text/Chapter1.html then just search for Chapter1.html
- This OPS document is reachable but not present in the OPF <manifest or spine>. "Reachable" means that a reference of some kind that points to this resource exists in the EPUB.
    - This error means you have a reference or a link to a file or image in your EPUB, but that file is not included in the EPUB. You need to add the file to the EPUB or correct the name of the file used in your text. You can usually find these by searching for the filename listed in all the HTML files. You may also have to re-generate or update your TOC.
- ID value 'SOMEIDNAME' is not unique
    - This error means you have an ID named SOMEIDNAME in your EPUB that is defined more than once. Typically these are in headers or footnotes and can be caused by copying and pasting or merging. You can find these by searching all HTML files for the ID name that is listed and changing them so all IDs are different from all other IDs. If the issue is in a heading and you have auto-generated your headings, you can delete the ID and they will be re-generated the next time you create your TOC.
- attribute 'INVALIDATTRIBUTE' is not declared for element 'SOMEELEMENT'
    - This error means you have an HTML tag named SOMEELEMENT that contains an attribute INVALIDATTRIBUTE, but INVALIDATTRIBUTE is not valid for that tag. For instance <h2 invalidattribute="test">. This can be caused by typos or by old versions of HTML. You need to search for INVALIDATTRIBUTE in all HTML files and remove, correct, or replace each occurrence.
- no declaration found for element 'INVALIDELEMENT'
    - This error means you have an HTML tag named INVALIDELEMENT that has not been defined. Typically this is simply because it is invalid and is old HTML syntax that need updating. You need to search for INVALIDELEMENT in all HTML files and remove, correct, or replace these with valid HTML.

- element 'INVALIDELEMENT' is not allowed for content model '(p|d|h2|h3|h4|h5|h6|div|ul|ol|...'
    - This error means you have an HTML tag named INVALIDELEMENT that is not valid HTML for an EPUB. Typically this is old HTML syntax that needs updating (such as the tag which should be implemented using a style). You need to search for INVALIDELEMENT in all HTML files and remove, correct, or replace each occurrence with valid HTML.
- value 'SOMENAME' is invalid NCName
    - This error means you have an ID name that is invalid. It probably contains a space or other invalid character. You need to search for SOMENAME and change it.

CSS Validation With W3C

You can also easily validate your CSS Stylesheet files using the W3C website by selecting the menu item **Tools=>Validate Stylesheets With W3C**.

This will send the contents of all of your stylesheets to the W3C service which will produce reports of any issues.

You can validate individual stylesheets by right-clicking on the stylesheet in the Book Browser and selecting the validate option.

> Validation is done using the W3C Stylesheet Validation Level that was selected in the Preferences menu (**Edit=>Preferences=>General Settings=>EPUB2/3 W3C Stylesheet Validation Level**).

CSSValidator Plugin

There also exists a CSS Validator plugin you can use to validate all of your CSS Stylesheet files. Use **Plugins=>Manage Plugins** to install it.

EPUB Validation With EpubCheck (EPUB2 and EPUB3)

FlightCrew identifies most issues with EPUBs, but it may not catch everything and can't be used to validate EPUB3 books.

For validating all EPUB3's and as an additional check for EPUB2s you should also run your EPUB through the EpubCheck program. You can use it online or download a local version of the program:

- IDPF EPUB Validator

But the best way to use EpubCheck with Sigil is to download the lastest version of the EpubCheck plugin and install it just like was done with FlightCrew earlier in this chapter.

Development of Epubcheck is done at https://github.com/w3c/epubcheck.

EpubCheck uses java on your machine and the latest version of the epubcheck.jar to verify your EPUB meets the EPUB specification.

When EpubCheck installed and used as a plugin, it will return errors and warnings in the Validation window just was described above for FlightCrew making it easier for the user to navigate to and fix any errors.

EPUB Validation With Devices

And in case you missed it: You should always test your EPUB on as many devices or programs as possible.

Spellcheck

— Checking and Correcting Spelling —

Sigil provides several ways to check your spelling – highlighting misspelled words, finding words one at a time, or showing all misspelled words at once. You can choose a standard dictionary to use, and create personal user dictionaries with your own words.

> Spellcheck uses Code View to show misspelled words.

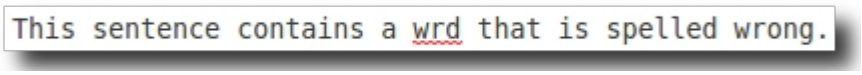
Configure Spellchecking

You can set the spell checking preferences that you want – including which main dictionary to use, and which personal user dictionaries to use if any. You can set up user dictionaries for different types of books or individual books, or use only the default user dictionary. You can change user dictionaries at any time.

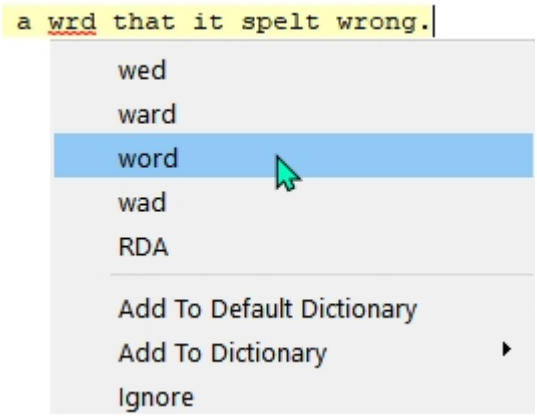
> Set up your dictionaries in Preferences.

Automatic Spellchecking

Code View will highlight any misspelled words with a wavy red underline when automatic spell checking is enabled in Preferences or using the menu item **Tools=>Spellcheck=>Highlight Misspelled Words**:



You can then right-click on the misspelled word to bring up a list of choices:



- **Suggested Words:** A list of alternative words to choose from – selecting one will replace the misspelled word.
- **Add To Default Dictionary:** Add this word to your current default user dictionary.
- **Add To Dictionary:** Add this word to the dictionary you select from the list.
- **Ignore:** Ignores the spelling of this word in the entire document until you restart Sigil.

Finding Misspelled Words

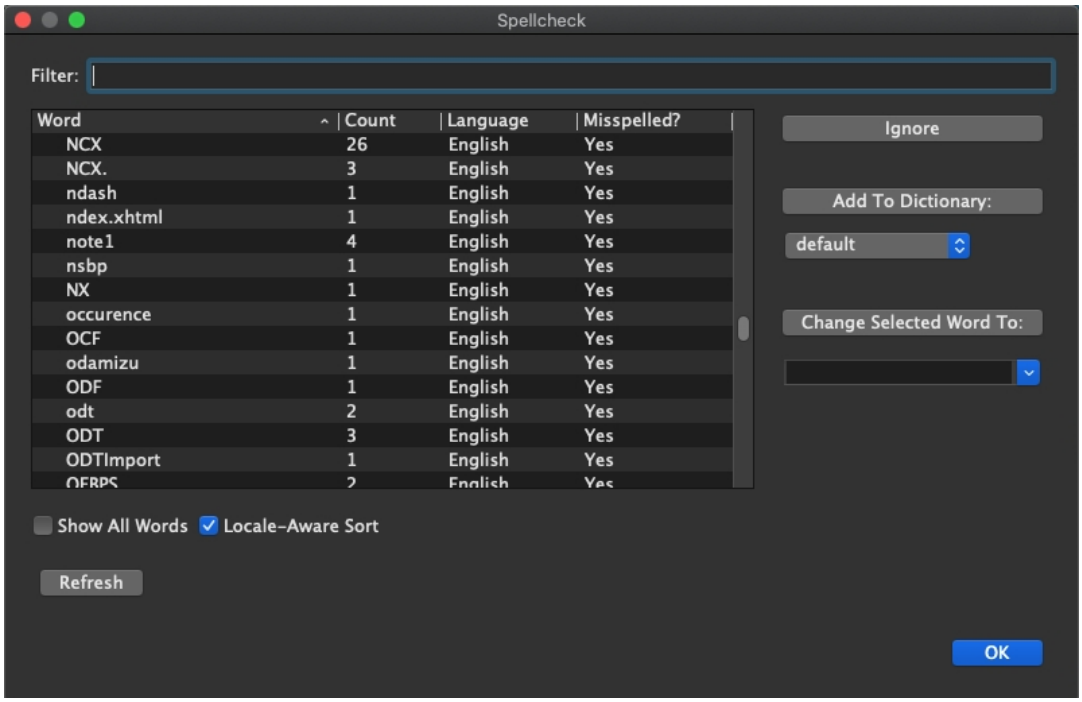
Find the next misspelled word in Code View by selecting the menu item **Tools=>Spellcheck=>Find Next Misspelled Word** or using its shortcut. You will be switched to Code View automatically if you are not in it already, and the next misspelled word will be highlighted each time you search for the next word.

In order to be able to right-click on the misspelled word to select a suggested word or add it to the dictionary you must have **Tools=>Spellcheck=>Highlight Misspelled Words** enabled.

Spellcheck

To open the Spellcheck dialog, click the Spellcheck toolbar icon

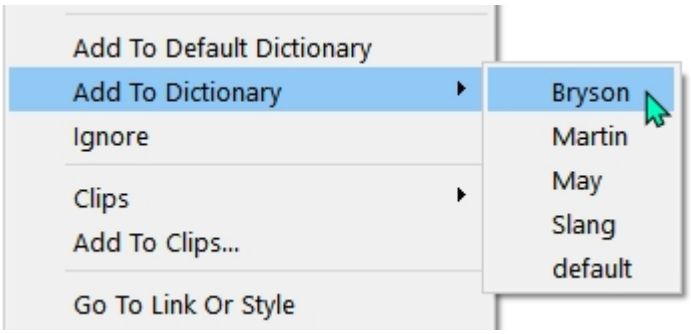
, or use the menu item **Tools=>Spellcheck=>Spellcheck** or the keyboard shortcut – Alt+Q (Windows), Option+Q (Mac).



Words are shown with the number of times they occur in the book and if they are misspelled. Select a word to perform an action.

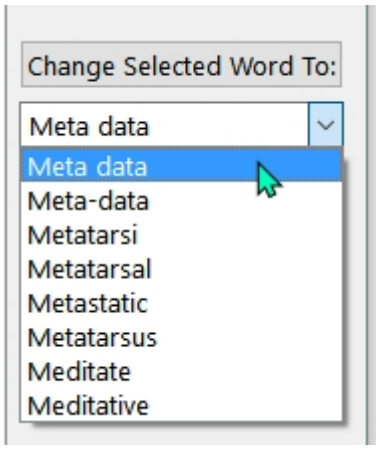
> Selecting a word will also show the first occurrence of the word in Code View so you can view its context and change just the one occurrence if needed.
>
> Double-click a selected word to find the next occurrence of the word.

You can add words to a dictionary by clicking **Add to Dictionary**. Select your user dictionary to add the word to from the dropdown list:

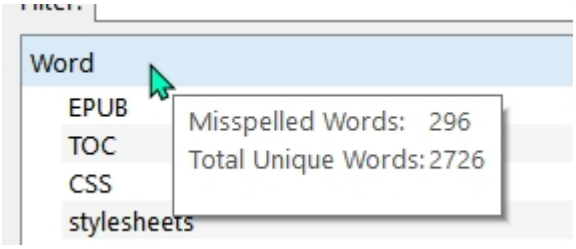


To Ignore words, click **Ignore**. The Misspelled Word column will update to "No" only if you have enabled the dictionary to be used in Preferences. To clear all ignored words use **Tools=>Spellcheck=>Clear Ignored Words**. Ignored words are also cleared when you open a new book.

To correct the spelling of words, select one of the suggested words from the dropdown below click **Change Selected Word To:** or type your own word in the box. Then click **Change Selected Word To:**. This will change every occurrence of the selected word to the new word. It will also automatically refresh the list of words from the HTML files.



You can see all words in your EPUB by selecting ☑ Show All Words. In addition, the tooltip for the column names shows the number of words in your EPUB:



If you enable ☑ Locale-Aware Sort then the words will be sorted as determined by your locale or region (as AaBbCc instead of ABCabc).

Use the **Refresh** button if you edit your book while Spellcheck is open. You do not need to use it if you are just ignoring, adding, or changing a word.

Images, Video, Audio

— Inserting Media Files into your book —

You can insert any image, video, or audio file into your book clicking the Insert File button or selecting the menu item **Insert=>File or its shortcut**. If the file has not already been added to your EPUB, you can select it from your computer.

> EPUB 2 and most e-ink readers only support images, but the ability to insert video and audio is available if needed. Both video and audio are natively supported in EPUB 3

When you insert a media file into your book, it is first added to the EPUB and then a link to it is created. This allows you to add a file once and then link to it many times, or to add lots of files at one time and then later create links to them.

Adding Media Files to Your EPUB

You can add image, video, or audio files to your book in several ways:

- Click the Add Existing Files button .
- Select the menu item **File=>Add=>Existing Files**.
- Right-click on a file in the Book Browser and select **Add Existing Files**.

All of these will display a standard file dialog allowing you to select one or more files from your computer. Select as many media files as you want. If any of the files already exists in the book, you will be prompted to confirm if you want to replace them.

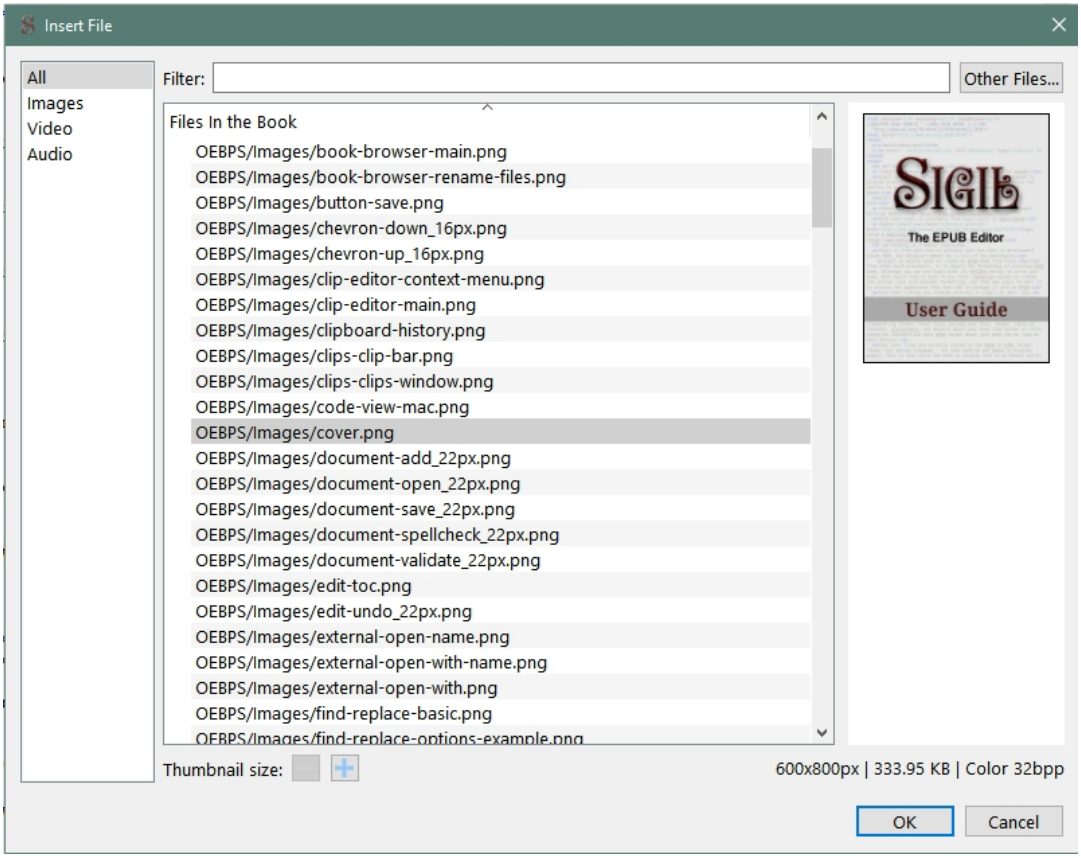
The files will be added to the appropriate folder in the Book Browser based on their file extension. However, they are not yet actually used anywhere in your book – they are just available to use. To use them, you need to link to or insert them in your book.

You can actually add files to your book and insert them in one step:

- **Click the Insert File button** and then click on **Other Files**.
- Select the menu item **Insert=>File** and then click on **Other Files**.

Inserting Files Into Your Book

To insert an image, video, or audio file into your book click on the location in your file where you want the image to be located in Code View. Then click the Insert File button :

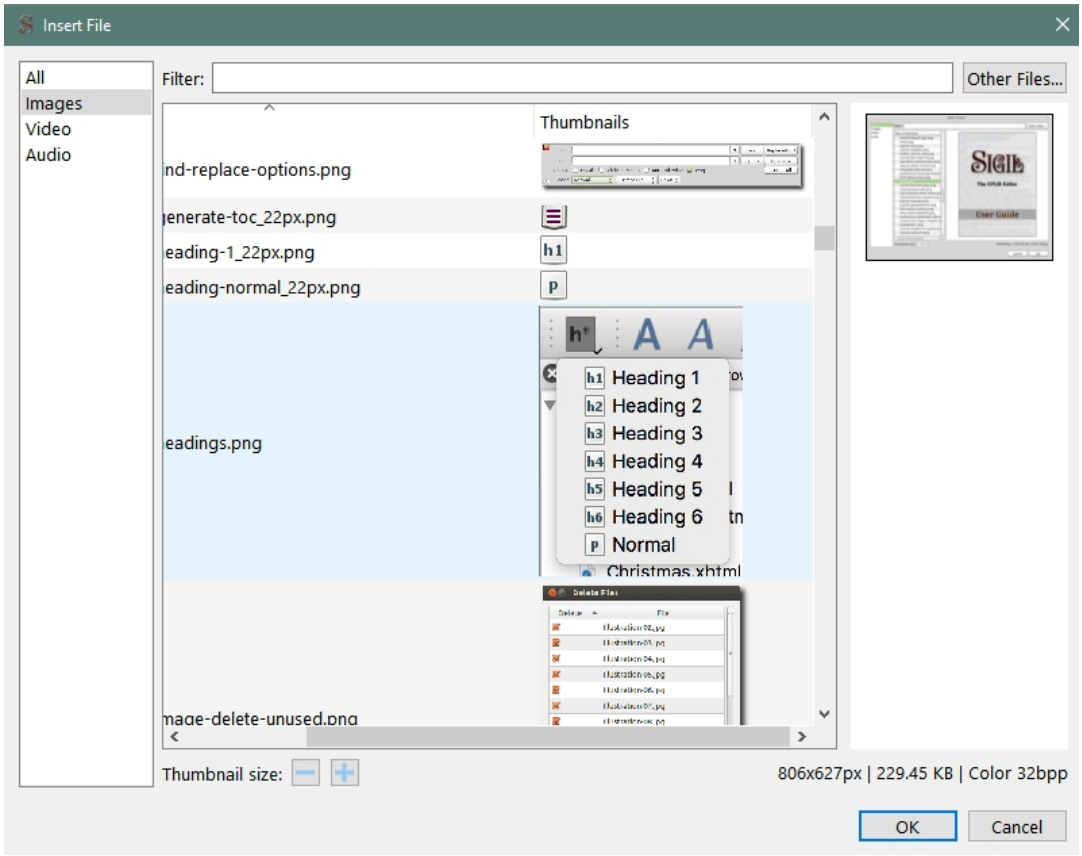


A list of the files in your book is displayed. Select one or more files to insert and click **OK**, or double-click on the file you want in order to select and insert it automatically.

> You can choose files from your computer instead of from the EPUB by clicking on the **Other Files** button. Files will be added to your book and then inserted automatically.

You can filter which files are shown by selecting the type of file on the left, or entering part of the filename in the Filter text box.

You can enable thumbnails for images in the Insert File dialog by clicking on the zoom buttons on the lower left to choose the size of the thumbnails. The dialog is slower with thumbnails so you can remove them by making them smaller until they are gone.



View File Details

To view details for an image, video or audio file double-click on its filename in the Book Browser.

For an image, the Width and Height in pixels, size of the file in kilobytes, and whether or not the image has any color defined are shown:

Editing Media Files

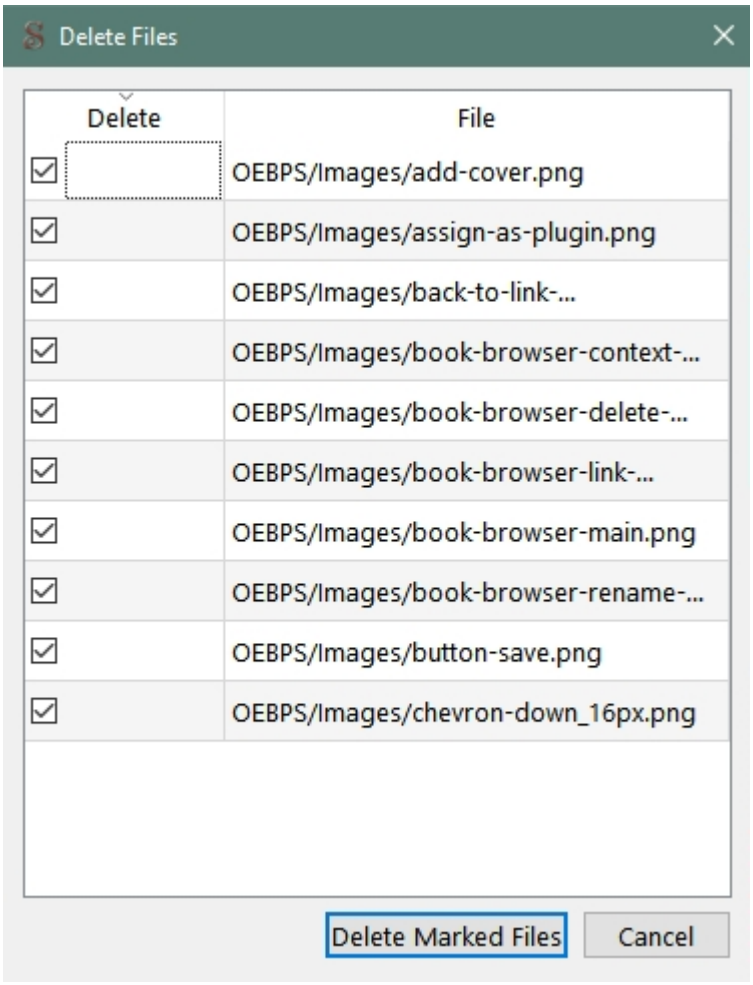
Sigil allows you to easily edit an image, video, or audio file once it is in your EPUB.

Simply right-click on its filename in the Book Browser, and select the menu item **Open With**. This will open the file in the external editor of your choice.

See the External Editors chapter for details.

Deleting Unused Media Files

You can select the menu item **Tools=>Delete Unused Media Files** to automatically delete any unused image, video, or audio files. Sigil will present you a list of files that do not have any links to them from your HTML or CSS files and then allow you to confirm which files to delete – or to cancel deleting them.



For more control over deleting individual files, you can use a report to view the number of times a file is used and then delete the ones you want – see the Reports chapter.

> You can double-click on a file to show that file in a tab.

Cover Images

You can quickly add a cover image using Add Cover.

But if you want to add a cover manually, then be aware that images for covers are special in that you must tell the EPUB which image should be used for the cover. Different e-readers use different methods to identify the cover, so you should use both methods:

- Create a cover.html file at the start of your book and then insert your image into it.

- Mark the cover.html file as a cover file by right-clicking on it in the Book Browser Text folder, selecting **Add Semantics** and choosing **Cover**.
- Mark your actual cover image file as a Cover by right-clicking on its file in the Book Browser Images folder, selecting **Add Semantics** and choosing **Cover**.

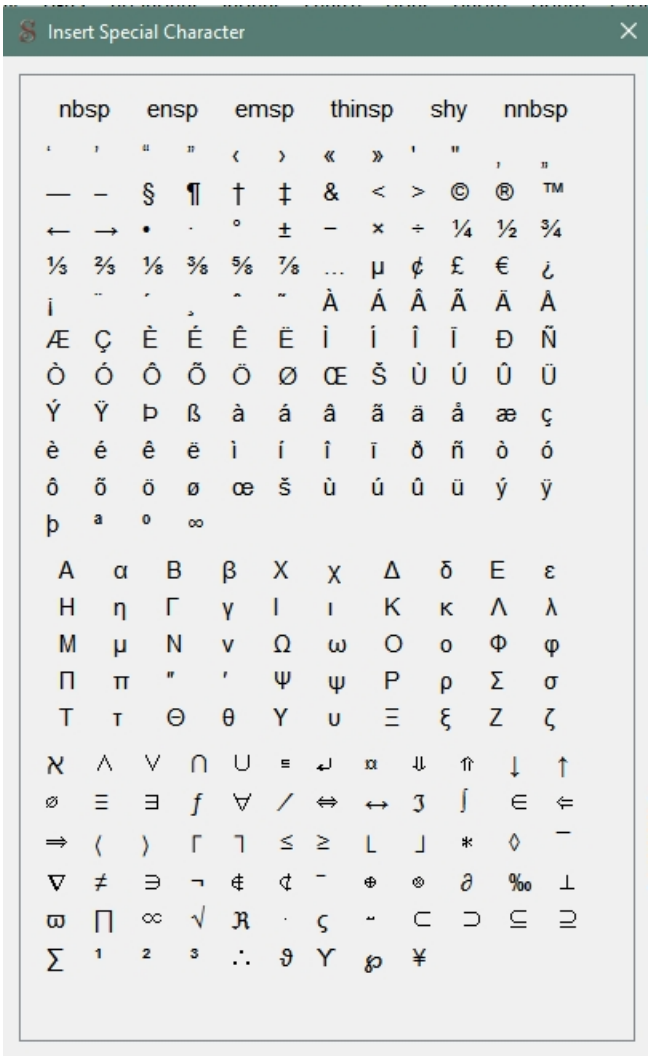
Please make sure you have permission to use any images in your book as most images are copyrighted and their use may require a license.

Special Characters

— Insert Special Characters —

You can easily insert characters that might not be on your keyboard into your document using the Special Characters dialog.

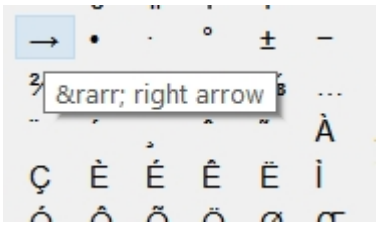
To insert special characters click the Insert Special Character button , or select the menu item **Insert=>Special Characters**.



The dialog will stay open to allow you to insert text multiple times, and you can move the window to any position on the screen you want.

You can use Ctrl-click to select and insert the character and close the dialog window immediately.

If you pause over a character you will see a tooltip listing the HTML entity code for the character and a description:



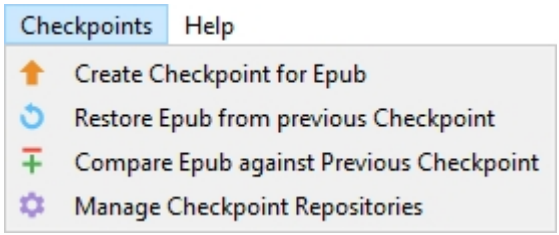
You can change the font and size of the characters in the dialog using Preferences.

Some characters may not display correctly in some fonts.

You can also create a Clip button and shortcut for special characters you use often.

Using Checkpoints to Quickly Save/Restore Epub State

Sigil has many powerful tools to automate repairing, splitting, merging, renaming, and moving xhtml, images, stylesheets, and other resources. These tools can make numerous changes that a simple Undo/Redo can not handle. To help prevent inadvertent data loss and to provide a way to "undo" even larger scale changes, we have added a Checkpoint capability to Sigil.

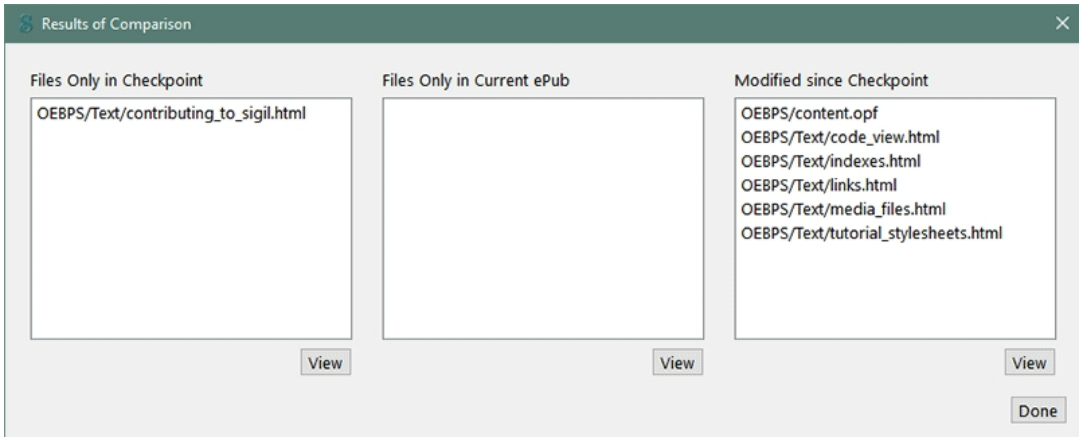


A checkpoint is a quick snapshot of the entire state of the EPUB that is stored in its own git style repository that is kept inside your Sigil Preferences folder. Creating a checkpoint can be as simple as one click on the Checkpoint icon or by using the Checkpoint Menus. Each Checkpoint made of an EPUB is named/numbered sequentially.

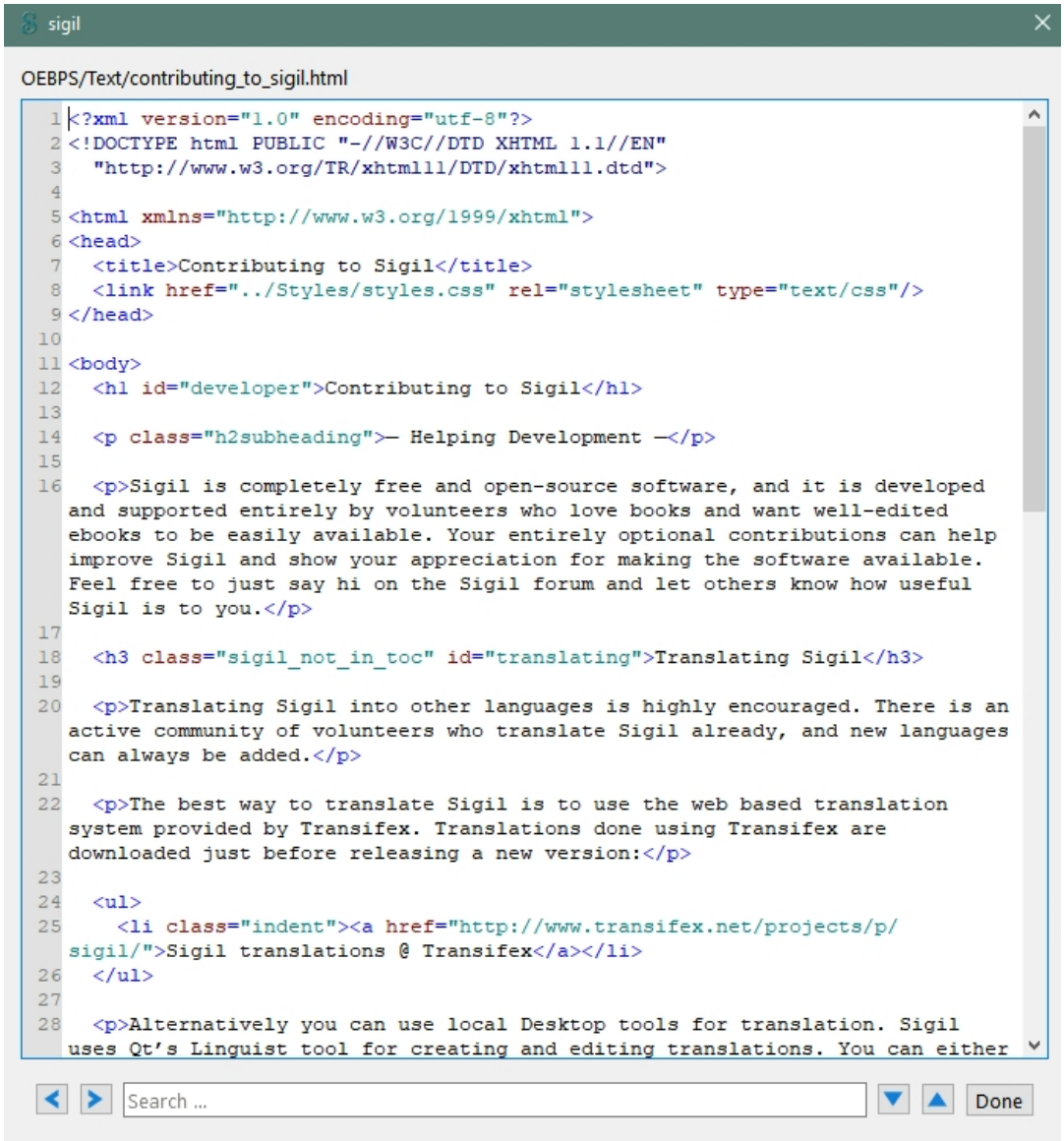
Once a Checkpoint is made, users can feel safe to make larger changes to their copy knowing that they can easily restore to their last checkpoint.

An added advantage of using checkpoints is that a user can always compare the current state of their EPUB against any of their earlier checkpoints to see exactly what was changed and how. Here is image showing the output of a comparison done against an earlier checkpoint during an editing session. After selecting the

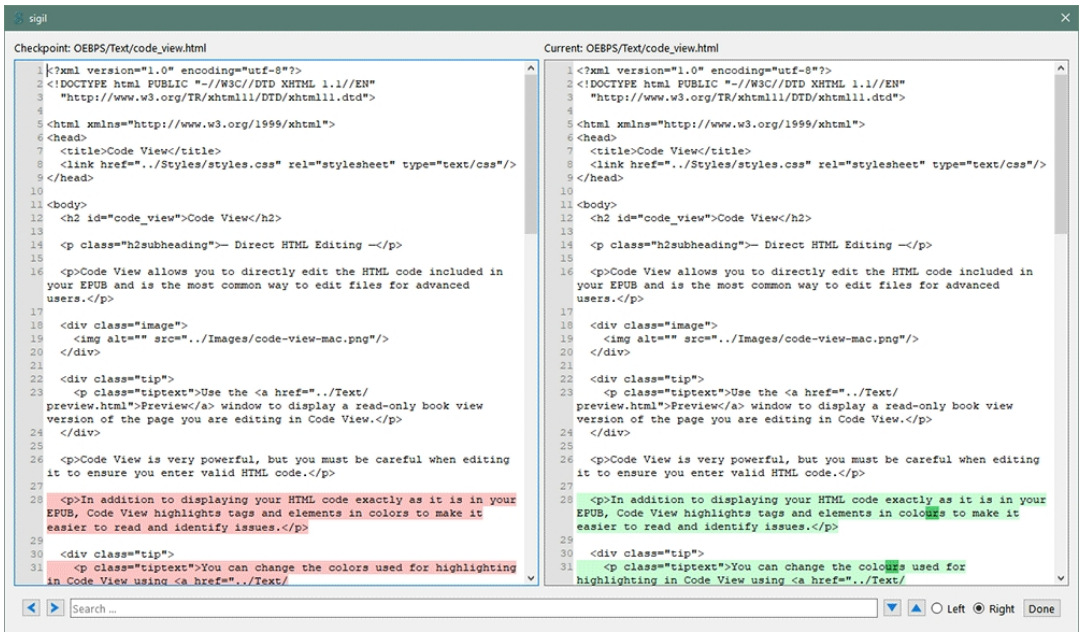
checkpoint to compare against the following dialog is shown showing which files have changed:



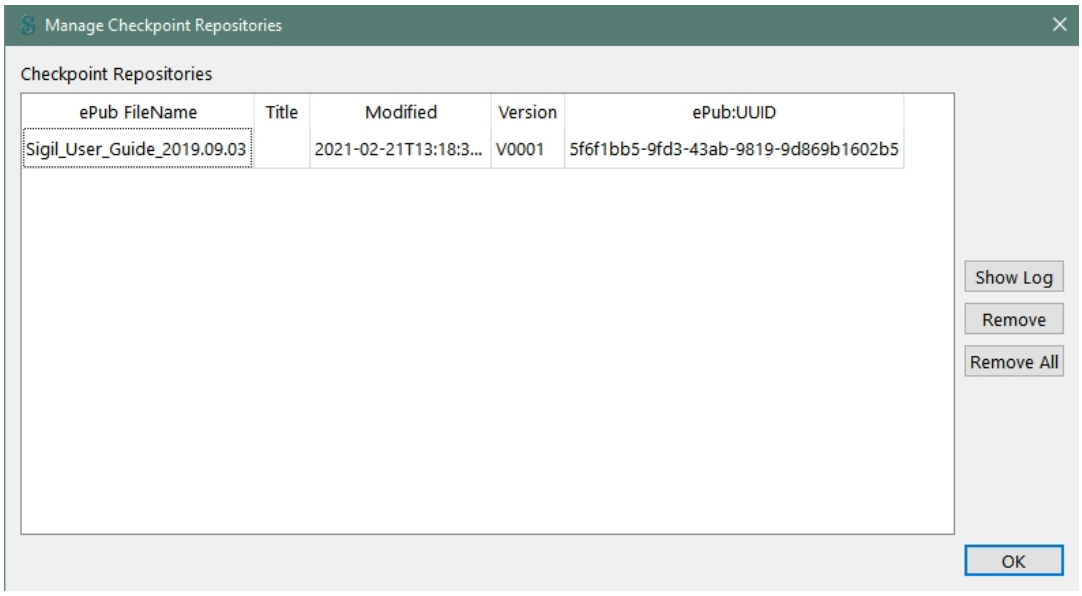
Users can select and view any of the files that only exist in the current EPUB or only exist in the checkpoint.



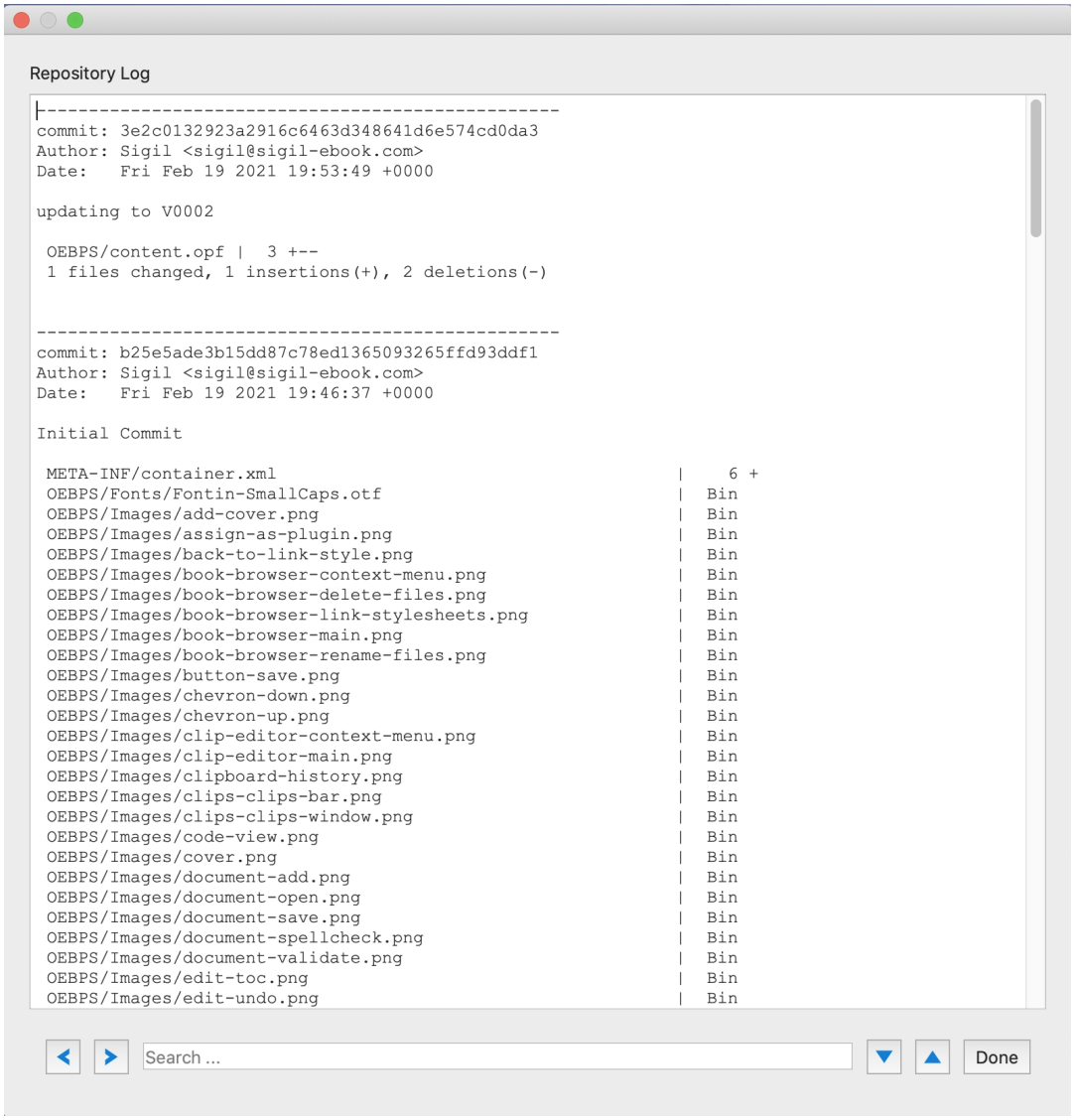
For text files that have been modified, the user is provided with a side by side highlighted view of both the lines changes and the areas within the lines that have been changed.



When the final version of the EPUB has been produced and saved, the user can manage their checkpoint repositories to remove them and free up disk space.



Or they can generate a summary of the changes made in each checkpoint to help them select earlier versions to restore from if needed.



Some Sigil tools like “Mend” or “Mend and Prettify” can make a large number of changes to white space which can help hide more significant changes later. Therefore it is a good idea that after using a tool that makes large white space changes, you checkpoint both right before and immediately after using the tool so you can make finding later changes much easier.

Since the both the EPUB file name and title can be easily changed, especially during editing, relying on the file name or title to uniquely identify an EPUB across changes not a good idea. To uniquely identify your EPUB, the Checkpoint tool will add a dc:identifier using the UUID scheme to your EPUB **if one does not already exist**. This identifier will form the basis of the repository name. It should remain unchanged during editing. This UUID identifier need not be your book’s unique-identifier, it just has to exist unchanged until you are finished using the Checkpoint tool.

Splitting and Merging

– Working with Chapters and Files –

An EPUB could consist of one large HTML file with all of your text but that EPUB would not load on some older ebook readers and will be slower to load and edit. Imagine having to layout an entire book in order to show the very first page. For this reason well designed EPUBs consist of multiple files each containing part of your book. This makes the EPUB accessible on even older e-reading devices while at the same time greatly speeding up loading, displaying, and even editing. All of your files can be easily accessed using the Book Browser.

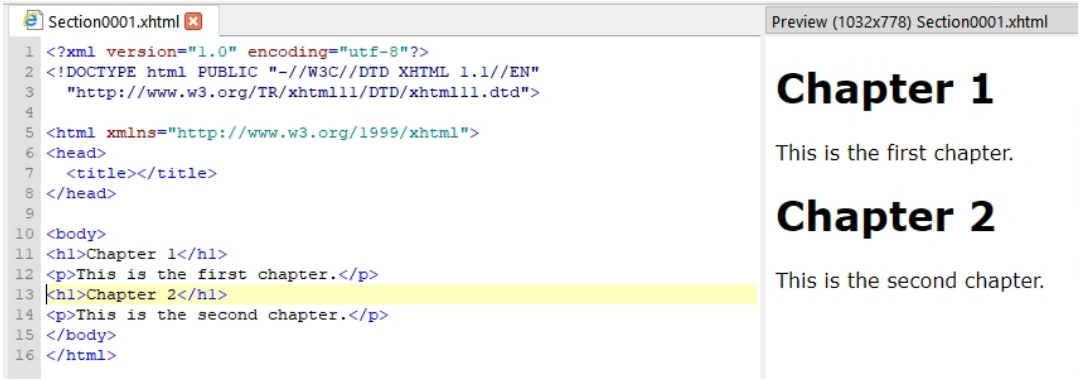
If you have loaded an large HTML file into Sigil and want to divide it into separate files or chapters, then you can use Sigil's Split functions. And if you have several smaller files that you want to combine into a larger file, then you can use the Merge function.

> Most e-readers start each new HTML file on a new page – giving you automatic page breaks between chapters if you split your Chapters into separate files.

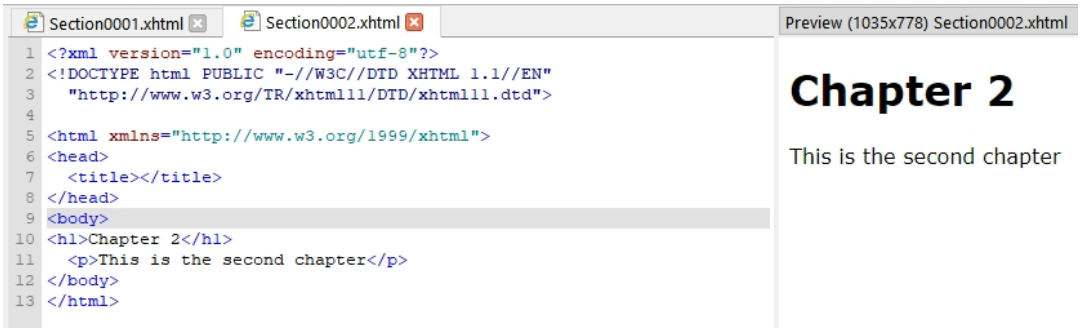
Split One File

You can split a file into 2 files at your cursor position immediately by clicking the Split At Cursor button or selecting the menu item **Edit=>Split At Cursor**.

This is the easiest way to split a single file. Simply click in your document exactly where you want the file to be split, e.g. at the beginning of a chapter name:



Then click to split the file:

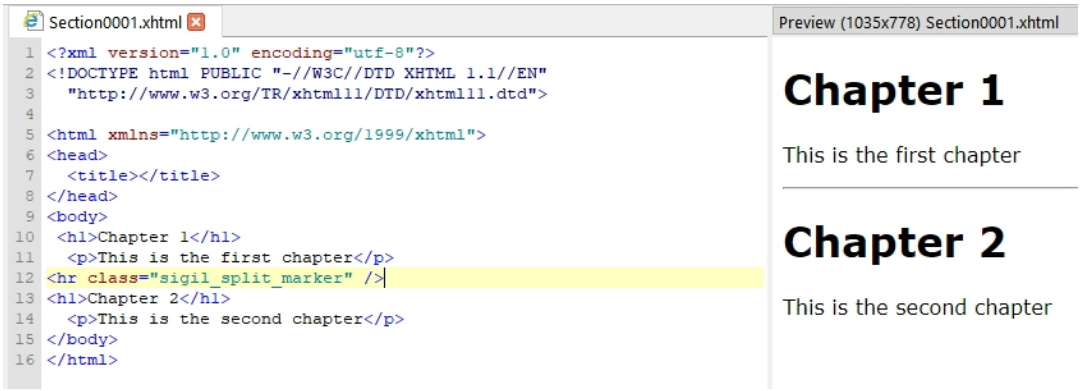


Everything from the beginning of the original file up to the point where you clicked is kept in the first file, and everything after the point you clicked on is put into a new file. Note that the first file kept its name, and a new file with a new name was created to hold the second file. You can rename and edit the files as necessary.

Split Multiple Files

If you want to split several files or a large file into many files, then you can use the file marking approach.

1. Scroll through your file, and wherever you want a new chapter to start click in the document to set your cursor position. Then select the menu item **Insert=>Split Marker** or its shortcut. This will insert code into your document telling Sigil to remember that you want a new chapter to begin here – you will see a horizontal line in Preview exactly where you inserted the marker:



2. When you have marked all your chapters, use **Edit=>Split At Markers**. This will automatically split all of your files into as many files as you marked. If you decide not to split the file, you can delete the horizontal marker lines.

Once you have split your files you can rename them. And you can even Merge some or all of them together again.

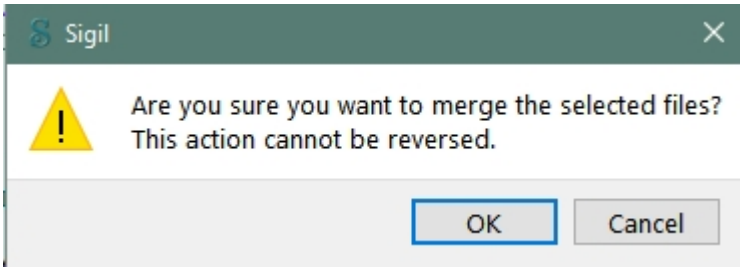
> You can use Find and Replace to insert split markers just before every chapter heading if you have a lot of chapters.
>
> If you split a file, any header information is copied to the new file.

Merging Files

If you want to combine two or more files into one file, you can use Merge.

To Merge two adjacent files, right-click on the second file and then select the menu item **Merge**, or select the second file and use the Merge shortcut. The second file will be merged into the first file (the file above). You will not be prompted to confirm the merge.

To Merge more than two files, select all the files you want to merge, then right-click on the selection and choose **Merge**. You will be prompted to confirm the merge:



Once you confirm OK, your files will be combined into the first file.

Save your EPUB before Splitting or Merging! Or alternatively make a Checkpoint to create a restore point in case you ever need to revert.

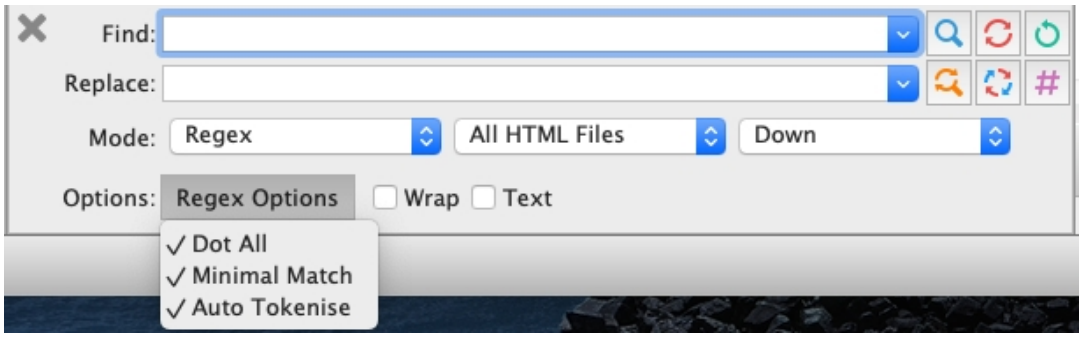
Find & Replace

— Searching in Code View —

Finding text and updating it is one of the most common tasks in Sigil – to look for words, to remove or replace unwanted text, or to change the formatting. In fact, if you ever need to make lots of changes to your text, you will almost certainly want to use Find & Replace (and its related Saved Searches feature), since it allows you to quickly make a large number of edits across all your files.

> See the Regex tutorial for information about advanced searching.

To open the Find & Replace dialog, select the menu item **Search=>Find & Replace**, its keyboard shortcut, or toolbar button .



When open, the Find & Replace dialog is always positioned at the bottom of the editing window for easy access. Advanced regular expression options have their own menu.

> Keyboard shortcuts exist for all Find & Replace operations, such as Ctrl+F to find and Ctrl+G to find again (on Mac use Cmd instead of Ctrl). See the **Search** menu for a list of shortcuts.

Action Buttons

The buttons on the right are used to run your search commands: Find , Restart, Replace, Replace/Find, Replace All, and Count All.

FIND ( )

Finds the next occurrence of your search term and highlights it in Code View.

To find text:

- Enter the text you wish to find in the Find box.
-

Click ☐ to search for and highlight the next match in the Code View window.

Any text in the Replace box is ignored if you are just doing a Find.

If you select text in Code View, then use Ctrl+F (on Mac: Cmd+F) to open the Find & Replace window, Sigil will automatically place your selected text in the Find box.

After you run a search, the text you entered in either the Find or Replace box is automatically saved to a history list that is available for later use. Click the down-arrow to the right of the Find or Replace text box to see a dropdown menu of your search history list.

Status messages appear in the lower right corner.

Restart ( )

Restarts a multiple file Find and and Replace Search back at the top of the first file. The next Find will begin from there.

Replace ( )

Replaces one occurrence of the text matched by Find.

To replace text:

- Enter the text to find in the Find box.
- Enter the text you want to replace it with in the Replace box.
-

  Click ☐ to search for and highlight the next match.
-

  Click ☐ to do the replacement.

To undo changes, use Ctrl+Z (on Mac: Cmd+Z) or the Undo button , or right-click in the Code View window and select **Undo** to change the text back. But if you are replacing across multiple files and your next match is in a different file, you need to switch back to the file containing the replaced text before selecting undo.

REPLACE/FIND ( )

Replaces one occurrence of the next match then automatically finds and highlights the subsequent match.

The first time you do a Replace/Find the first match will be highlighted but not replaced. Subsequent use of Replace/Find will replace the highlighted matched text and then automatically find the next match.

To replace text then automatically find the next match:

- Enter the text to find in the Find box.
- Enter the text you want to replace it with in the Replace box.
-

Click ☐ to search for and highlight the first match.

•

Click ☐ to replace the match, then automatically find and highlight the next match.

> You can safely use Replace/Find instead of Find if no text is selected since it will act just like Find until a match is highlighted.

Replace All ( )

Replaces all text matching your search term with your replace term, according to your settings. This is done all at once automatically.

> You cannot undo a Replace All – save first! Or create a Checkpoint

To replace all:

- Enter the text to find in the Find box.
- Enter the text you want to replace it with in the Replace box.
•

Click ☐ to replace all matches in one go.

Replace All always replaces from the start of a file.

To avoid unpleasant surprises, test your find and replace string using Count All to see if the number of matches is in the expected range. You can also tests with a simple Replace a few times.

Starting with Sigil-1.9.10 you can use the Dry-Run Replace All tool to safely see all of the potential replacements in one table before actually performing any replacements.

Save your book or create a Checkpoint before a Replace All and check it for issues afterward.

Before executing a Replace All, check if your mode is set for Current File or All HTML Files.

Count All ( )

Counts the number of times the search term you entered in the Find text box occurs in the selected files, then displays the results in the lower right corner of the Find & Replace window. Count All uses all Mode settings including Up/Down, plus the Wrap option.

To count all matches in the selected files:

- Enter the text to find in the Find box.
-

Click .

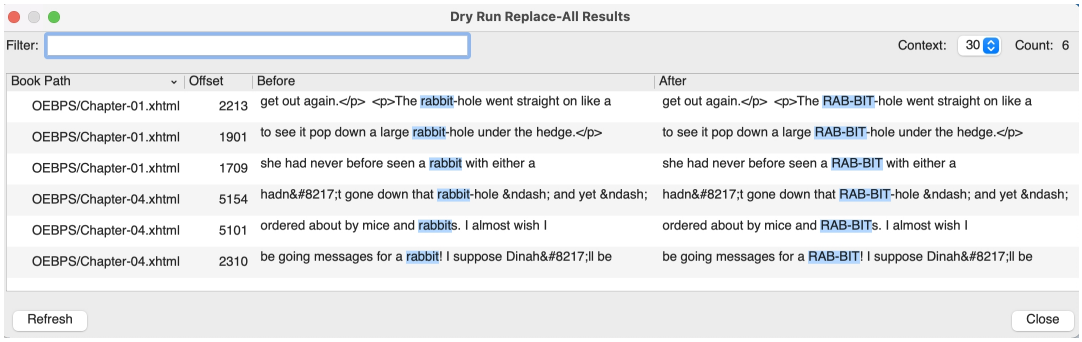
It’s a good idea to use Count All before doing a Replace All to confirm that the number of replacements will be what you expect. You should also run it after a

Replace All in case the particular search and replace created new matches. Alternatively use the new Dry Run Replace All tool (see next).

#

Dry Run Replace All (SHIFT +) (New in Sigil 1.9.10)

Often when crafting regular expression for Find and Replace, it is difficult to know if your find string will properly detect the cases you want while excluding the cases you do not want. This table shows the results of a trial dry-run of your current find and replace terms in context. No actual replacements/changes are ever made. The amount of context both "before" and "after" a match (in characters) can be controlled by the user. This tool can by launched by Shift clicking on the Count-All button.



Dry Run Replace All allows the user to safely experiment with their search terms to see which if any would be most effective. Any row in the table can be double-clicked and the exact context for that particular match will be brought up in CodeView for the user to examine in more detail or to edit. This way a user can verify that a find replace all sequence will do exactly what they expect it to do. If the user makes changes to the underlying text that will impact the table, the user can click on the "Refresh" button at any time to force the table to be rebuilt.

To create a Dry Run Replace All Table:

- Enter the text to find in the Find box.
- Enter the text you want to replace it with in the Replace box.

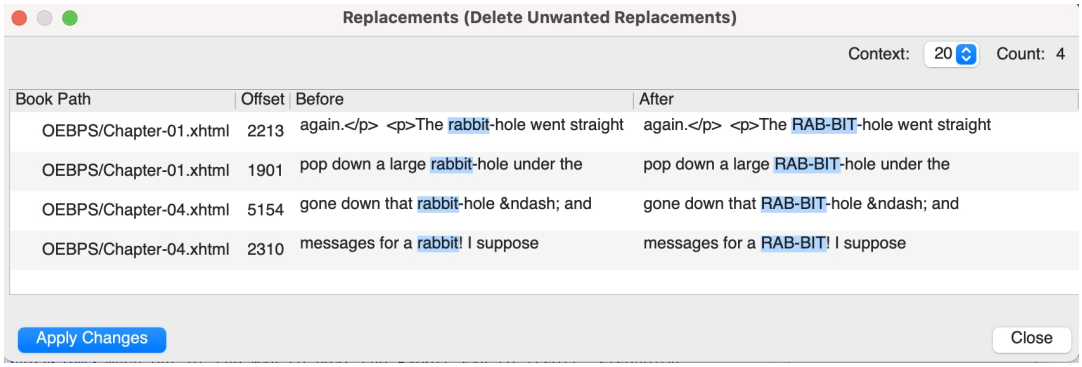
•

#

Click **SHIFT +** .

Filter Replacements (SHIFT +) (New in Sigil 1.9.10)

This tool helps users use Find and Replace All more effectively. It lists all of the matches found in context and shows each replacement in that same context. The amount of context both "before" and "after" a match (in characters) can be controlled by the user.



If the user sees a match that is unwanted - i.e. they do not want that replacement to be made, they can select that match's row in the table and use the delete key or select delete on the pop-up menu to remove it. When all unwanted matches have been found and deleted, the user can then use the "Apply Changes" button to have just the changes remaining in the table made.

Filter Replacements allows the user to more safely use Replace All by lowering the risk or undesirable replacements.

To create a Filter Replacements Table:

- Enter the text to find in the Find box.
- Enter the text you want to replace it with in the Replace box.

-

  Click SHIFT + .

Search Modes

Sigil provides three dropdown menus containing different search modes to give you greater control over your search.

What to Search For

- **Normal**: Search for exactly what you type in the Find box except ignore case, e.g. “A” and “a” will be considered the same letter.

- **Case Sensitive**: Search for exactly what you type in the Find box, e.g. “A” and “a” will be considered different letters.

- **Regex**: Short for “Regular Expression”, Regex mode allows you to search for patterns instead of exact words or characters. Regex is very powerful and one of the main benefits of using Sigil to work with ebooks. For instance, you can search for “Page [0-9]+” which will find “Page ” followed by one or more numbers.

> See the Regular Expressions chapter for more details about using regex.

Where to Search

- **Current File**: Search only in the file displayed in the Code View window. You can search all file types, including HTML, XHTML, CSS, OPF, NCX.

  If your search reaches the end of the file and Wrap is turned on, it will automatically wrap around and search from the other end of the file. A small

picture of an arrow will flash on the screen to indicate that the search wrapped around.

> To temporarily run the search on the Current File only, hold the Ctrl key down (on Mac: Cmd key) while clicking on an action button. This is useful if you want to search a stylesheet while the mode is set on All HTML Files.

- **All HTML Files**: Search in every HTML/XHTML file – but do not search in other file types, such as CSS, OPF or NCX.

  When a search reaches the end or beginning of a file, it will automatically move to the next or previous file (in the order listed in the Book Browser). If Wrap is turned on, it will loop around until all files have been searched.

- **Selected HTML Files**: Search only the HTML files selected in the Book Browser.

  When a search reaches the end or beginning of a file, it will automatically move to the next or previous selected file (in the order listed in the Book Browser). If Wrap is turned on, it will loop around until all selected files have been searched.

- **Tabbed HTML Files**: Search only the HTML files that are open as Tabs in the Tab Manager.

  When a search reaches the end or beginning of a file, it will automatically move to the next or previous selected file (in the order opened in the Tab Manager). If Wrap is turned on, it will loop around until all selected files have been searched.

- **All CSS Files**: Search in every CSS file – but do not search in other file types, such as HTML, OPF or NCX.

  When a search reaches the end or beginning of a file, it will automatically move to the next or previous file (in the order listed in the Book Browser). If Wrap is turned on, it will loop around until all files have been searched.

- **Selected CSS Files**: Search only the CSS files selected in the Book Browser.

When a search reaches the end or beginning of a file, it will automatically move to the next or previous selected file (in the order listed in the Book Browser). If Wrap is turned on, it will loop around until all selected files have been searched.

- **Tabbed CSS Files**: Search only the CSS files that are open as Tabs in the Tab Manager.

  When a search reaches the end or beginning of a file, it will automatically move to the next or previous selected file (in the order opened in the Tab Manager). If Wrap is turned on, it will loop around until all selected files have been searched.

- **OPF File**: Search only the OPF file.

  Special care must be used when using Replace in the OPF file as it is effectively the heart of the epub and controls pretty much everything. The importance of Checkpoints or saving the file first can not be overstated.

- **NCX File**: Search only the NCX file in EPUB2 and EPUB3 if one exists

- **Marked Text**: This is a special option that is only shown when you mark text. To restrict your search to a portion of text, first select some text in Code View, then go to menu item **Search=>Mark Selected Text** or use right-click context menu **Mark Selected Text**.

  Text can be unmarked by using the menu item again without a selection. Text will also be unmarked if you use Undo, type any text, or change tabs. If you just use Replace or Replace All, the marked text will remain marked, but it will be revised according to your replacement.

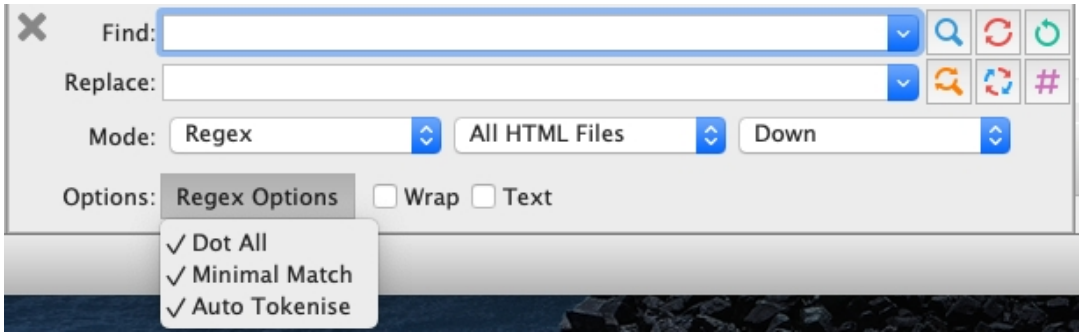
Search Direction

- **Up**: Search from your position in the document upwards and find the previous occurrence of your search.

- **Down**: Search from your position in the document downwards and find the next occurrence of your search.

Search Options

The Options menu offers useful switches when using Regex mode, as well as a checkbox to control whether Wrap in the Current File is turned on or off.



Regular Expression Search Options include:

- **DotAll**: This regex option prepends (?s) to all regex searches and is used when you want .* to match any character, even across lines.

- **Minimal Match**: This regex option prepends (?U) to all regex searches and is used when you want a pattern to match the shortest occurrence instead of the longest match.

- **Auto-Tokenise**: When using Ctrl-F (on Mac: Cmd-F) on selected text to copy it into the Find box, this will convert spaces to \s and escape certain characters so they are suitable for regex.

Other General Search Options

- **Wrap**: When enabled in Current File mode, if your search reaches the end of the file, it will automatically wrap around to the beginning of the file to continue the search (or the reverse if searching Up). When disabled, searches will stop at the end of the file (or the beginning if searching Up).

  When searching in a mode that uses multiple files, when your search reaches the last file (in the order listed in the Book Browser), you will need to use the **Restart** button otherwise the search will stop. **Wrap** does not apply when searching multiple files.

- **Text**: When enabled in any mode, if searching XHTML files, it will match only text outside of < XHTML tags >. This option has no effect when not searching in XHTML files.

  This is can be useful when doing find and replace on words that can incidentally be valid XHTML attributes, attribute values, or tag names that you no want to replace.

Context Menu

If you right-click on the Find or Replace box, it will bring up a context menu with the following options:

- ***Saved Searches*:** At the top of the context menu will be a list of your Saved Searches (see Saved Searches chapter for more information). Selecting one will load it into the Find & Replace window.

- **Save Search:** Save the current Find & Replace search to Saved Searches.

- **Tokenise Selection:** Convert spaces to \s and escape certain characters.

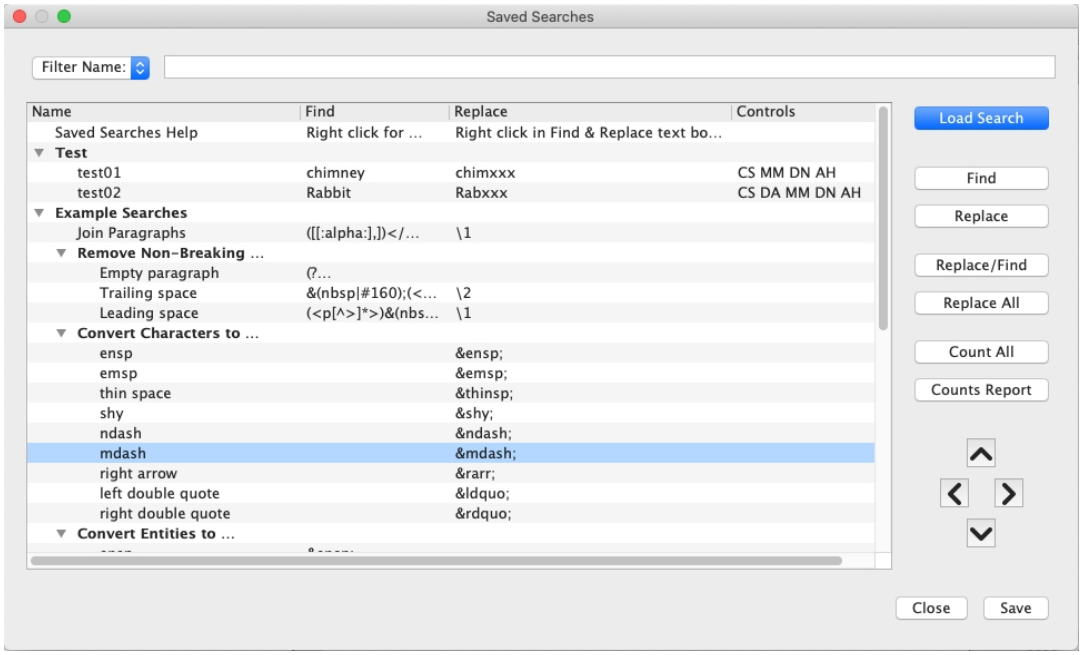
- **Clear Find Replace History:** Clears out the all recent historyor previous Find and Replace values.

Saved Searches

— Saved Searches for Find & Replace —

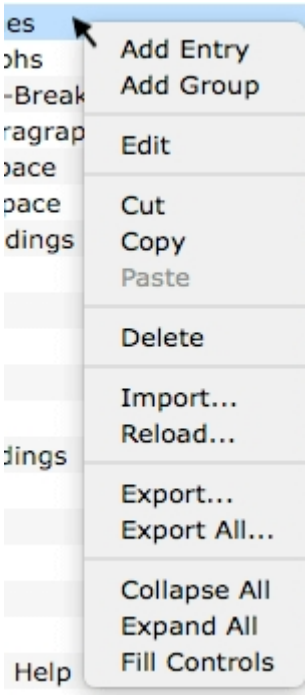
If you have Find & Replace searches that you routinely run – or complicated searches that you occasionally run but don't want to have to recreate every time – you can save them in the Saved Searches dialog. Once saved, you can use the dialog to edit and run the searches.

To open the Saved Searches dialog, select menu item **Tools=>Saved Searches**, its keyboard shortcut, or toolbar button .



Context Menu Commands

Most actions concerning creating and editing saved searches are accessed using the right-click context menu:



Select one or more saved search entries then right-click on the selection for the following options:

- **Add Entry:** Add a new blank search under the entry you clicked.
- **Add Group:** Add a new group under the entry you clicked.

- **Edit:** Edit or rename the entry (same as double-click).

- **Cut:** Delete one or more entries that can be pasted later.
- **Copy:** Copy one or more entries that can be pasted later.
- **Paste:** Paste any cut/copied entries.

- **Delete:** Delete selected entries or groups.

- **Import:** Read a list of exported entries from a file into the list.
- **Reload:** Restore the list from the last saved state.

- **Export:** Save selected entries to a file for sharing or backup.
- **Export All:** Save all entries to a file.

- **Collapse All:** Collapse the display of all groups.
- **Expand All:** Expand the display of all groups.

- **Fill Controls:** Copy the first Controls value to all other entries in the same group that are highlighted via shift-click. I.e., if you want the same Controls for multiple entries in a Saved Search group, enter the Controls for the first

entry, use shift-click to highlight the remaining entries that should use the same Controls, then select Fill Controls. (Requires Sigil 1.6.0 and higher.)

Selecting “Import…” will allow you to read in a new search group from a csv file, or from a previously saved Sigil ini file, or from a simple text file.

To move search entries or groups, use the arrow keys or drag-and-drop.

To make a group of searches, first create the group, then move the search entries into the group.

Running Searches

The Saved Searches dialog can be used to run searches as well as manage them. All of the Find & Replace action buttons are available in the Saved Searches dialog, so you do not need to open the Find & Replace window to run a saved search.

Using Controls - Newly Added in Sigil 1.6.0

The “Controls” column allows you to set search mode, search direction, any search options and the search targets (the files to search).

Controls are represented by two-character codes, which can be in any order, separated by commas or spaces:

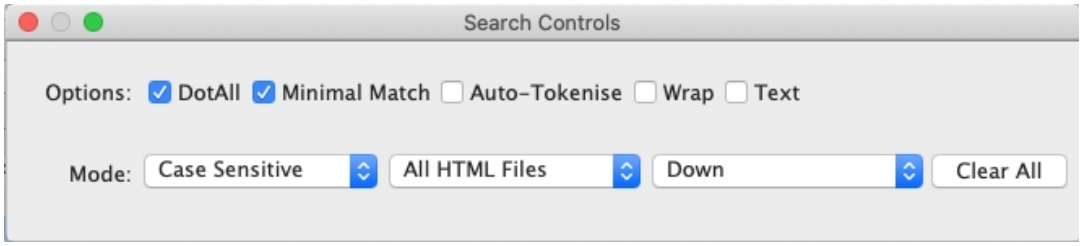
- **NL** - Mode: Normal
- **RX** - Mode: Regular Expression
- **CS** - Mode: Case Sensitive

- **UP** - Direction: Up
- **DN** - Direction: Down

- **CF** - Target: Current File
- **AH** - Target: All HTML Files
- **SH** - Target: Selected HTML Files
- **TH** - Target: Tabbed HTML Files
- **AC** - Target: All CSS Files
- **SC** - Target: Selected CSS Files
- **TC** - Target: Tabbed CSS Files
- **OP** - Target: OPF File
- **NX** - Target: NCX File

- **DA** - Regex Option: DotAll
- **MM** - Regex Option: Minimal Match
- **AT** - Regex Option: Auto Tokenise
- **WR** - Option: Wrap
- **TO** - Option: Text

To edit the Controls, double-click in the Controls column of your search to see a Find & Replace-like interface for setting the Mode, Options, Direction, and Targets. Tooltips are provided to make quick overview of existing Controls settings simpler.



If the Controls field is left blank, then the search setting in the current Find & Replace window will be used. If in doubt, open the Find & Replace window to check what modes and options you last set, as that is what will be used when you run any saved search.

To run a search, select an entry and click one of the action buttons: Find, Replace, Replace/Find, Replace All, Count All. Status messages will appear in the lower left corner of the Saved Searches dialog; they will also appear in the lower right corner of the Find & Replace window if you have it open.

> All Saved Searches are run using the mode, options, direction and targets set in the Controls column.
>
> To run a search only on the Current File, regardless of the mode setting, hold the Ctrl key down (on Mac: Cmd key) while clicking on an action button.

Running Multiple Searches at One Time

You can run multiple searches with a single button click if you:

1. Select a group name instead of an individual entry,
   or
2. Select multiple non-group entries by holding down the Ctrl key (on Mac: Cmd key) while clicking on multiple individual entries.

Once selected, click **Replace All**, and each search in the group or selection will run sequentially, one after another. When running multiple searches in a group, the Controls value if blank will inherit the values of the previous search in the search grouping. Setting the Controls for the first search of any search grouping will suffice.

Save

Once you are done creating and editing your searches, be sure to save them by clicking **Save**.

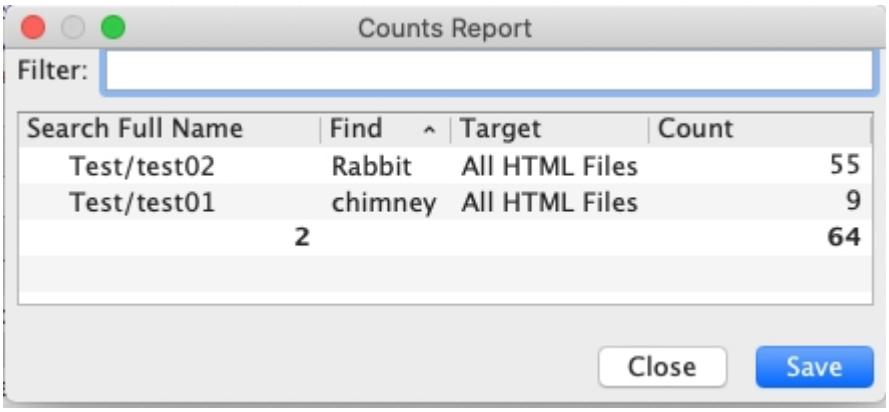
To exit without saving your entries, close the window with the **Close** button or Esc key, then confirm that you don't want to save, if prompted. You can also use context menu **Reload** to reload your entries from disk, which will delete current unsaved entries.

Use the **Export** context menu item to save a copy of your searches so that they can be shared with others (who can then use **Import** to add them to their Sigil).

> If you save and are running more than one copy of Sigil, then the other instances of Sigil will automatically reload the newly saved searches to keep all copies in sync.

Counts Report (New in Sigil 1.9.10)

This feature was introduced to take any user selected Saved Search Group and generate a report of the counts of all matches for each specific saved search in the group. This allows users to create custom report tables on the number of uses for any feature or tag (ie. lists, tables, images) that can be defined by a regular expression search.



A pop-up context menu allows the user to export the Counts Report to a CSV file for subsequent processing in a spreadsheet.

Filter

Use the filter text box to restrict the list of entries shown, making it easier to find the search you want.

Load Search

If you prefer to run your searches from the Find & Replace window, select the search from the Saved Searches dialog, then click **Load Search**. It will then be loaded into the Find & Replace window. Alternatively, you can right-click directly in the Find & Replace text boxes and select a saved search from the context menu (see more below).

Loading does not run the search – it is only a way to populate the fields in the Find & Replace window.

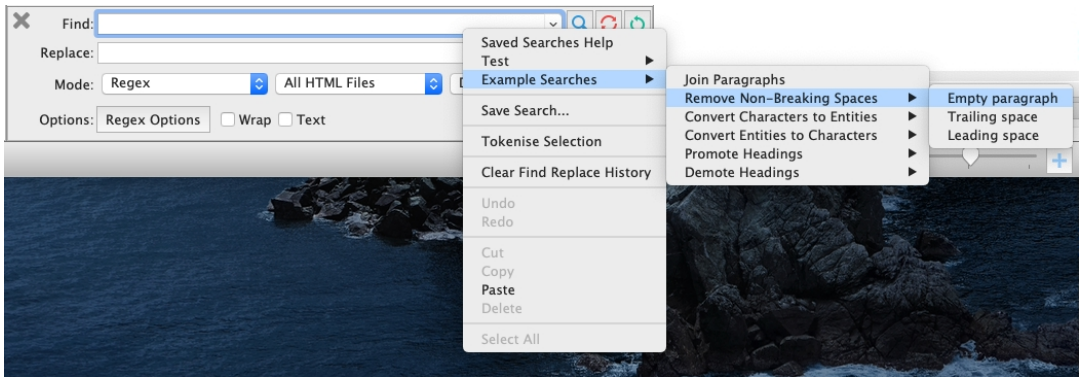
You do not need to use the **Load Search** button if you are running a search from the Saved Searches dialog.

> You can load a search using just the keyboard by using the shortcut to open Saved Searches, typing your search name in the filter box and hitting return (followed by Esc if you want to close the window).

Using Saved Searches from the Find & Replace Box

You can access your saved searches directly from the Find & Replace window (without ever opening the Saved Searches dialog) by right-clicking in the Find or Replace text field and bringing up the context menu.

Your saved searches will appear at the top of the context-menu. Select one to load it into the Find & Replace window.



If you've run a search in the Find & Replace window that you want to save, call up the context menu by right-clicking in the Find or Replace text field, then select **Save Search**.

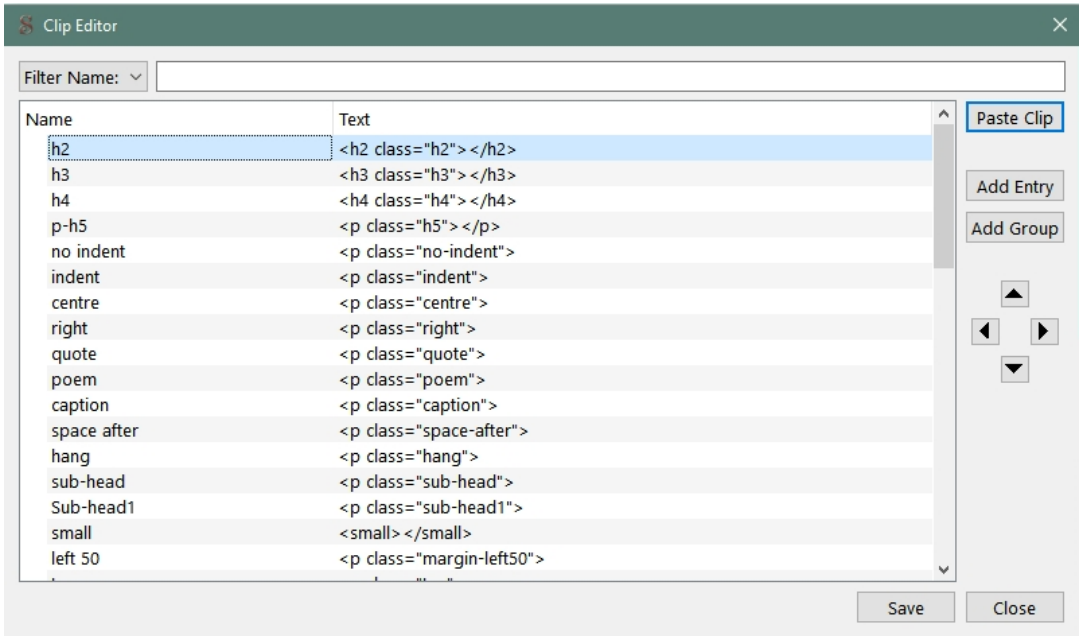
Clips

— Saving Text and Styles —

Clips allow you to save snippets of text and paste them into your document at any time. They can also be used to insert HTML style tags around your text. Create clips using the Clip Editor. You can even autofill all the styles in your CSS stylesheets into the Clips list. Then select and paste your clips using the Clip Editor, Clip Bar, Clips window, context menu, keyboard shortcuts, or main menu bar.

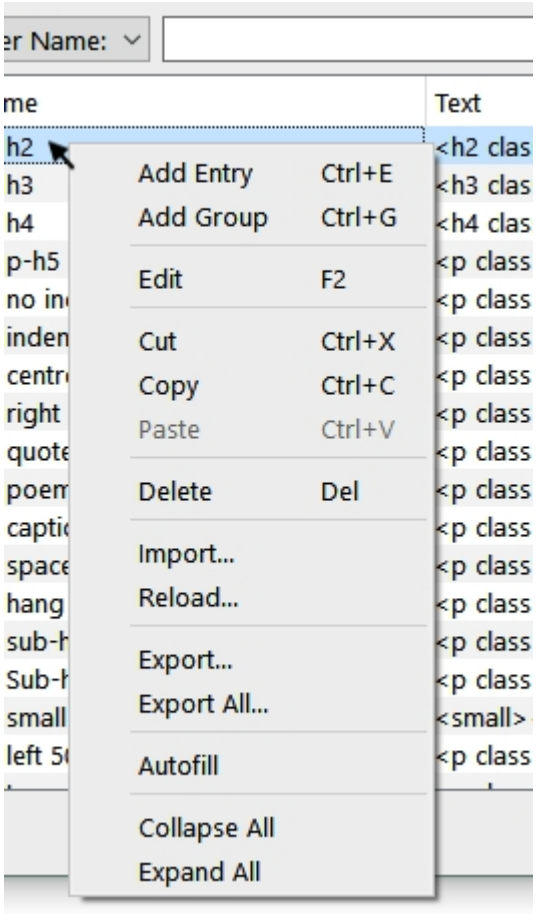
Creating & Editing Clips

To create and edit clips, first open the Clip Editor using the menu item **Tools=>Clip Editor** or its keyboard shortcut.



Context Menu

While you can add clips and groups using the **Add Entry** or **Add Group** buttons in the Clip Editor window, the full array of commands are accessed via the right-click context menu:



To use the context menu, first select one or more entries then right-click on the selection for the following options:

- **Add Entry:** Add a new blank clip under the entry you clicked.
- **Add Group:** Add a new group under the entry you clicked.

- **Edit:** Edit the entry.

- **Cut:** Delete one or more entries that can be pasted later.
- **Copy:** Copy one or more entries that can be pasted later.
- **Paste:** Paste any cut/copied entries.

- **Delete:** Delete entries or groups.

- **Import:** Read a list of exported entries from a file into the list.
- **Reload:** Restore the list from the last saved state.

- **Export:** Save selected entries to a file for sharing or backup.
- **Export All:** Save all entries to a file.

- **Autofill:** Create clips from styles in your CSS stylesheets.

- **Collapse All:** Collapse the display of all groups.

• **Expand All:** Expand the display of all groups.

> You can move entries or groups around using the arrow keys or drag-and-drop.
>
> To make groups of clips, first create a group to put the clips in.

Save

Once you are done creating and editing your clips, be sure to save them by clicking **Save**.

To exit without saving your entries, close the window with the **Close** button or Esc key, then confirm that you don't want to save, if prompted. You can also use the context menu **Reload** to reload your entries from disk, which will delete current unsaved entries.

Use the **Export** and **Import** context menu items to save a copy of your clips so that they can be shared with others.

> If you save and are running more than one copy of Sigil, then the other instances of Sigil will automatically reload the newly saved clips to keep all copies in sync.

Filter

Use the filter text box to restrict the list of entries shown, making it easier to find the clips you want.

Autofilling CSS Entries

The Autofill context menu item will scan your CSS stylesheets and create a clip entry for each simple selector it finds. (For more on how formatting text with clips works, see next section.)

For example, if you have the selector or style "p.first" then Autofill will create a clip for it as follows:

```
<p class="first">\1</p>
```

If the selector is "*.quote" then Autofill will create entries for "p.quote", "div.quote", and "span.quote".

All new Autofill clips are stored in the “Autofill” group. If the “Autofill” group does not exist, it will be created. You can rename the group once it has been created, and you can move or copy any entry to any group.

Using Clips to Format Text

Clips provide a great way to paste text before and after selected text, making it easier to format text.

First define a clip using “\1” in the text. Clips will then treat it as a Regex replacement command. When you paste the clip, the “\1” will be replaced with whatever text was selected.

For example, if your clip is

```
<span class="name">\1</span>
```

and you select the text

```
Mad Hatter
```

then when you apply the clip, the text will be changed to

```
<span class="name">Mad Hatter</span>
```

You can use this to build a library of styles and formatting that can be applied quickly.

You can use other regex commands in your clip, although complex regex statements might not work.

> Before pasting a formatting clip, be sure to remove any existing formatting by using the Remove Formatting keyboard shortcut (Ctrl+Space on Windows; Cmd +Space on Mac) or clicking the Remove Tag Pair toolbar button . (For Remove Formatting, include the tag in your selection; for Remove Tag Pair, place your cursor inside the tag you want to remove.)

Pasting Clips

Once you have created a list of clips, you can quickly and easily paste them into the active file in Code View using the Clip Editor, Clip Bar, Clips window, context

menu, keyboard shortcuts, or main menu bar. You can also paste clips into the Find or Replace fields in the Find & Replace box using the Clip Editor or Clip Bar.

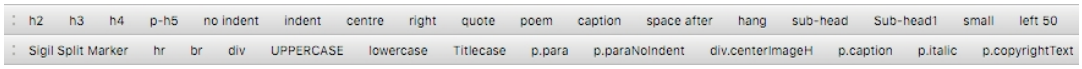
Clip Editor

To paste your clips using the Clip Editor, select the clip you want (if the list is long, use the Filter box to restrict the list of clips shown), then click the **Paste Clip** button. If you select multiple clips or a group, the **Paste Clip** button will paste all of the clips selected or in the group with a single click.

You can also paste your clip into the Find or Replace field in the Find & Replace box if you place your cursor in the appropriate field immediately before opening the Clip Editor window.

Clip Bar

Clip Bar and Clip Bar2 are optional toolbars that will show buttons for the first 40 clips (20 clips per bar) that are not in a group. You can display the clip bars by selecting the menu item **View=>Toolbars=>Clip Bar** and **View=>Toolbars=>Clip Bar2**:

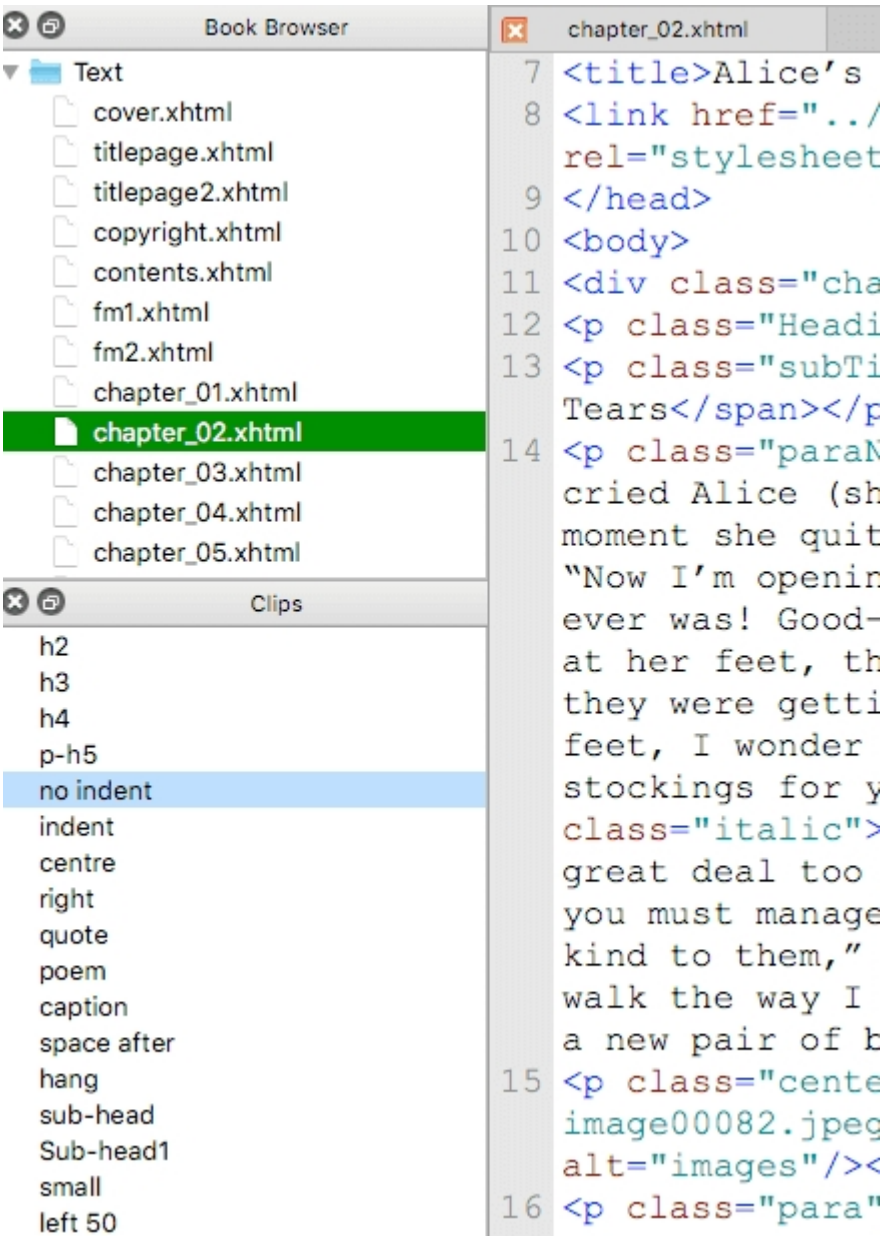


The buttons will show the clip's name, so it helps to keep names short. You can use them as shortcuts for your own styles or even for common special characters.

You can use the clip bars to paste clips into both the active file in Code View or the Find or Replace field in the Find & Replace box.

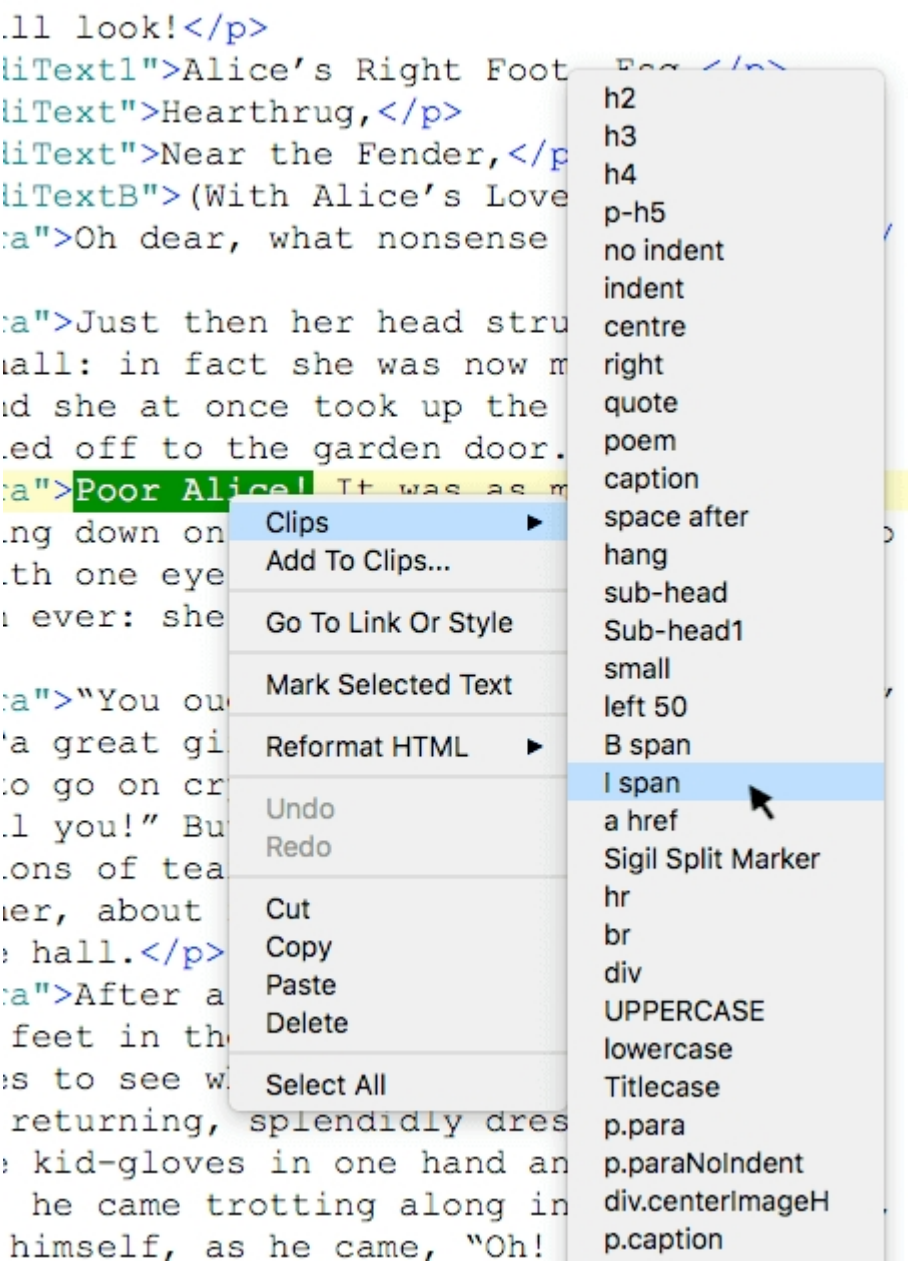
Clips Window

The Clips window shows all of your Clips in a scrollable list. Simply click an entry to paste it into your book. You can dock the window in the user interface (shown here docked below the Book Browser), or drag it to the desktop as a floating palette. To display the Clips window go to menu item **View=>Clips**.



Context Menu

Position your cursor or select the appropriate text in the Code View window where the clip should be inserted, then right-click to bring up the context menu and select a clip from the dropdown list.



Keyboard Shortcuts

You can assign keyboard shortcuts for up to 40 clips by going to **Edit=>Preferences=>Keyboard Shortcuts.**

You can also paste a clip from the Clip Editor using just the keyboard by using the shortcut to open the Clip Editor, typing the clip name in the filter box, hitting return, then Esc (if you want to close the window).

Main Menu

Clips can also be accessed from the Main Menu under **Insert=>Clip**.

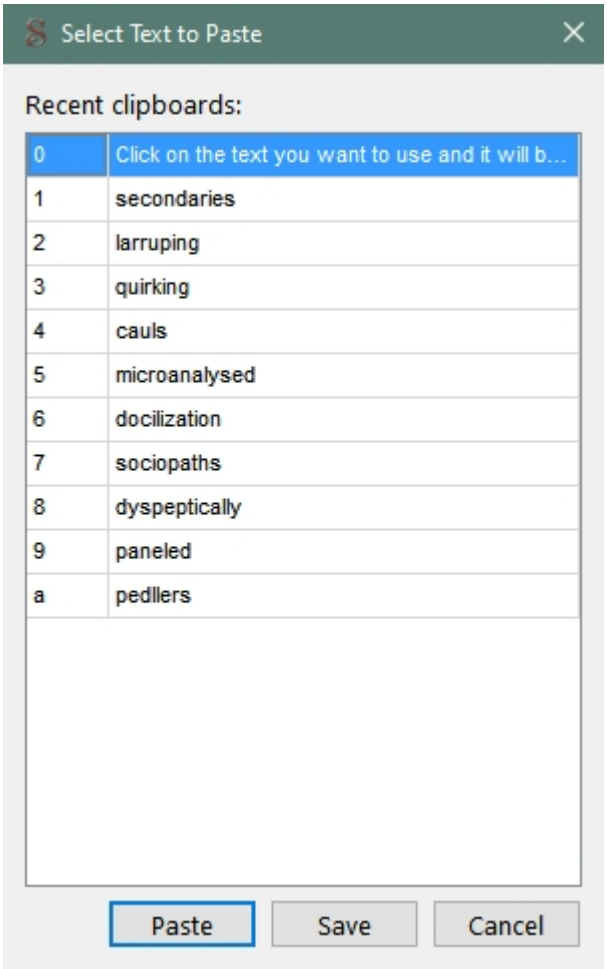
Clipboard History

— Clipboard List —

Clipboard History allows quick access to your most recently copied or cut text.

To add text to your clipboard, just use Copy or Cut.

To paste text from the clipboard, select **Edit=>Paste From Clipboard History** or its keyboard shortcut:



Click on the text you want to use and it will be pasted into your last selected window and the dialog will close automatically.

You can also just type the character shown for the entry to select it.

To delete entries from this history, simply highlight the entry you wish to delete and hit the delete key. This can prove useful to remove a particulary large clipboard copy to save space or to remove clipboard items you no longer need or want.

Links and IDs

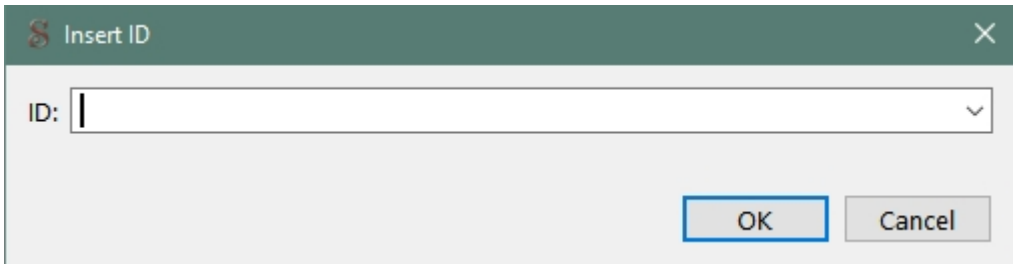
— Links, Notes, and IDs —

Sigil provides some helpful tools for creating ids and links which can make creating footnotes, endnotes, and cross-references easier.

Insert IDs

To mark text as a destination for a link you need to give it a name by inserting an anchor ID.

Select the text you want to mark with an ID and click the button or select the menu item **Insert=>ID**:



Type the name you want for this ID, e.g. footnote-1 (it must start with a letter).

To edit the ID, select the word or click somewhere in the word then click the button again.

The ID is actually inserted using HTML code like this:

```
<a id="footnote-1">Footnote 1</a>
```

> You cannot use identical IDs in the same file.

If you click on the drop-down arrow you will see a list of existing IDs for the file for reference.

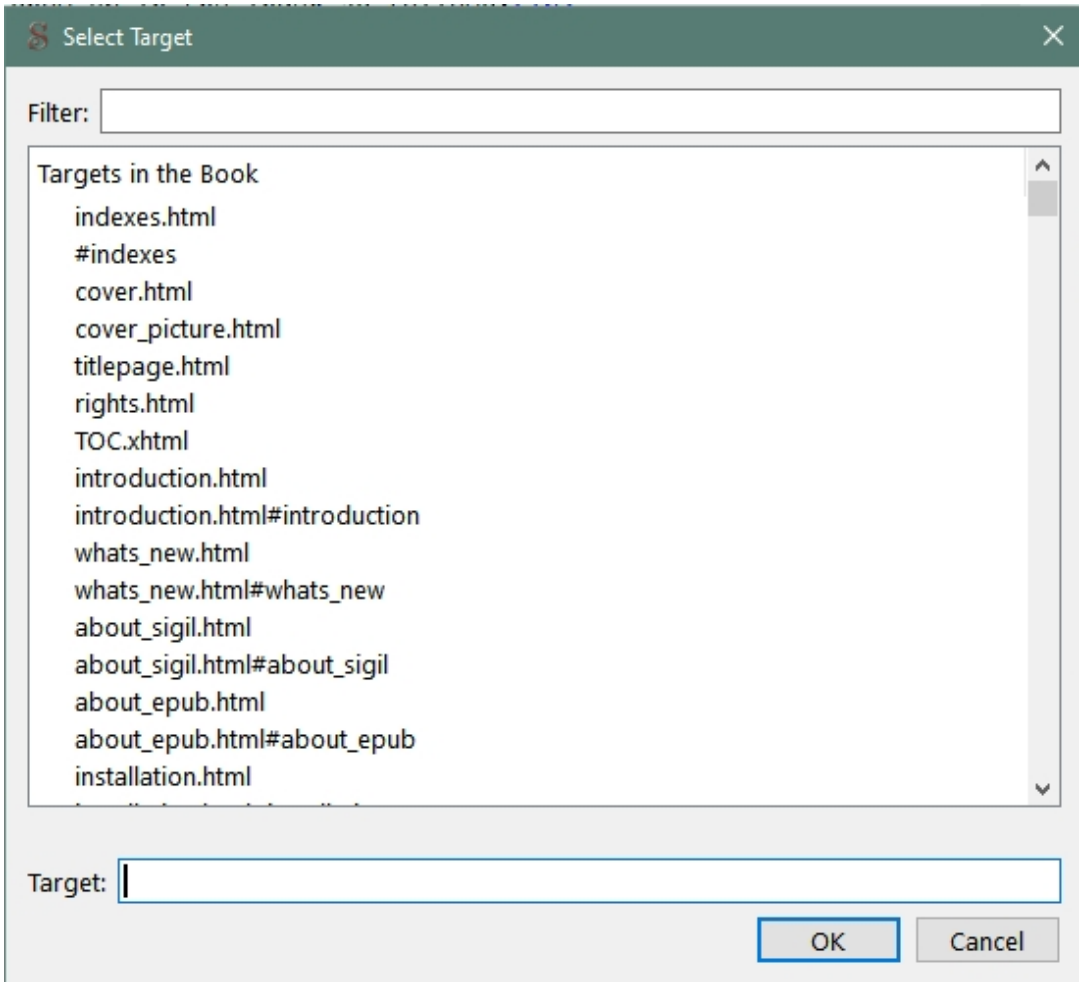
To delete an ID you will need to edit the HTML code in Code View to remove the tags completely.

To identify IDs in your book you can search in Code View. Or you can highlight them in Preview in color by adding the following CSS temporarily to your stylesheet:

```
[id class!=sigil_index_marker] { color:#00ff00; }
```

Insert Links

To create a link to another part of the document select the text you want to link and then click the button or select the menu item **Insert=>Link**:



You will be given a list of valid targets in the EPUB that you can select (double-click one or select and click **OK**). A valid target in the EPUB is any filename, any anchor IDs that have been created, or even any images. The list will be sorted so that targets in the current file are listed first, followed by all remaining files in book order.

To edit a link, select the text again or move your cursor in the text using the arrow keys (you can't click in it or you will be taken to the link) and then click the button . The link will be highlighted in the list if it is valid.

You can also enter any URL you want as a target for the link (e.g. to point to web sites) by typing the entry in the Target box.

Navigating Links

To go to a link in Preview just click on it.

To go to a link in Code View, hold the Ctrl key down while clicking on the HTML opening tag.

> Ctrl-clicking on text will take you to the link if the text is a link, or it will take you to the CSS style defined for the tag if the text is just a tag.

To go back to where you clicked the link, just click the Back button or select the menu item **Go Back to Link or Style**.

Styles

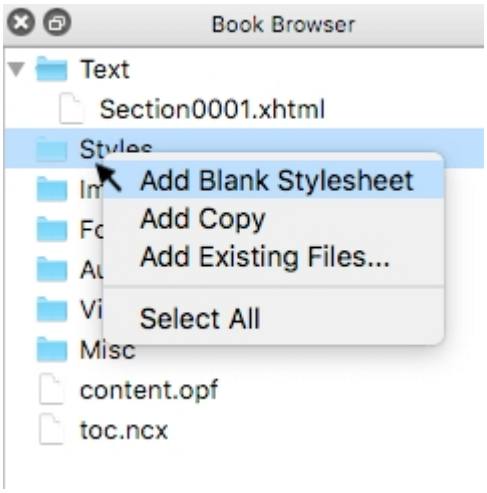
— Working with Stylesheets —

EPUB editing is all about styles. Sigil provides very easy-to-use yet powerful tools for helping you work with stylesheets.

While styles can be inserted directly into individual HTML files, the usual approach is to define an external stylesheet containing your styles, then link that file to your HTML files.

Creating Stylesheets

To create a stylesheet, go to the main menu bar and select **File=>Add=>Blank Stylesheet** or right-click on the Styles folder in the Book Browser and select **Add Blank Stylesheet**. Once created, type in the style definitions you want to use.

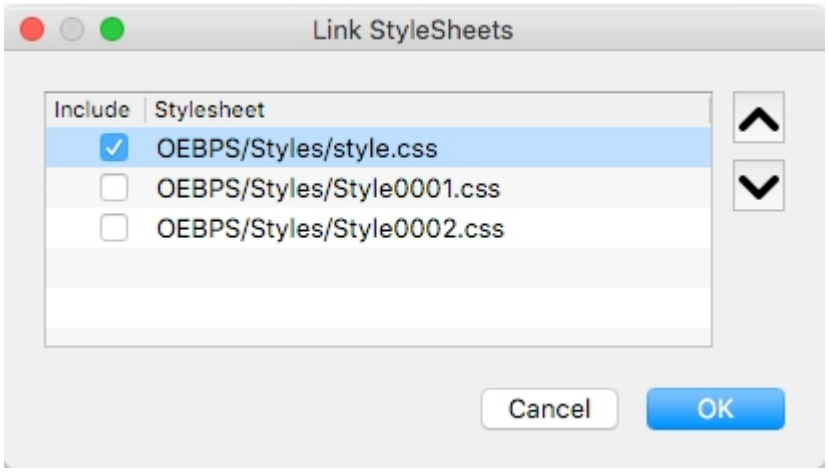


You can also add an existing stylesheet file, such as a template you've created, using **File=>Add=>Existing Files** or right-click on the Styles folder and select **Add Existing Files**.

> Do not include spaces in your stylesheet filename.

Linking Stylesheets

Link your stylesheets to your HTML files by right-clicking on the files in the Book Browser and selecting **Link Stylesheets**. Then choose which stylesheet(s) to link or unlink.



Navigating Styles

When working in Code View, there are times when you want to view the style definition of a tag. To quickly and easily jump from your HTML file to the specific style definition in question:

- Hold the Ctrl key down while clicking on the HTML tag. (On Mac hold the Cmd key.)
- Place your cursor inside a tag and select menu item **Search=>Go To Link Or Style** or use its keyboard shortcut.
- Right-click on the tag and select **Go To Link Or Style** from the context menu.

You will immediately be taken to the exact location of the style used for the HTML tag you are in (if one exists).

To return to where you were in the HTML file, click the Back button , or select the menu item **Search=>Back** or use its keyboard shortcut.

Format Your Stylesheet

To neaten up and format a stylesheet, open the stylesheet in the Code View window, then right-click inside the window and choose:

- **Reformat CSS=>Multiple Lines Per Style**
  OR

- **Reformat CSS=>Single Line Per Style**

Most stylesheets are formatted with individual styles arranged over **multiple lines**, making them easy to read:

style.css

```
24
25 .para {
26   display: block;
27   margin-top: 0%;
28   margin-bottom: 0%;
29   text-align: justify;
30   text-indent: 7%;
31 }
32
33 .paraNoIndent {
34   display: block;
35   margin-top: 0%;
36   margin-bottom: 0%;
37   text-indent: 0%;
38   text-align: justify;
39 }
40
41 .paraNoIndentB {
42   display: block;
43   margin-top: 0%;
44   margin-bottom: 4%;
45   text-indent: 0%;
46   text-align: justify;
47 }
48
49 .paraB {
50   display: block;
51   margin-top: 0%;
52   margin-bottom: 4%;
53   text-indent: 7%;
54   text-align: justify;
55 }
56
```

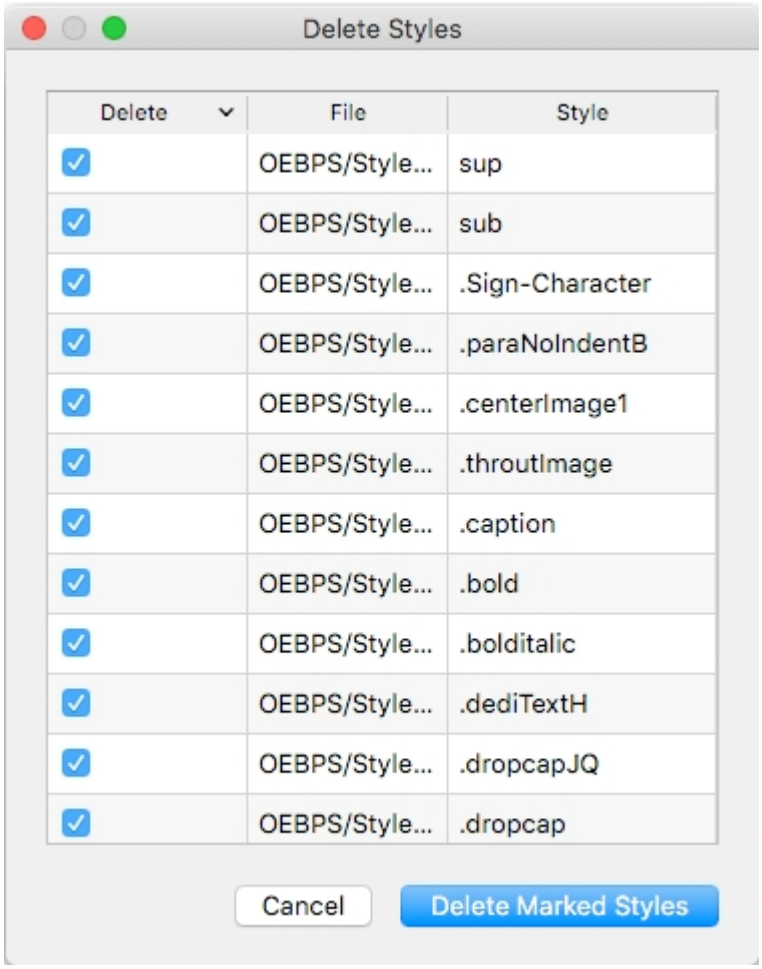
However, some stylesheets are formatted with each style listed on a **single line** with no spaces, resulting in a smaller file size:

style.css

```
6  .para{display:block;margin-top:0%;margin-bottom:
   0%;text-align:justify;text-indent:7%;}
7  .paraNoIndent{display:block;margin-top:0%;margin-
   bottom:0%;text-indent:0%;text-align:justify;}
8  .paraNoIndentB{display:block;margin-top:0%;margin-
   bottom:4%;text-indent:0%;text-align:justify;}
9  .paraB{display:block;margin-top:0%;margin-bottom:
   4%;text-indent:7%;text-align:justify;}
10 .centerImageH{display:block;margin-top:15%;margin-
   bottom:0%;text-indent:0%;text-align:center;}
11 .centerImage{display:block;margin-top:2%;margin-
   bottom:2%;text-indent:0%;text-align:center;}
12 .centerImageB{display:block;margin-top:4%;margin-
   bottom:4%;text-indent:0%;text-align:center;}
13 .centerImage1{display:block;margin-top:2%;margin-
   bottom:1%;text-indent:0%;text-align:center;}
```

Deleting Unused Styles

To automatically delete any unused selectors in stylesheets, select the menu item **Tools=>Delete Unused Stylesheet Selectors**. Sigil will present you with a list of styles in your stylesheets that do not have any references in your HTML files. Confirm that you want to delete the marked styles, or cancel.



While Sigil makes several checks to ensure a style is not being used, a complicated CSS declaration may list a selector incorrectly, so you should check the styles carefully.

To see a style in the CSS stylesheet, double-click on the style.

You can also use Sigil's Reports tool to review the "Classes Used" and "CSS Selectors" reports, which provides useful information on the styles in use. To view these reports, go to menu item **Tools=>Reports**. For more information, see the Reports chapter.

Indexes

— Create an Index For Your Book —

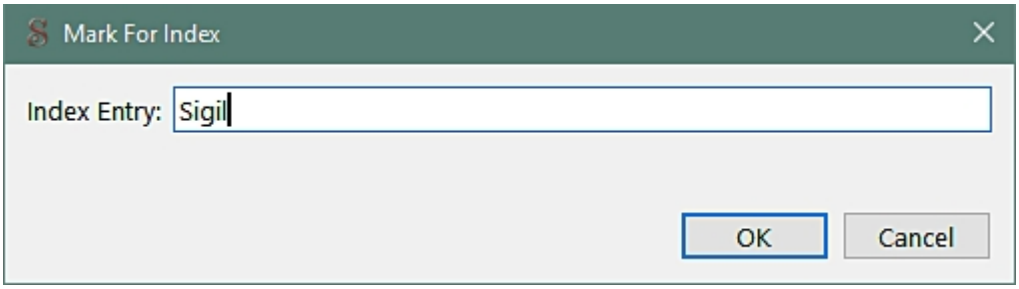
If you want to create an Index for your book, you can use Sigil to identify what to index and then to automatically create a new HTML page containing the index.

You can mark individual words in your text for indexing, or use the Index Editor to make sure every occurrence of the word is indexed.

Mark Entries for Indexing

The first way to identify entries to include in the index is to mark specific words or phrases in your text. Only the occurrence of the word or phrase you mark will be included in the Index – if you need all occurrences use the Index Editor.

Highlight a word or phrase in Code View and then select the menu item **Tools=>Index=>Mark For Index** or its keyboard shortcut (which you can change). This will then open a prompt for you to enter how you want these words to appear in the index:



The text you enter as the name to use in the Index will be show up in the Index as follows:

- **[blank]:** If you leave the name empty, then the exact words you highlighted will be used in the index.
- **words:** If you enter a word or phrase, then that word or phrase will be used (e.g. Oak instead of oak).
- **heading/words:** If you use a "/" in the name then a hierarchical group will be created (e.g. Forest/Tree/Oak will create an entry Oak filed under Tree filed under Forest).
- **entrya;entryb:** If you use a ";" then multiple entries will be created for this word or phrase.

When you mark text for indexing, it just adds add some Sigil-specific HTML code that tells Sigil to include these words when you create the index. In code view it might appear as follows:

```
<a class="sigil_index_marker" title="Character/Alice">Alice</a>
```

If you no longer want a word or phrase to be indexed by Sigil, you will need to delete the special HTML tags around the word.

> To find indexed words you can search in Code View. Or you can highlight them in Preview in color by adding the following CSS temporarily to your stylesheet:
>
> ```
> [class="sigil_index_marker"] { color:#335500; }
> ```

Add Entries to Index Editor

The second way identify words to index is to add the words to a list. Then when you create the index every occurrence of that word will be included in the index automatically. You can use words marked for indexing and a list of words to index at the same time.

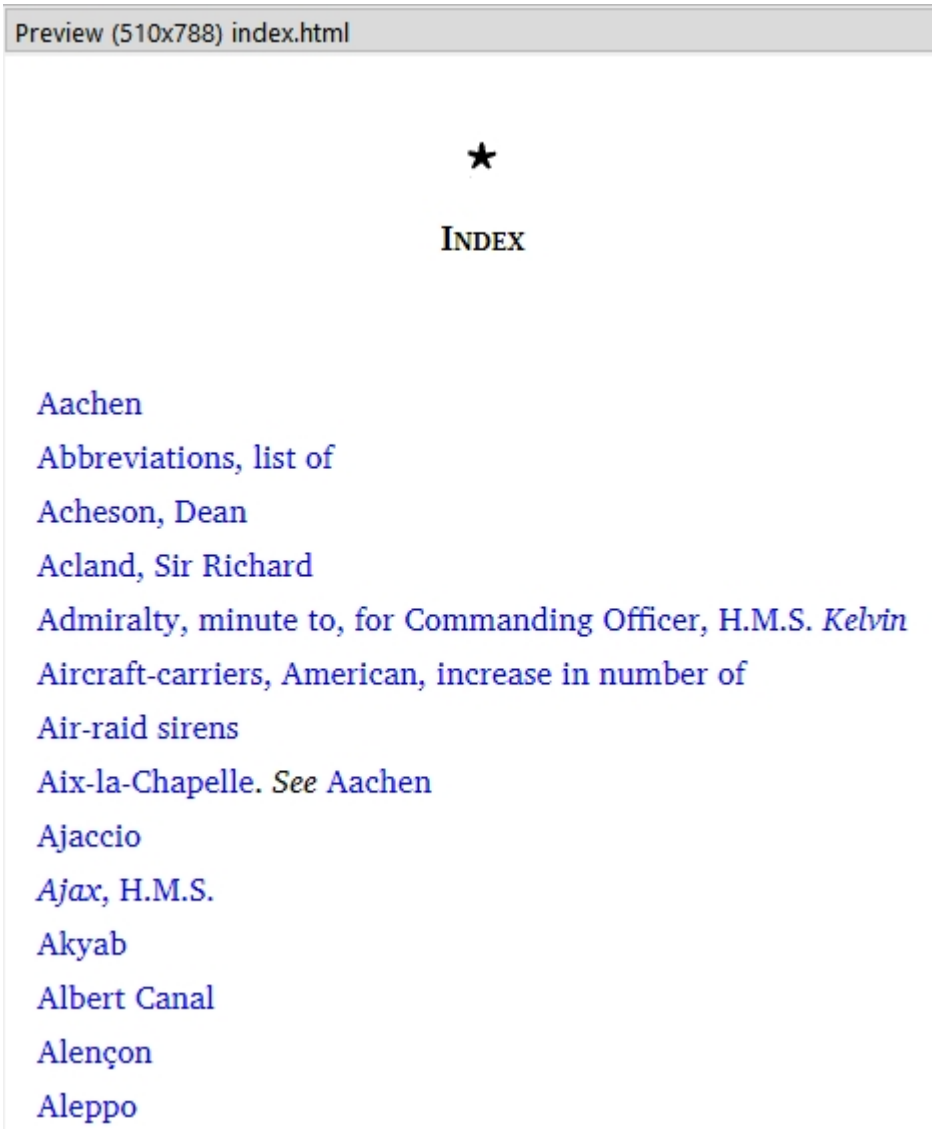
To add specific words to the index list, highlight the word or phrase, and then select the menu item **Tools=>Index=>Add To Index Editor**. If your selected text includes any HTML tags, you should remove them.

This will bring up the Index Editor with the word added to the list of entries to index. You can also enter words directly in the Index Editor as described below.

Create the Index

To actually create your index once you have marked words and/or added them to the list just select the menu item **Tools=>Index=>Create Index**.

This will create or update an HTML page called index.xhtml. It will be marked with the semantic type “index”. Entries are listed alphabetically, and, using the default styling, look like this:



You can edit the index page as with any HTML page, but if you re-create the index your changes will be over-written.

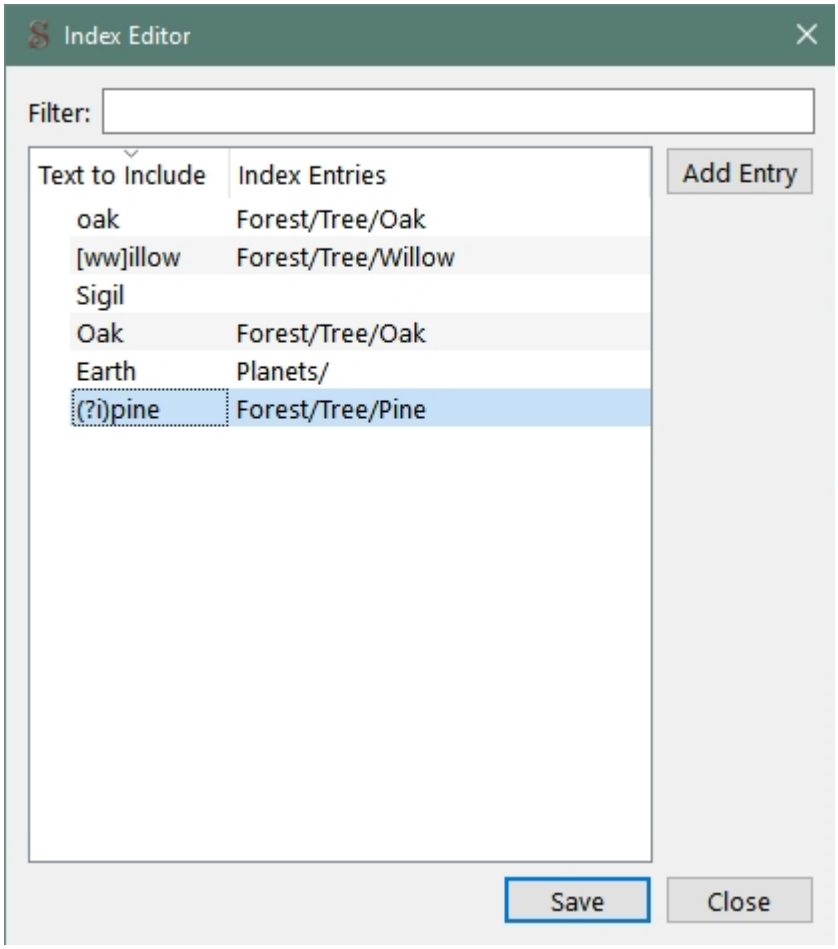
Custom Index Stylesheet

The index page uses a separate CSS file (sgc-index.css) to format the index. If the file already exists in your book it will not be overwritten (so you can customise it without losing your changes). But if it does not exist, it will look for sgc-index.css in your preferences directory and use that as the default file. If there is no file in your preferences directory, then Sigil will create a default stylesheet for the index.

Index Editor

The Index Editor is used to manage the list of entries that will be included in the index. It does not include the words you marked for indexing – they are included automatically.

To open the Index Editor select the menu item **Tools=>Index=>Index Editor** or its keyboard shortcut:



Text To Include

The "Text To Include" column is a list of patterns you want Sigil to search for in your text and then to create an Index entry for using the format in the "Index Entries" column. Entries in the "Text To Include" column can use the following formats:

- **words:** The words in the book must exactly match the listed words (case-sensitive). Do not include any HTML tags, just the words as they appear in Preview.
- **Regex:** You can enter most regex patterns – they will be match on a paragraph by paragraph basis. For example, to match words while ignoring case sensitivity you can use "[Gg]utenberg" or "(?i)Gutenberg".

You can paste text into the Text To Include field – e.g. from a file containing a list of words – and the text will be automatically split into separate entries for each line.

You can also use Open to open a text file of words (and entries separated by tabs).

Index Entries

The "Index Entries" column is what will actually appear in the Index for words matching the "Text To Include" column.

Entries in the "Index Entries" column can use the following formats:

- **blank:** If you leave the entry blank, then the words entered in the Text to Include column will be used for the entry's name.
- **words:** If you enter a word or phrase, then that word or phrase will be used (e.g. Oak instead of oak).
- **heading/words:** If you use a "/" in your name then a hierarchical group will be created (e.g. Forest/Tree/Oak will create an entry Oak filed under Tree filed under Forest).
- **heading/:** If you use a "/" at the end of the entry name, then a hierarchical group will be created and the words in the Text to Include column will be used as a sub-entry.

Context Menu

Most actions concerning editing clips can be done by using the context menu.

Select one or more entries then right-click on the selection for the following options:

- **Add Entry:** Add a blank search under the entry you clicked.

- **Edit:** Edit the entry.

- **Cut:** Delete one or more entries that can be pasted later.
- **Copy:** Copy one or more entries that can be pasted later.
- **Paste:** Paste any cut/copied entries.

- **Delete:** Delete entries.

- **Open:** Replace the current Index List with entries from a file (either an index file or a text file).
- **Reload:** Restore the list from the last saved state.

- **Save As:** Save all entries to a file for keeping a per-book list of index entries.
- **Autofill:** Fill the list with every word in the book.
- **Select All:** Select all entries (for easy deletion).

Save

To save your entries to disk, click **Save**.

To exit without saving your entries you can just close the window (with the close button or the Esc key). You can also use the context menu **Reload** to reload your entries from disk – which will delete your current entries.

> If you save and are running more than one copy of Sigil, then the other instances of Sigil will automatically reload the newly saved index entries to keep all copies in sync.

Filter

You can use the filter text box to restrict the list of searches shown to make it easier to find a search.

Opening or Importing Text Files

The **Open** context menu item can be used to open saved Index files, but it can also open text files.

You can list just the Text To Include words in the text file – one per line. Or you can include both the Text To Include and Index Entry values by separating them with a tab – one per line.

The Index Toolbar

To make creation of an index a bit quicker, users can use the **View=>Toolbars=>Index** menu to show the Index Toolbar.

- Open the Index Editor
- Add word to Index Editor

- Mark Text for Inclusion in the Index
- Generate the Index HTML file.

Reports

— View Details About Your Book —

Sigil provides several reports that list details about the files and css styles in your book.

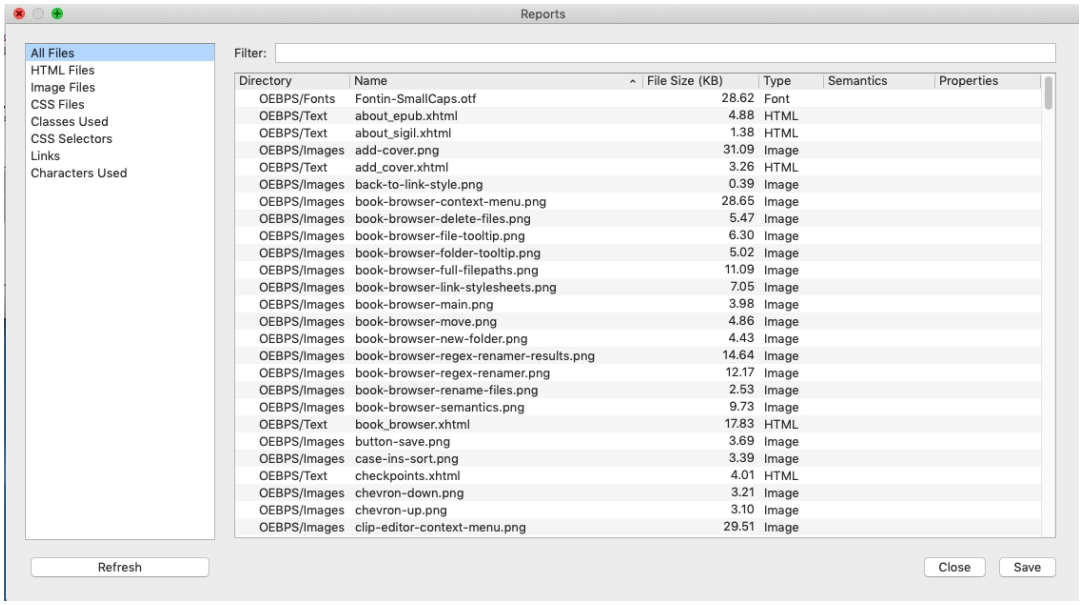
You can usually double-click on an entry in the report to jump to the filename listed.

> Some entries have tooltip information. For instance in the Image Report the number of times used column shows which files use the image.

Use **Tools=>Reports** to generate the reports and then select which report to view:

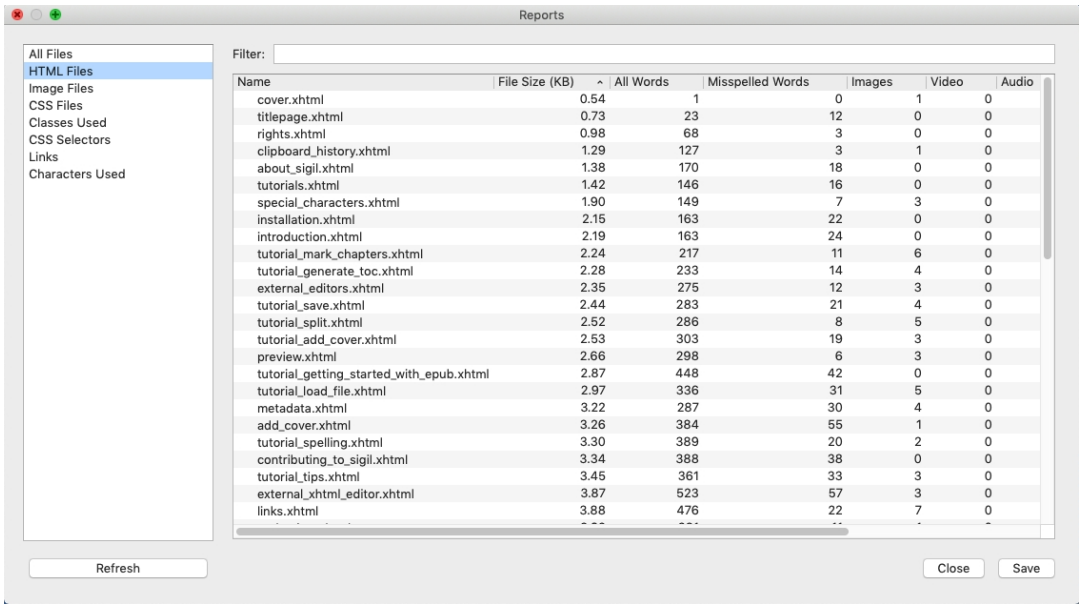
All Files

This report lists all of the files in the EPUB, the size of the file, the location and type of the file, and whether there are any guide semantic types applied to the file.



HTML Files

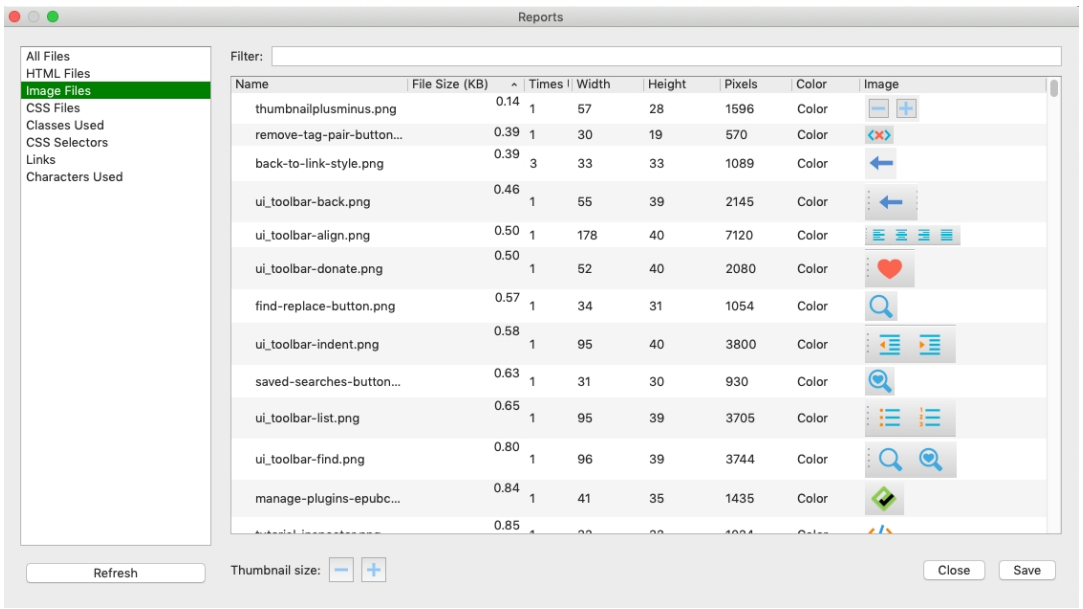
This report lists all of your HTML files and details about the files.



You can delete files from the EPUB while in the report by right-clicking on selected files and choosing **Delete From Book**.

Image Files

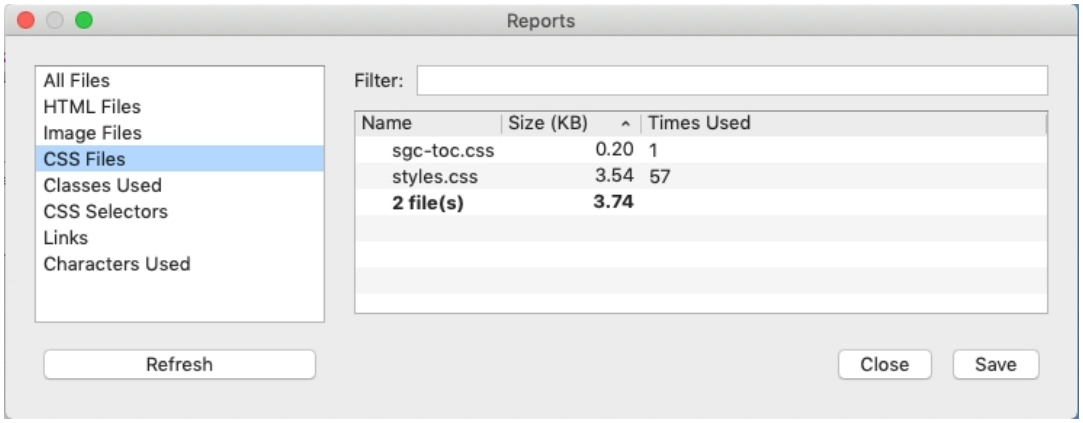
This report lists all of your images and indicates whether or not they were used in the book and how many times. You can double-click on an entry to jump to the first occurrence of the image in your HTML files (not images in CSS files).



You can delete files from the EPUB while in the report by right-clicking on selected files and choosing **Delete From Book**.

CSS Files

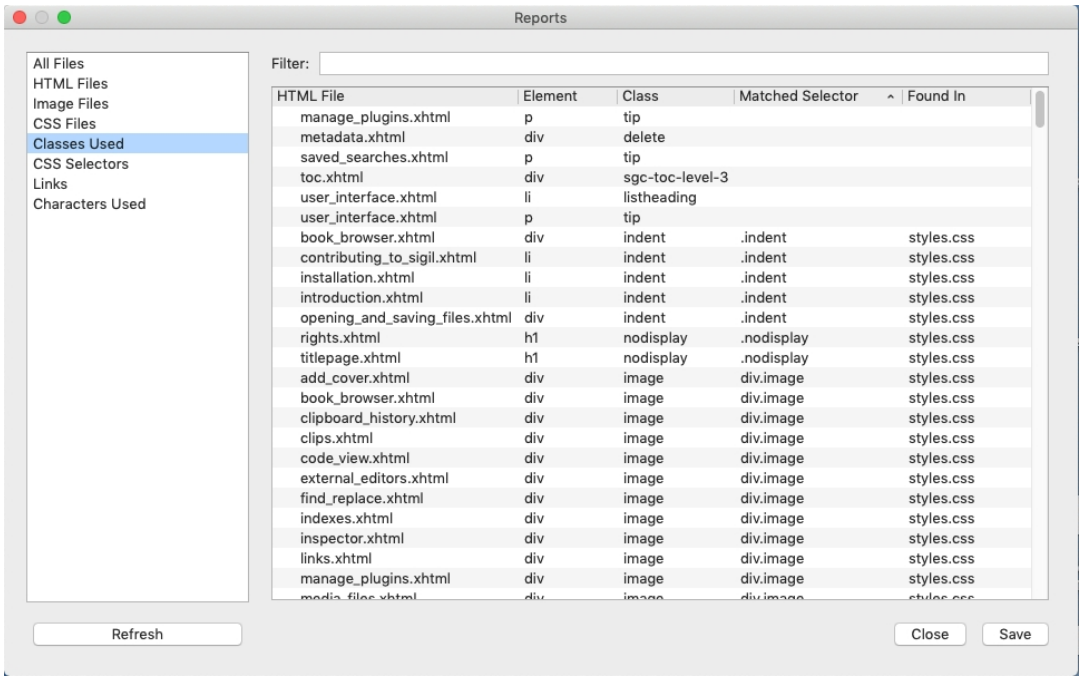
This report lists all of your CSS Stylesheets and indicates whether or not they were linked to and how many times.



You can delete files from the EPUB while in the report by right-clicking on selected files and choosing **Delete From Book**.

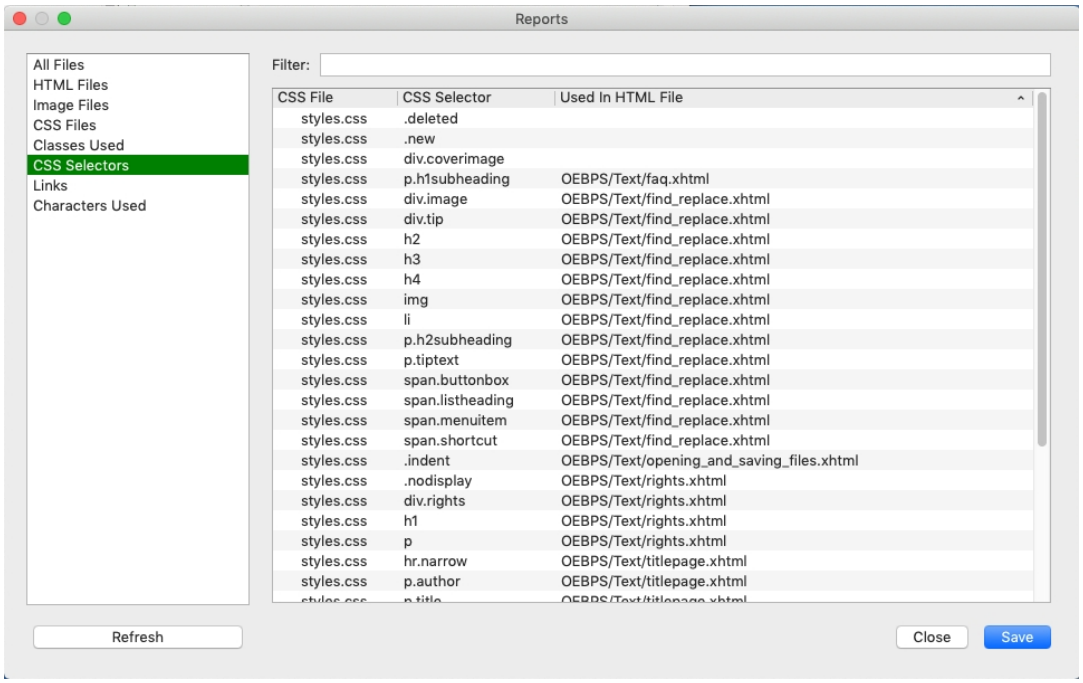
Classes Used in HTML Files

This report scans the HTML files in your book and extracts a list of style class names that are used in the files. For each class name it lists whether or not it was found in a CSS Stylesheet or an Inline Style.



CSS Selectors

This report scans the CSS stylesheets in your book and for each style selector it lists whether it was used in the book or not.



You can delete specific style classes from the stylesheets while in the report by right-clicking on selected styles and choosing **Delete From Stylesheet**.

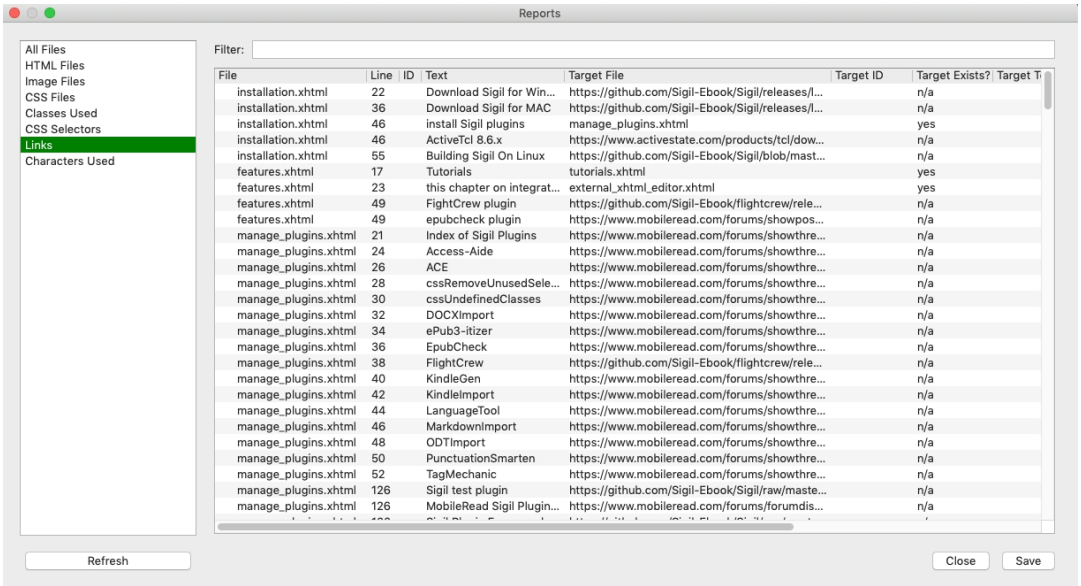
Links And Footnotes

This report lists every link (href) found in your HTML files. These are the "<a>" tags in your code. It includes whether the link has an id, what the target of the link is, and whether that target exists.

In addition, to help with footnotes or endnotes, it checks for reverse links. A reverse link is a link from a target that points back to the original link. In other words, if you create links like this for footnotes the report can confirm if they both link to each other:

```
Link in text:      <a id="t1" target="../Text/footfile1.xhtml#f1">text to footnote</a>
Link in footnote: <a id="f1" target="../Text/file1.xhtml#t1">footnote back to text</a>
```

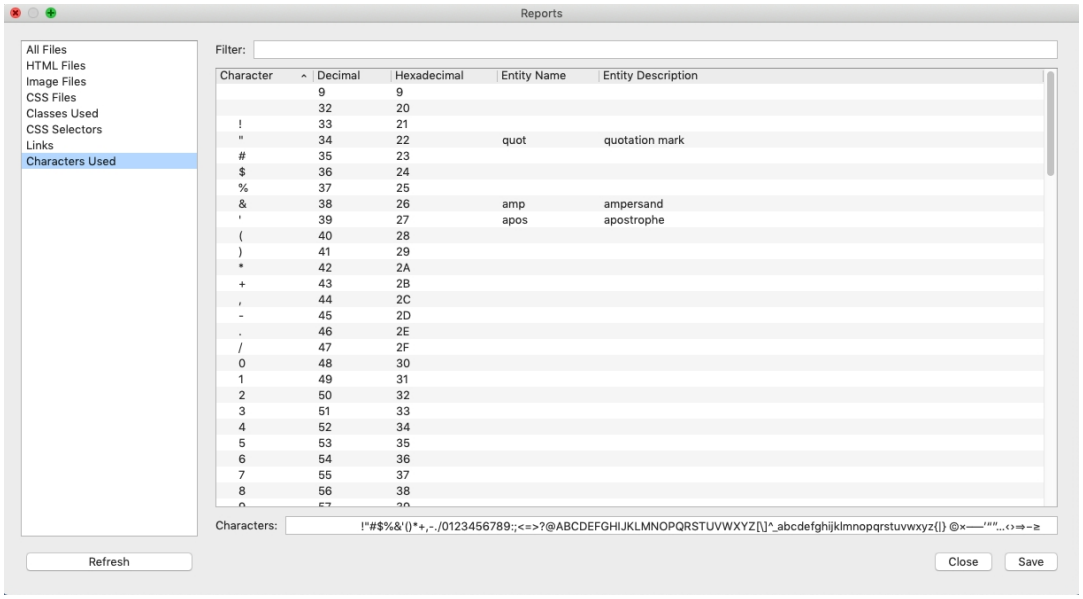
This only works if you use ids within the anchor ("a") tag.



Characters in HTML Files

This report extracts a list of all the characters that are used in your HTML files. More specifically, it lists any character that is displayed in Preview. HTML entity codes are listed if they exist for the character, along with their decimal and hexadecimal codes. And all characters are listed at the bottom for easy cut and paste.

This report can be used to identify if extra characters are being used in the book unexpectedly, and it may be of assistance if you are subsetting fonts.

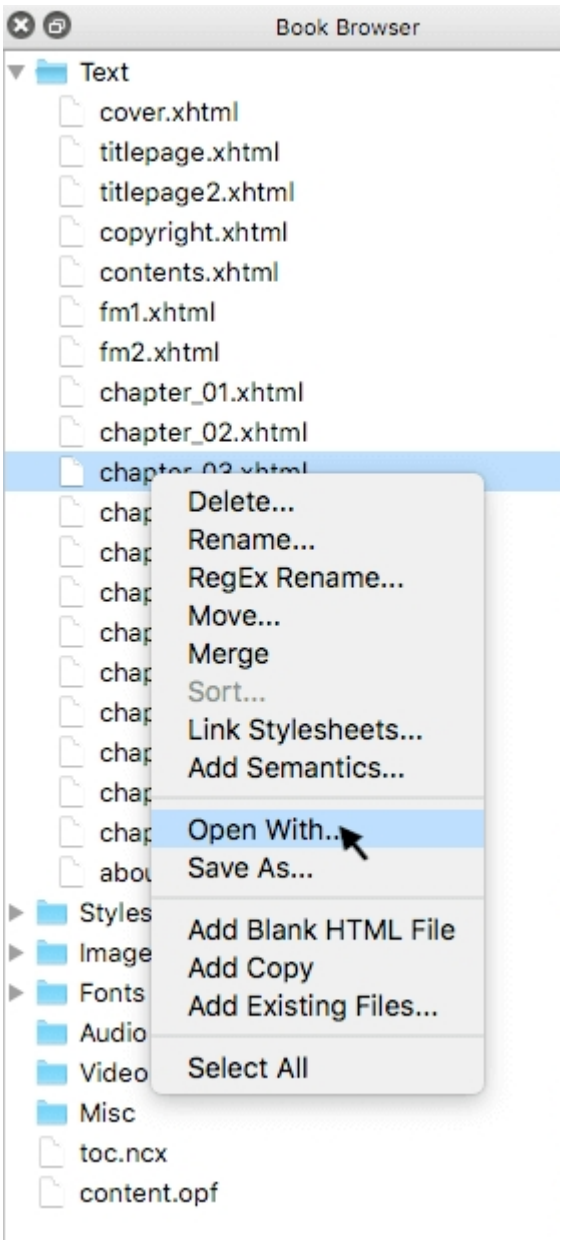

Reports
All Files
HTML Files
Image Files
CSS Files
Classes Used
CSS Selectors
Links
Characters Used
Filter:
Character
Decimal
Hexadecimal
Entity Name
Entity Description
9 9
32 20
! 33 21
" 34 22 quot quotation mark
35 23
$ 36 24
% 37 25
& 38 26 amp ampersand
' 39 27 apos apostrophe
(40 28
) 41 29
* 42 2A
+ 43 2B
, 44 2C
- 45 2D
. 46 2E
/ 47 2F
0 48 30
1 49 31
2 50 32
3 51 33
4 52 34
5 53 35
6 54 36
7 55 37
8 56 38
Characters:
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_abcdefghijklmnopqrstuvwxyz{|} ©×—‘“”…‹›⇒−≥
Refresh
Close
Save

External Editors

— Using Open With to Edit Files —

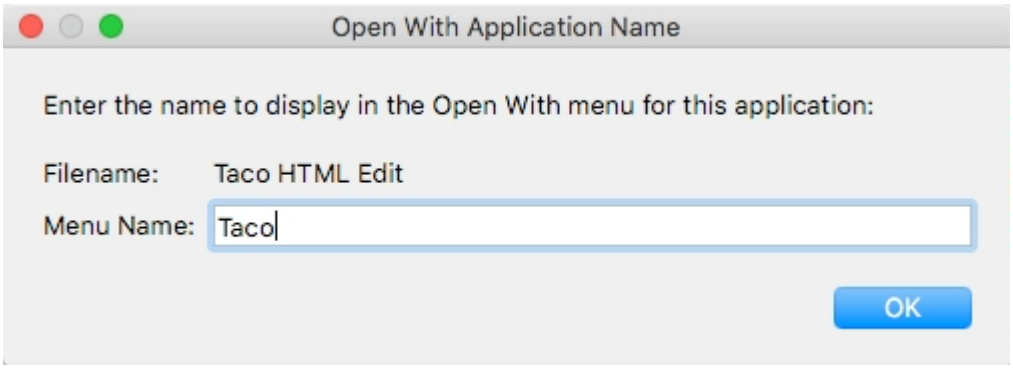
Sigil is a powerful editor, but it can't edit images, and it may not have some of the features you use in other editors. When this happens, you can use Sigil's Open With feature to open and edit files in your preferred external program – and when you save, Sigil will immediately and automatically update its copy. This is very useful if you want to quickly edit images or CSS stylesheets and immediately see how changes affect the layout of your book (in Preview), all while keeping an HTML file open in Code View.

To edit a file in an external program, right-click on the file's name in the Book Browser and choose **Open With** from the context menu:

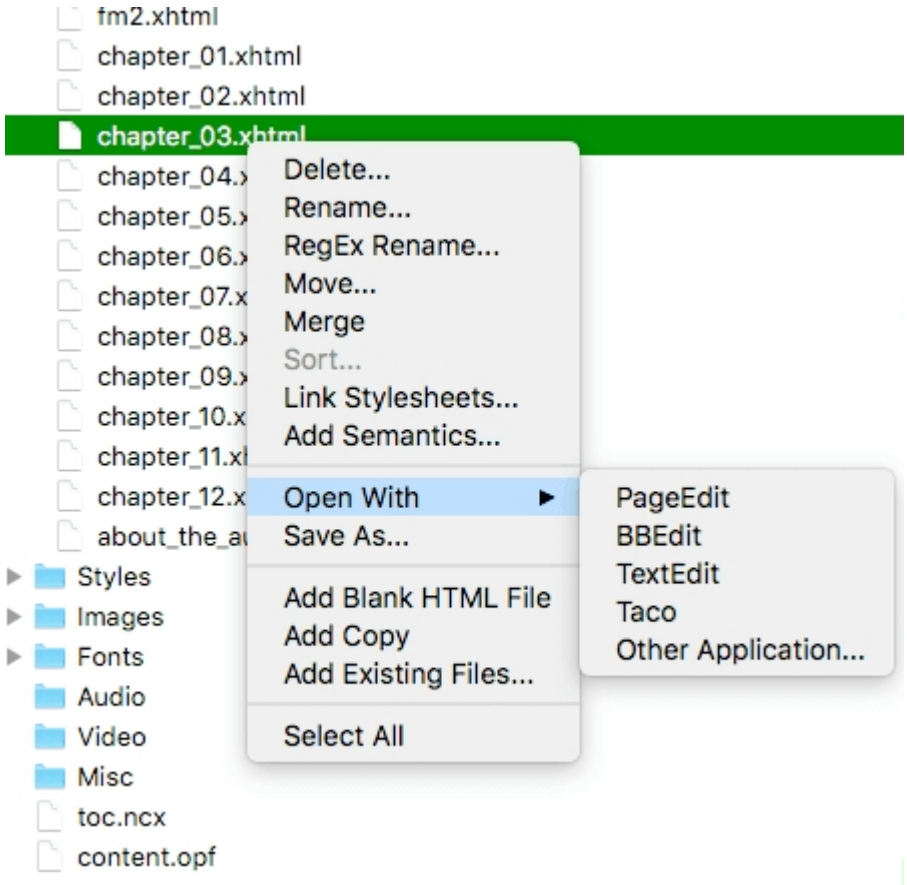


The first time you open a file type this way, you will need to choose an application to use with that file type. You can define and save up to five different external applications for each file type (.html, .css, .jpg, .png, .opf, etc.).

You can then give the application a friendly name that will be used in the menu the next time you use **Open With**:



When you next use **Open With** you will see the friendly names you've given your existing applications, and the option to choose another application (up to five per file type).



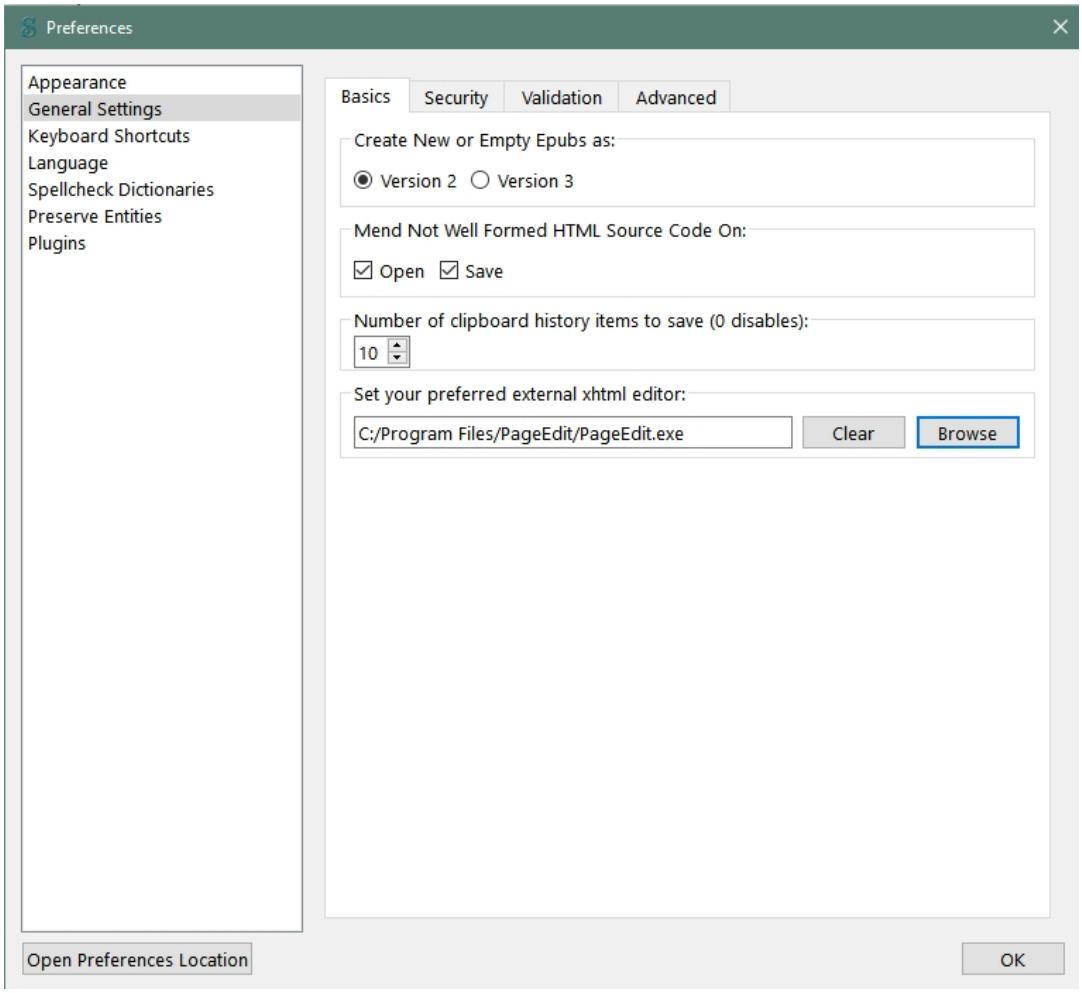
If you make changes to a file in Sigil while the file is being edited in an external program, you may lose data. When using an external editor, always make the changes, then save and close the file in the external application before making any changes to that file in Sigil.

Preferred External XHtml Editor

– Integrating PageEdit WYSIWYG Editor into Sigil –

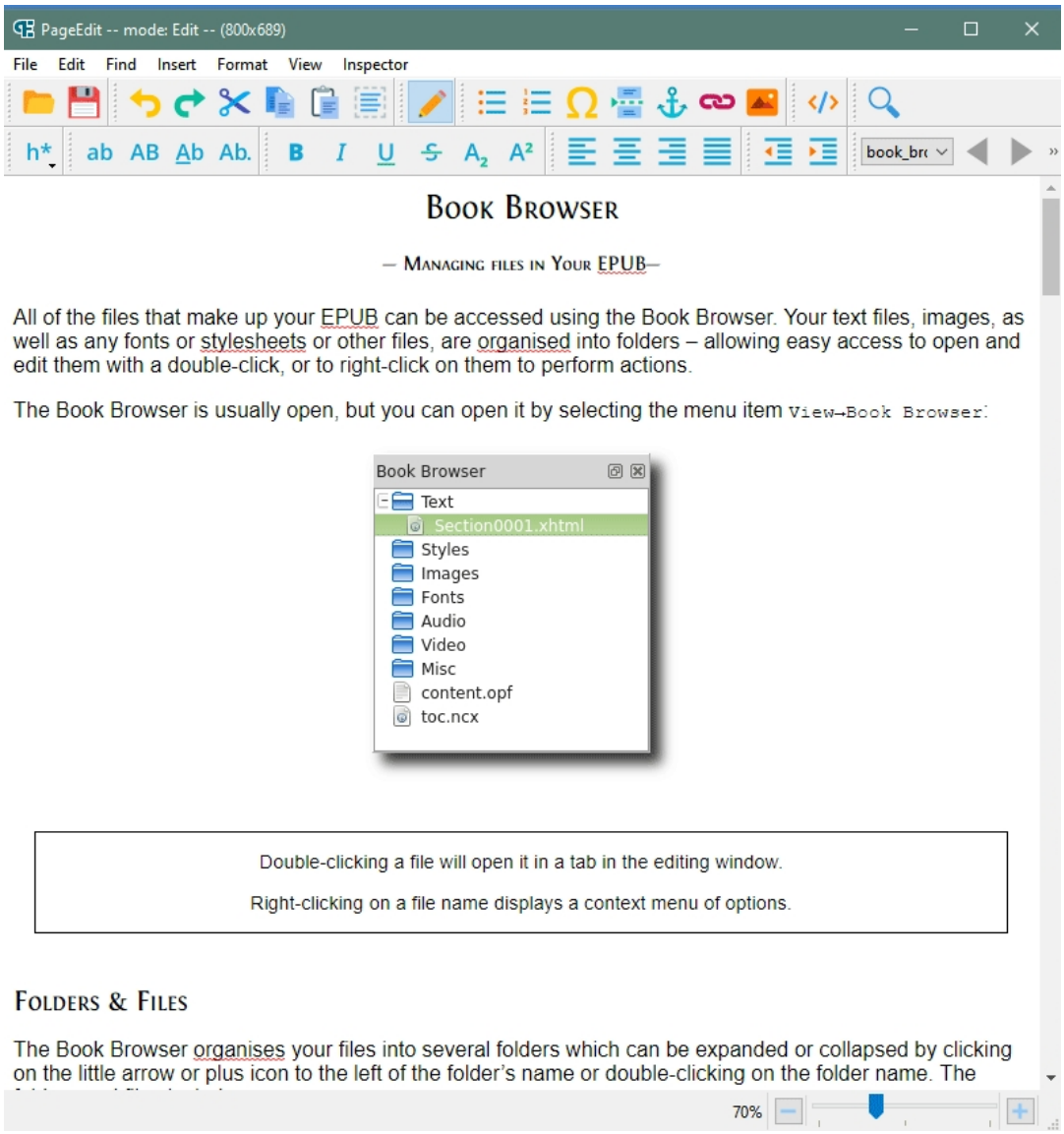
WYSIWYG editing mode – referred to as BookView in older Sigil User Guides – was removed from Sigil in version 0.9.15 because that editing mode used live html editing based on the QtWebEngine/Chrome browser engine that often resulted in incorrect or broken xhtml. Since this browser engine was embedded in Qt, we had no way of changing or fixing that behaviour. This broken xhtml was often rejected by epubcheck and therefore most publishers would not accept it. Sigil 0.9.14 is the last version of Sigil to include the BookView mode of editing.

If you need still need the ability to do minor editing and proofing in a WYSIWYG environment, the maintainers of Sigil have created a standalone app called PageEdit that can be launched via Sigil's Preferred External XHtml Editor tool by setting the path to the external editor's executable in Sigil's preferences (**Edit=>Preferences=>General Settings=>Set your preferred external xhtml editor**).



Users can use the icon on the Sigil toolbar to launch the file/chapter in Sigil's current Code View tab inside of PageEdit. Users can also use/assign the keyboard shortcut for the button to launch the editor (defaults to F2).

When editing is complete in PageEdit these edits are then saved back into Sigil so they appear in Code View and can be validated and corrected if needed.



The editing interface to PageEdit is very similar to the old BookView interface with toolbars and menus to format text, set paragraph alignment, cut and paste, insert existing images, insert links to existing xhtml files, move forward or backwards by chapter, and etc. Page edit can be used in either of two modes: Preview Mode to enable navigation by clicking links, and Edit mode to enable editing of content. PageEdit always opens in Edit mode. Toggling this mode is done by hitting the pencil icon in PageEdit.

If the user would like to edit multiple xhtml files from within Sigil at the same time, the OPF (ie. the content.opf) can be selected and opened via Book Browser before hitting the External Editor Icon. This will enable all Chapters/Files listed in the OPF spine element to be edited. PageEdit will provide a simple pulldown menu to allow the user to select which file to work on. All edits can be saved back into Sigil. Using this approach, PageEdit makes it easier to proofread an entire EPUB in chapter order, making minor edits or fixes as needed.

> Just like when using Sigil's Open With features, if you make changes to the file in Sigil while the file is being edited in PageEdit then you may lose data. You should always open the file for editing externally, make the changes, then save and close the file in PageEdit before making any changes to that file in Sigil.

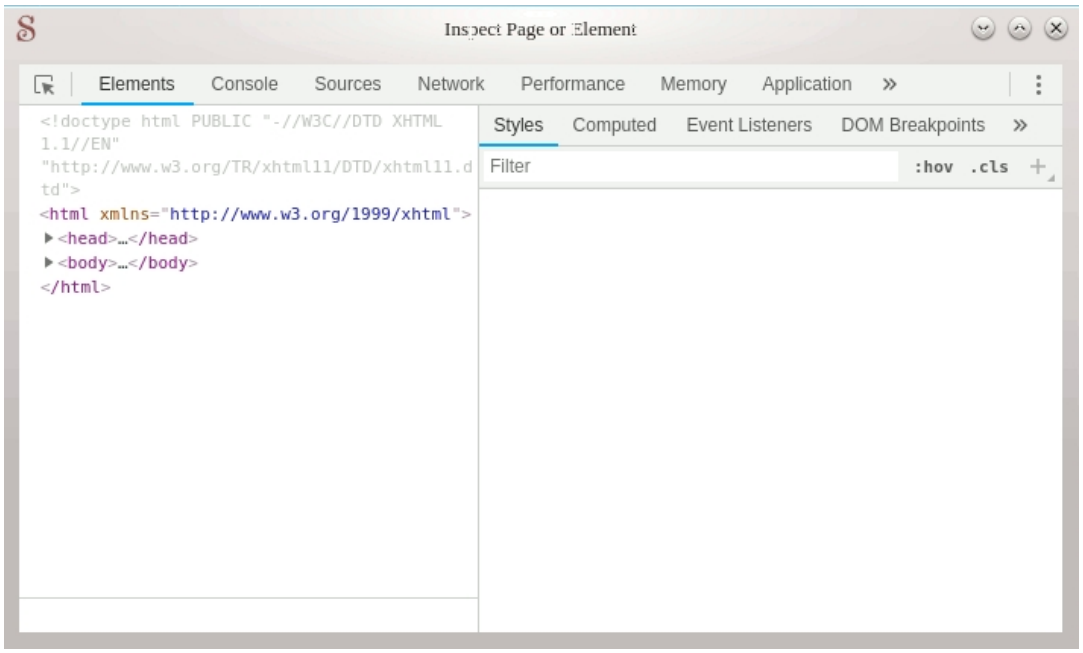
PageEdit is available for all 3 main platforms including Linux, Windows, and macOS. It is free and can be downloaded from Sigil-Ebook's GitHub repository for PageEdit. Linux users should check their local repositories for PageEdit in the same places they find Sigil itself.

Using the Page Inspector

— Inspect the DOM of Your Page —

At the bottom of the Preview window is a button that displays the Inspector tool. The icon looks like this:

When you click on the button, the Page Inspector tool is displayed.



The Page Inspector allows you to inspect and modify the Document Object Model (DOM) of the page being viewed in the Preview window.

The DOM is a memory-resident version of the page that you have composed using HTML. Sigil parses the HTML file and creates a tree structure in memory that represents the information you have entered into the HTML. The Preview Page is actually displaying what is in the DOM.

If you are interested in learning more about the DOM, you can find those details at:

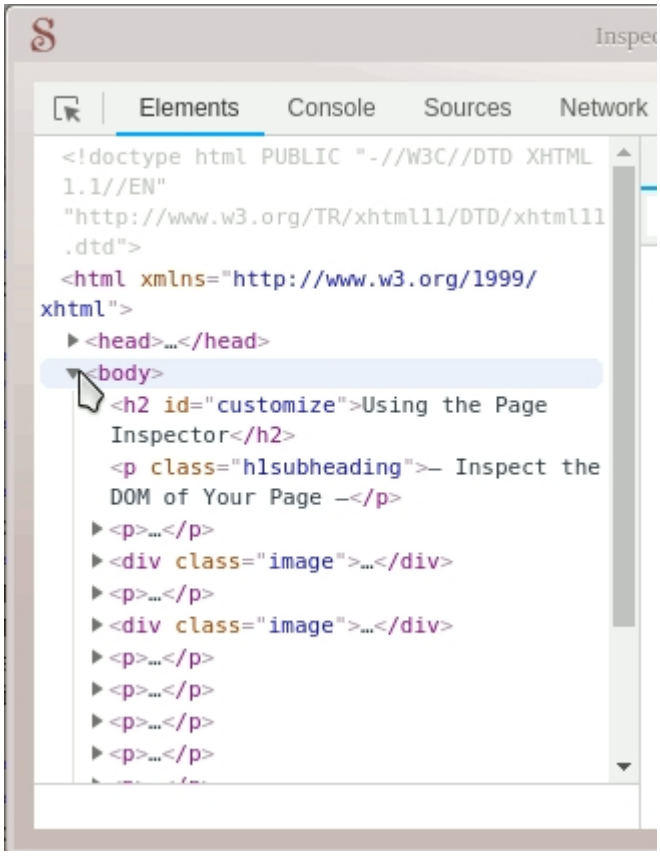
https://www.w3schools.com/js/js_htmldom.asp

Since the DOM is a memory-resident version of your page, you can change it on the fly and see the results immediately in the preview.

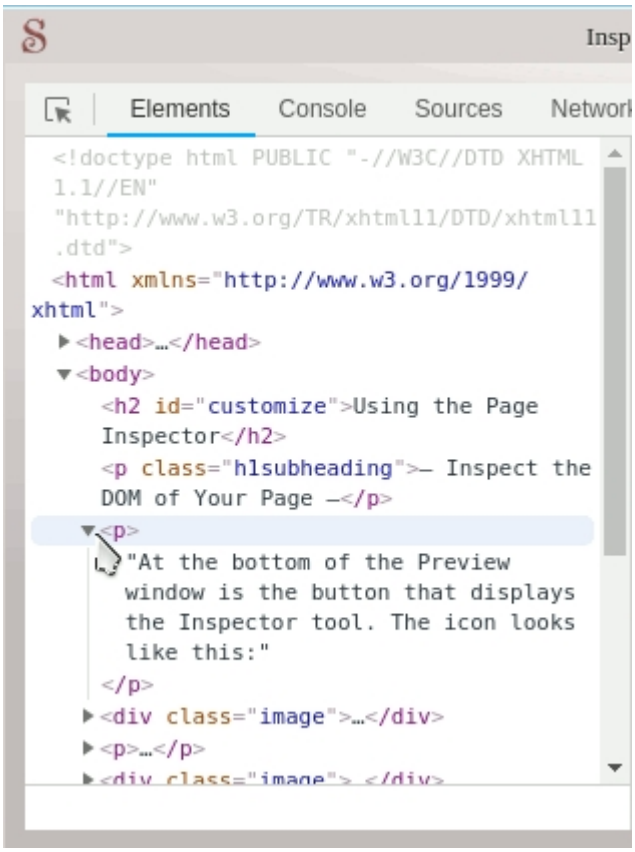
The Inspector allows you to inspect and modify the DOM as you like. That makes it possible for you to change the contents of the page, and/or the style sheets and see the results of those changes immediately in the preview.

> Since you are changing only the DOM in memory, none of the changes that you make in the Inspector will change your HTML file.

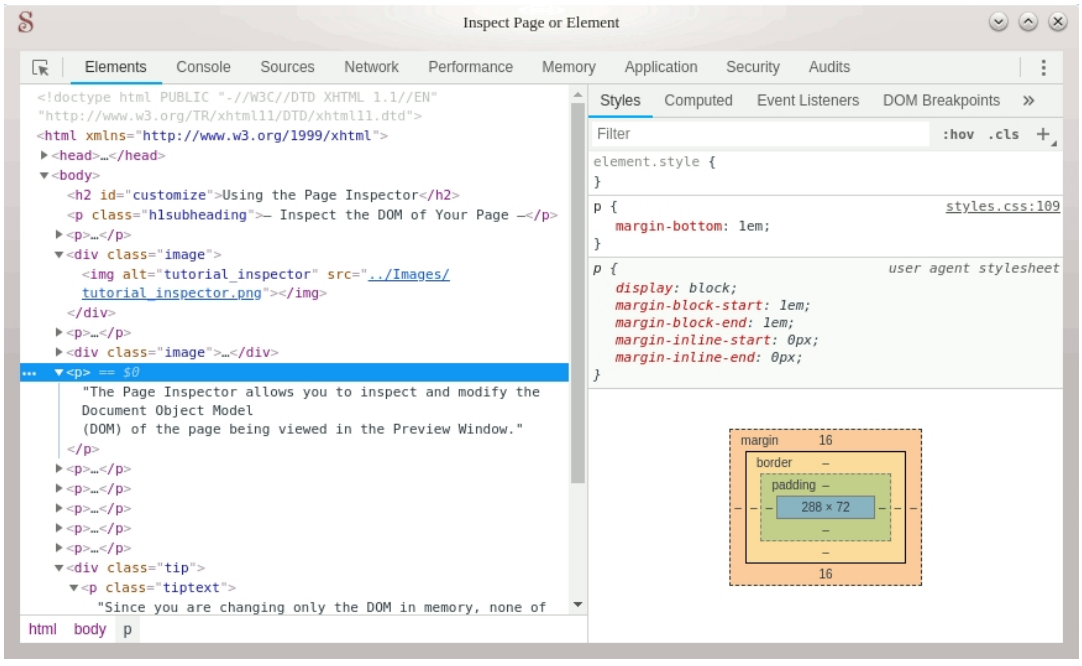
When you first open the Inspector, the DOM tree shows two nodes with arrows, ▸ , pointing to the right, as you can see in the previous figure. There is one for the <head> tag and one for the <body> tag. If you click on one of the arrows, the tree expands so show the contents of that node, as shown below.



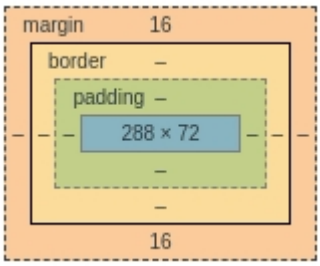
In the figure above, you can see that the <body> tag has been expanded to show the contents of the body. Clicking successively on the arrows allows you to dig down through the tree of nodes.



When you select one of the tags in the tree, the window on the right hand side displays the style information about that node.

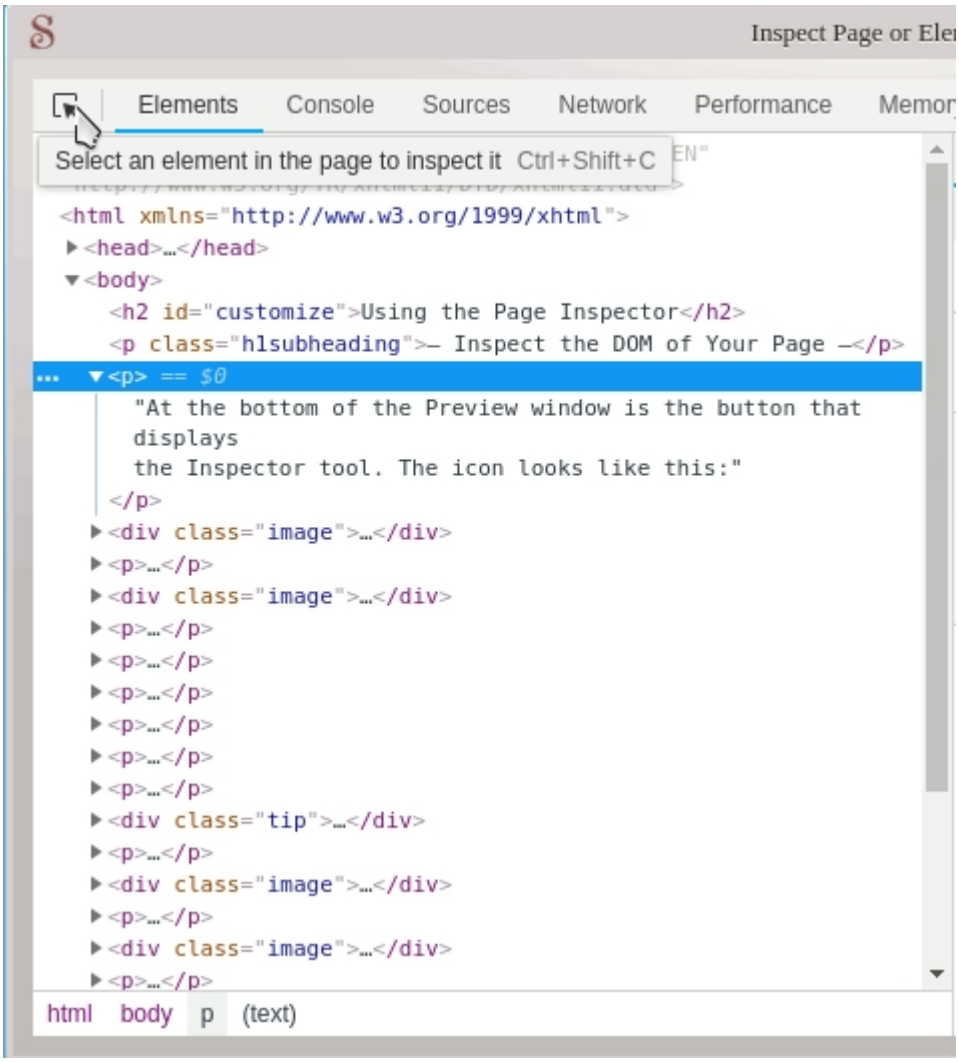


You can see that the styles applied to that node are shown in the Styles panel. Of particular interest is the figure in the lower right hand corner of the Styles panel. That is called a Box Model.

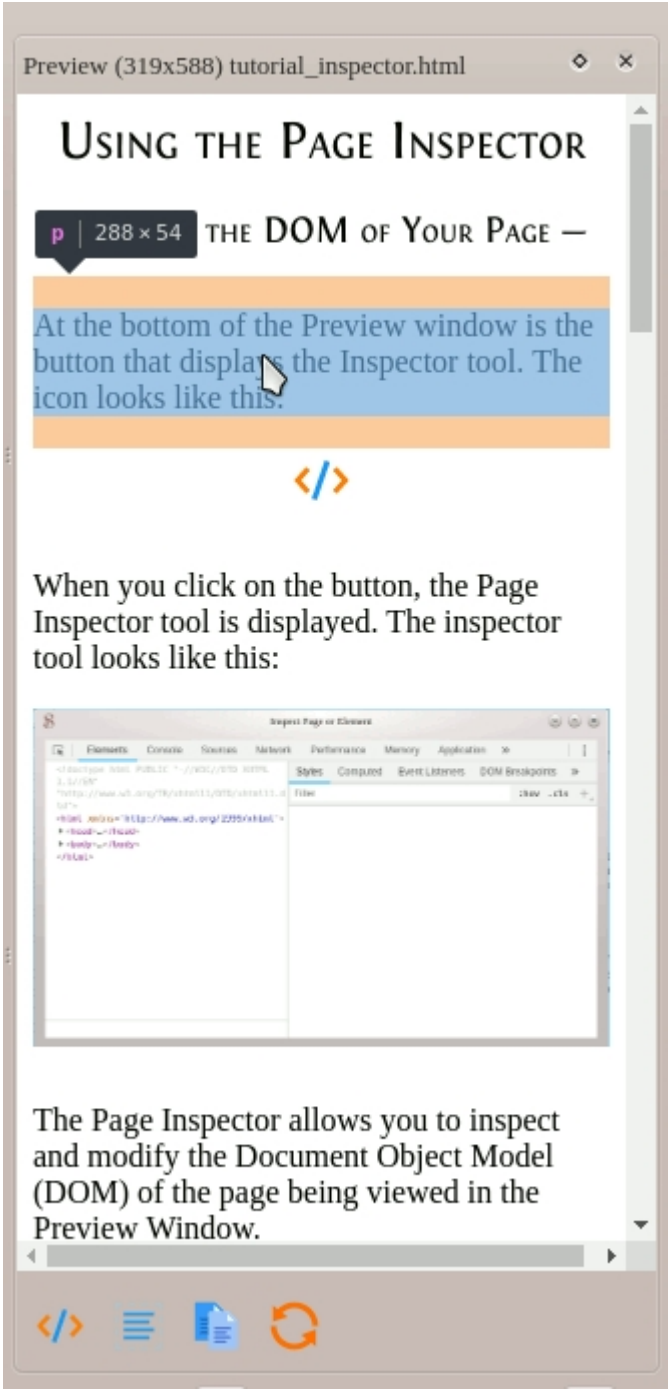


The box model shows the values of position, margin, border, padding and contents of the node. The units of the numbers shown are pixels. So you can tell from the image above that the selected paragraph has a top and bottom margin of 16 pixels, no border, no padding, and is 288 pixels wide and 72 pixels tall.

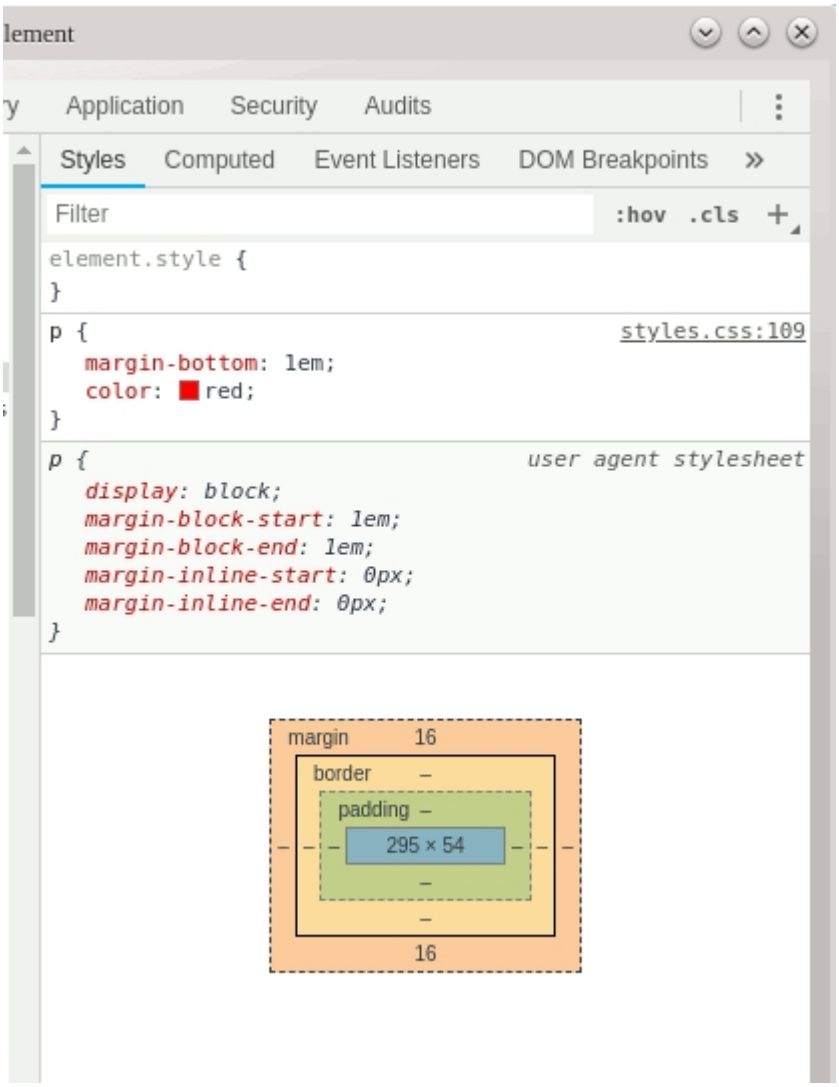
Another way that you can select a node for inspection is to use the Element Tool, which allows you to select the desired node in the Preview panel. Activate the Element Tool by clicking the button in the upper left corner of the Inspector window, indicated by the mouse cursor in the figure below.



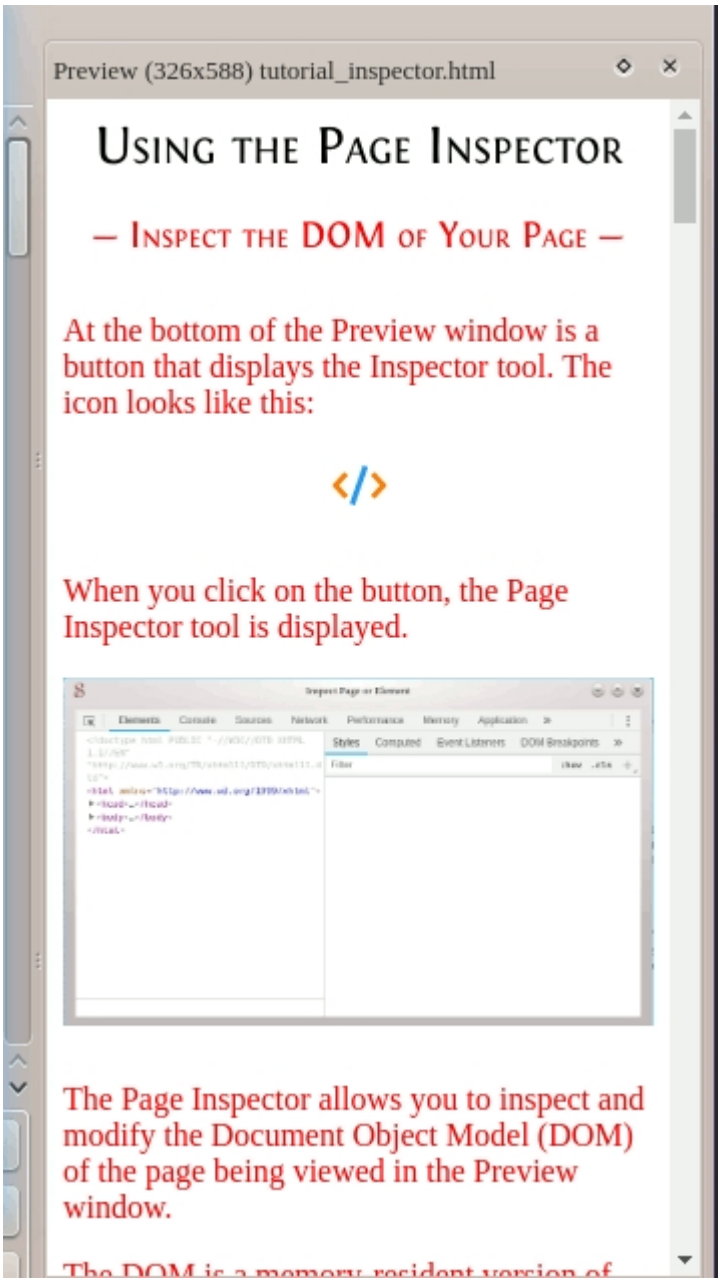
Once you have activated the Element Tool, you can select the desired element in the Preview panel. The elements in the Preview panel are highlighted as you track the mouse cursor over them.



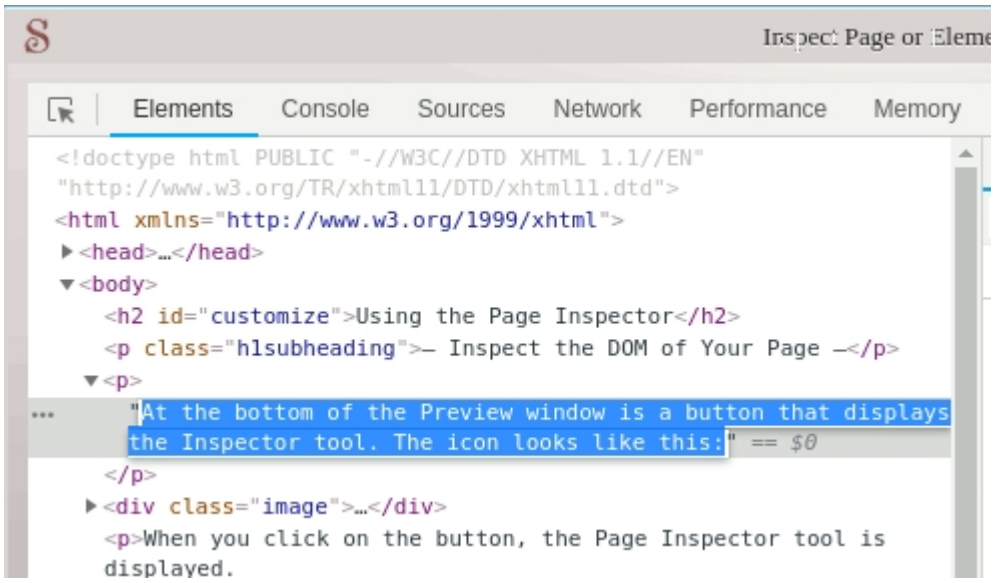
Simply click on the highlighted element. Once you have selected a node in the Elements tree, you can make changes to its contents and styles and see the effect of those changes immediately in the Preview window. In the figure below, I have changed the style for the paragraph tag to add the color: red.



You can see immediately in the Preview window that all the paragraph tags show the text in red.



You can change the contents of the node as well. Double-click on the contents of the paragraph to select it.

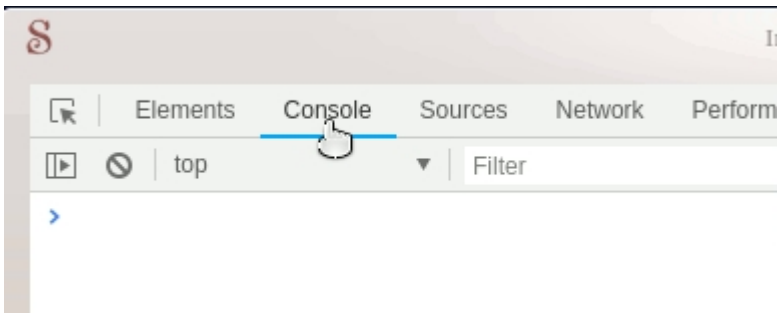


Enter the new text to see it change immediately in the Preview window.

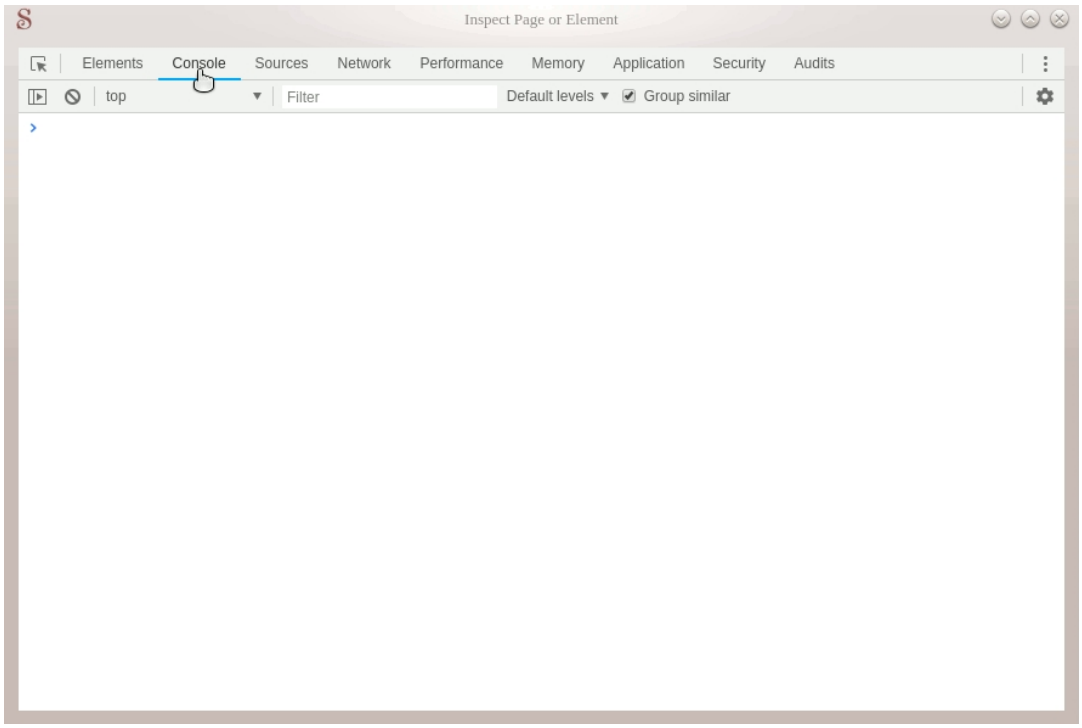
> Remember, since you are changing only the DOM in memory, none of the changes that you make in the Inspector will change your HTML file.

Another feature of the Inspector is the Console, which interprets JavaScript commands.

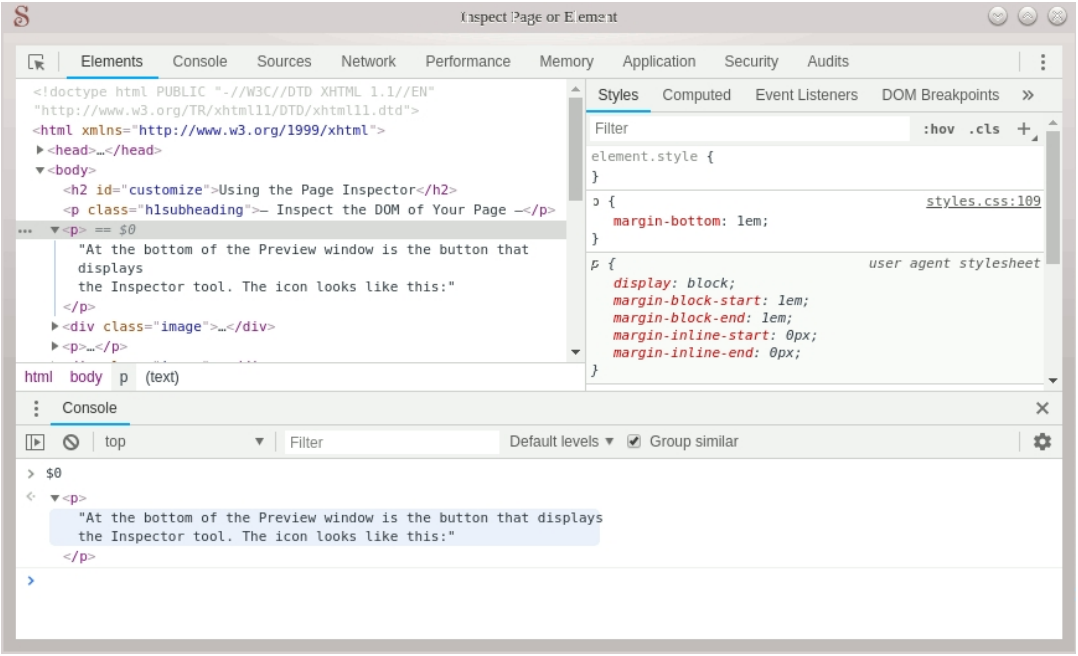
Once you have selected a node in the inspector tree, you can access it using normal JavaScript commands in the Console. The Console can be activated by switching to it with the Console tab.



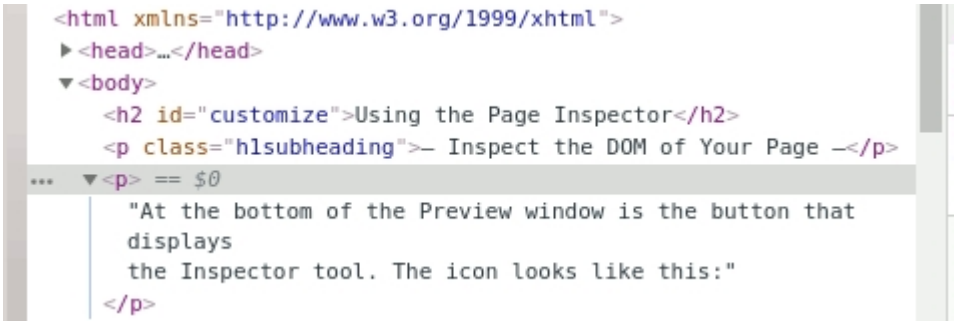
That results in the Console taking over the entire Inspector window.



You can also place a Console panel beneath the Elements panel by typing the Esc key.

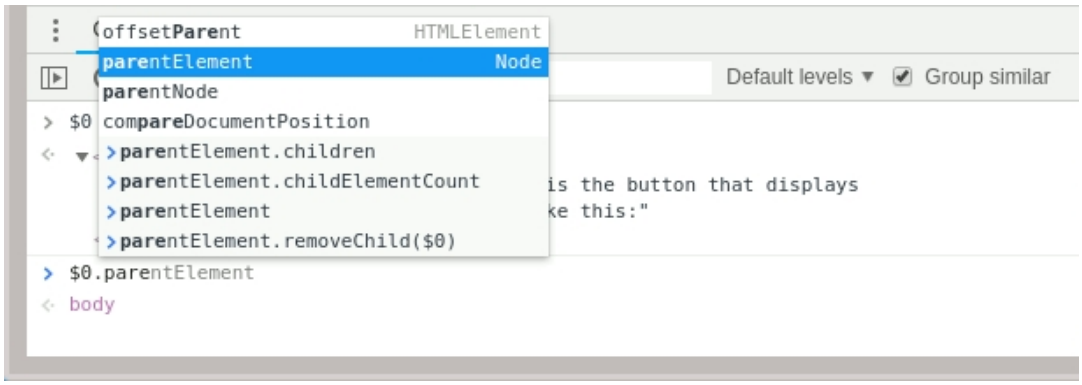


The currently selected node in the Elements panel is accessed using '$0'. The node that is accessed using $0 is indicated in the Elements panel where the tool has appended '== $0' to the line of text that is selected.



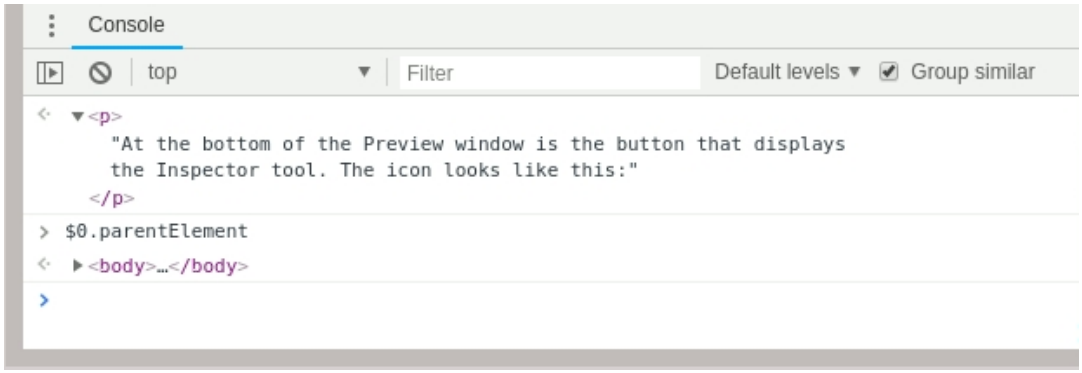
In the previous figure of the Console, you can see that I have typed '$0' into the Console, and it is displaying the contents of the selected node. Clicking on the arrows in the Console allows you to dig down into the node that is displayed.

You can use $0 to access and traverse the DOM tree as you normally would with JavaScript commands. For example you can access the parent of the selected node by typing $0.parentElement. As you enter the command, autocompletion popups are displayed to help you.

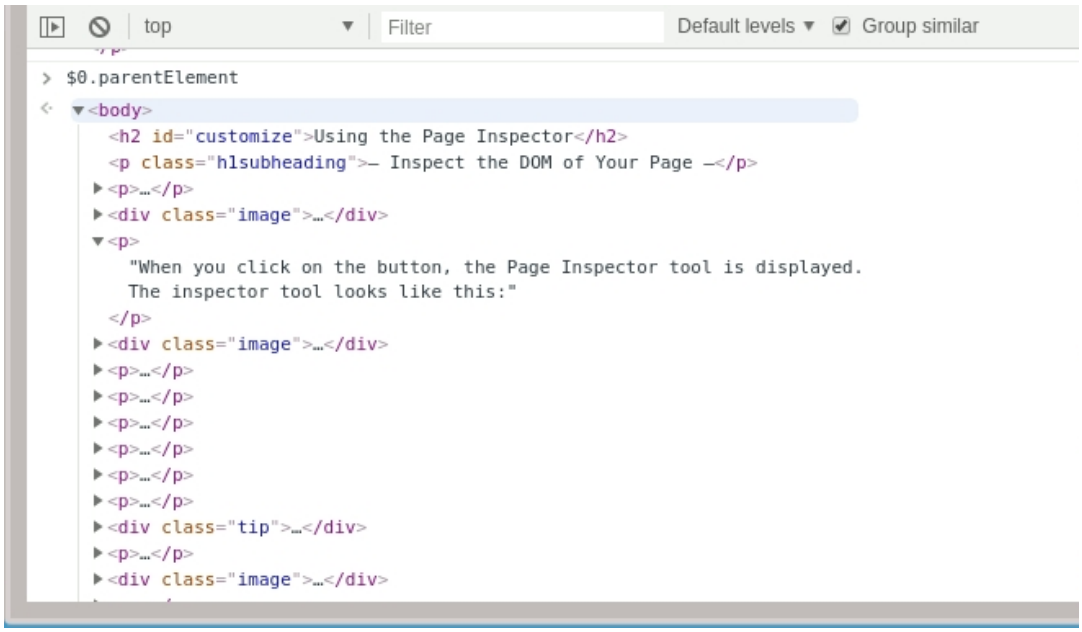


You can use the mouse to select the desired method from the menu, or you can type the Tab key to finish the selection by accepting the highlighted element.

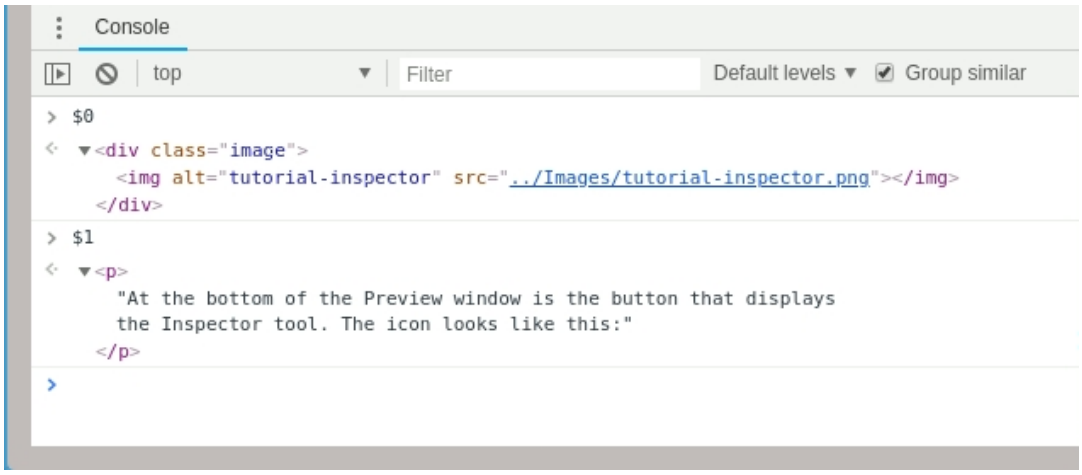
You can see in the following figure that the parent of the selected paragraph node is the <body> node.



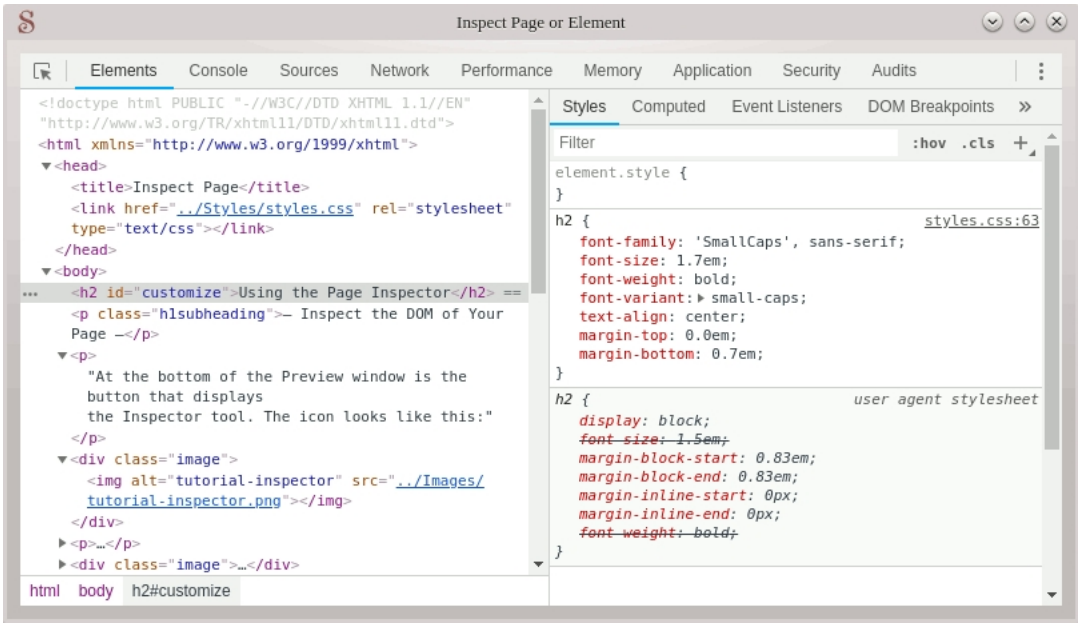
Once again, you can use the arrows in the Console output to dig down into the node that you are inspecting.



One last tip for using the JavaScript console is that the Inspector tool keeps a history of selected nodes. The most recently selected node is accessed using $0, and the node selected before the current selection is accessed as $1. In the figure below, I first selected the paragraph node, and then I selected the image node. So $0 refers to the image node, the most recent selection, and $1 refers to the paragraph node, the previous selection.



Cascading style sheets are also supported in the Inspector. The Styles panel provides information about cascading styles.



In the figure above, the cascading styles are shown with information about the source of each style. The styles that are crossed out with a line through them, have been superseded by the other styles in the display.

This tutorial is only a brief overview of the features of the Page Inspector. More details can be found at Google Chrome Developer Tools Documentation

Manage Plugins

— Extend the Functionality of Sigil —

Python 3 plugins provide a powerful way to extend the functionality of Sigil. On Windows and macOS, Sigil already comes with an embedded Python 3 interpreter and several Python libraries optimized for Sigil, so you won't have to install Python 3 on your system. Just download the plugins, add them to Sigil, and you're ready to go.

> On Linux, users should install the relevant python 3 packages required to use plugins on Sigil. Typically these are the suggested (not required) packages for Sigil in your distribution's repository. Check with your distribution's Sigil maintainer for details.

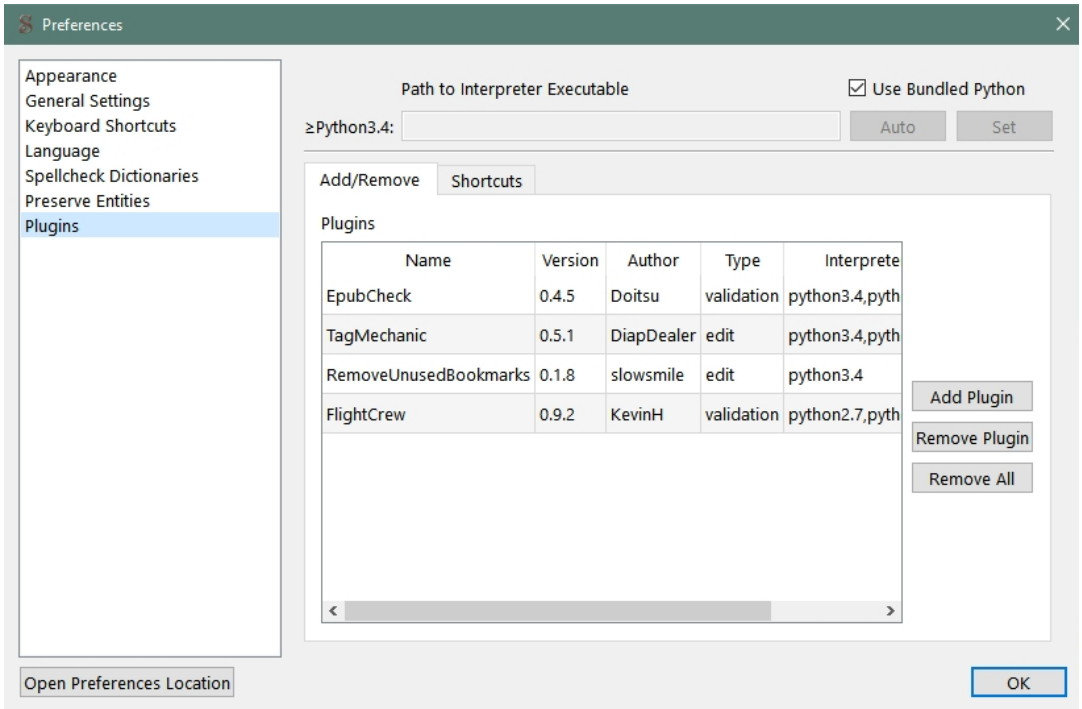
A complete list of supported plugins can be found in the Index of Sigil Plugins on MobileRead. Highlights include:

- **Access-Aide**: Improves the accessibility of EPUBs to help meet ACE (Accessibility Checker for EPUB) requirements.
- **ACE**: A simple DAISY ACE (Accessibility Checker for EPUB) wrapper.
- **cssRemoveUnusedSelectors**: A powerful tool for removing unused CSS selectors.
- **CSSValidator**: Validate CSS files using the *offline* version of the W3C CSS Validator.
- **cssUndefinedClasses**: Removes all classes and ids not referenced in stylesheets or elsewhere.
- **DOCXImport**: Imports DOCX documents into Sigil as EPUB.
- **ePub3-itizer**: Converts a valid EPUB2 into an EPUB3.
- **EpubCheck**: A simple EPUBCheck wrapper to validate EPUB2 and EPUB3.
- **FlightCrew**: A validator for EPUB2.
- **KindleGen**: A simple kindlegen wrapper to generate master mobi, azw3, mobi7, or KFX from EPUB.

- **KindleImport**: Imports Kindle KF8/azw3 and older-style mobi-only files into Sigil as EPUB.
- **LanguageTool**: A simple LanguageTool wrapper to check grammar.
- **MarkdownImport**: Imports Markdown files and plain text files without markup into Sigil as EPUB.
- **ODTImport**: Imports OpenOffice ODT files into Sigil as EPUB.
- **PunctuationSmarten**: Converts quotes, apostrophes, dashes, and ellipses to their typographic equivalents.
- **TagMechanic**: Removes spans, divs, and a few other tags; changes their attributes; and/or converts them to different html elements.

… and many more!

Install Plugins



To install a plugin:

1. Download the desired plugin .zip file.
2. Go to **Plugins=>Manage Plugins=>Add/Remove** then click the **Add Plugin** button.
3. Select the plugin .zip file and click **OK** (on Mac, click **Open**).

4. Enable **Use Bundled Python**, if it isn't already enabled.
   (If Use Bundled Python can't be selected, click **Auto** or **Set** to select an external Python interpreter.)

Don't unpack or rename plugin .zip files!

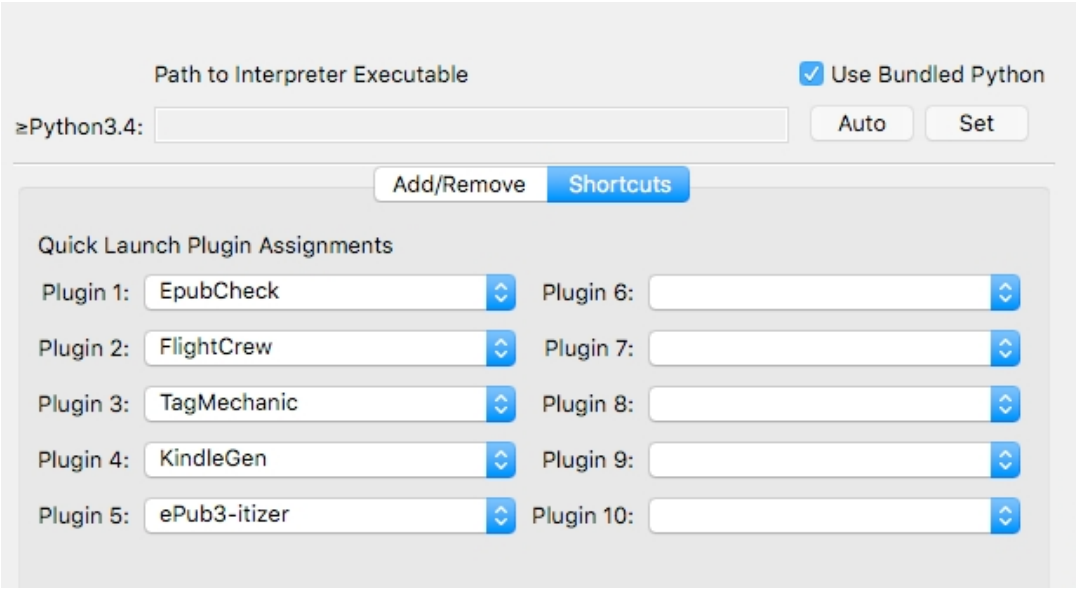
To remove a plugin:

1. Go to **Plugins=>Manage Plugins=>Add/Remove**.
2. Select the plugin you want to remove and click **Remove Plugin**.
   You can also click **Remove All** to remove all plugins in one go.

Assign Plugins to Toolbar Buttons

For quick and easy access to your plugins, go to **Plugins=>Manage Plugins=>Shortcuts** and use the dropdown menus to assign up to ten of your most frequently used plugins to ten toolbar buttons, then click **OK**. If a plugin has a custom icon, it will appear; otherwise a generic icon will be used.

Toolbar buttons:



To show or hide plugin toolbars, go to menu item
View=>Toolbars=>Plugins Set 1 (for plugins 1-5) and
View=>Toolbars=>Plugins Set 2 (for plugins 6-10).

> If a plugin doesn't have a custom icon or you prefer a different icon, you can change it. First you'll need to create a SVG icon or a 48x48 pixel PNG icon. Most importantly, the icon must be named "plugin.svg" or "plugin.png", with priority being given to the SVG version if both exist.
>
> Then go to **Plugins=>Manage Plugins** and click **Open Preferences Location** in the lower left corner. Finally, open the "plugins_prefs" folder and drop your new icon into the folder for that plugin. If there is no preferences folder for the plugin, create one using the exact same name of the plugin in the "plugins" folder.

Running Plugins

Once a plugin is installed, running it is as easy as selecting it from the **Plugins** menu or clicking its toolbar button.

For example, to run the EpubCheck plugin, select

Plugins=>Validation=>EpubCheck or click .

> If you don't recognize a toolbar button, hover over it to see the plugin's name.

Troubleshooting

```
Error: Interpreter python3.4 has no path set
```

This error message is usually displayed if Use Bundled Python is disabled and the ≥Python 3.4 text box is empty or contains an invalid path. Click **Auto** or **Set** to select an external Python interpreter or enable Use Bundled Python.

```
Error: Plugin is not a valid Sigil plugin
```

This error message is usually displayed if the plugin .zip file is invalid, was unzipped, or was renamed. If you renamed the plugin .zip file, re-download it and retry the installation.

If you're still experiencing problems, install and run the Sigil test plugin and post the results in the MobileRead Sigil Plugins forum.

Write Your Own Plugins

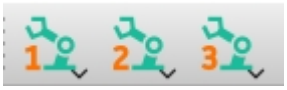
Thanks to a comprehensive Python framework, it is relatively easy to write plugins for Sigil. If you're interested in writing your own Python 3 Sigil plugins, have a look at the Sigil Plugin Framework guide.

Automate Lists

— Perform a List of Routine Tasks with a Single Click —

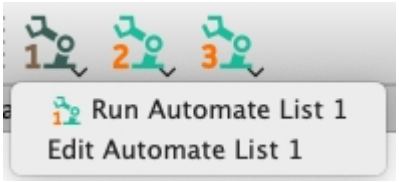
Many EPUB developers find themselves repeating the same set of tasks every time they start a new EPUB, or to clean up HTML imported from various sources, or in preparation for saving an EPUB to submit to a publisher. These tasks typically involve running some combination of internal Sigil commands/tools (e.g., Mend and Prettify HTML, Update Manifest Properties, etc.) and external plugins (e.g., EpubCheck, FolderIn, ePub3-itizer, Access-Aide, etc.).

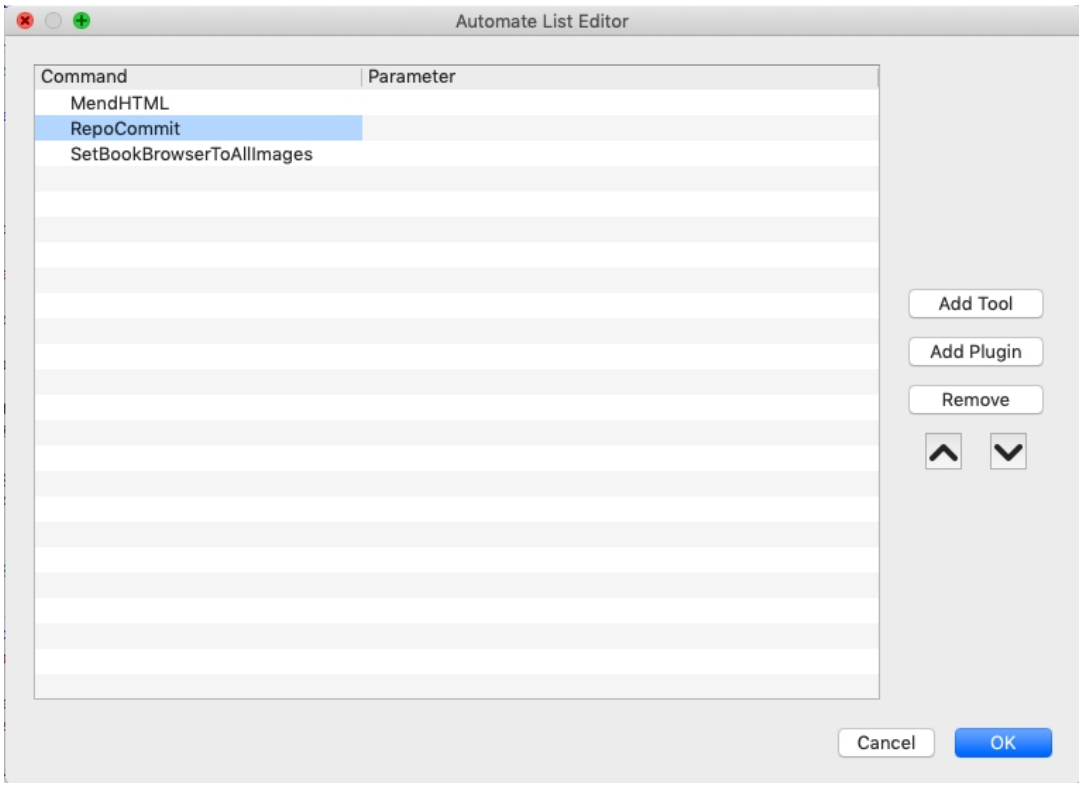
Starting with Sigil 1.8.0, a new feature called Automate Lists makes it possible to create a list of routine tasks, assign it to one of three **Run Automate List** toolbar buttons, then run the entire list of tasks in sequence with a single click of the button.



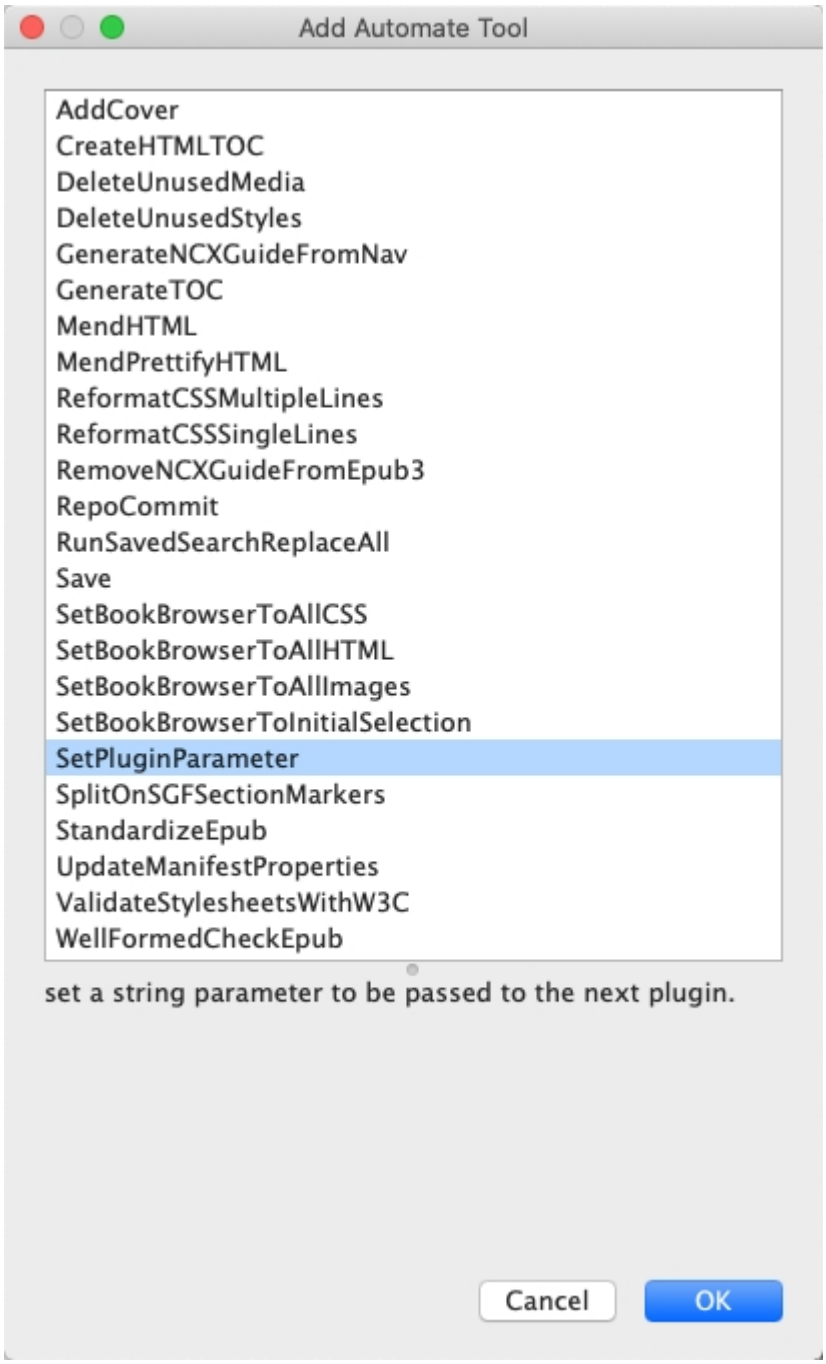
Creating and Editing Automate Lists

To create and edit an Automate List, click and hold one of the three Automate List toolbar buttons and select Edit Automate List from the dropdown menu. This will bring up the Automate List Editor.



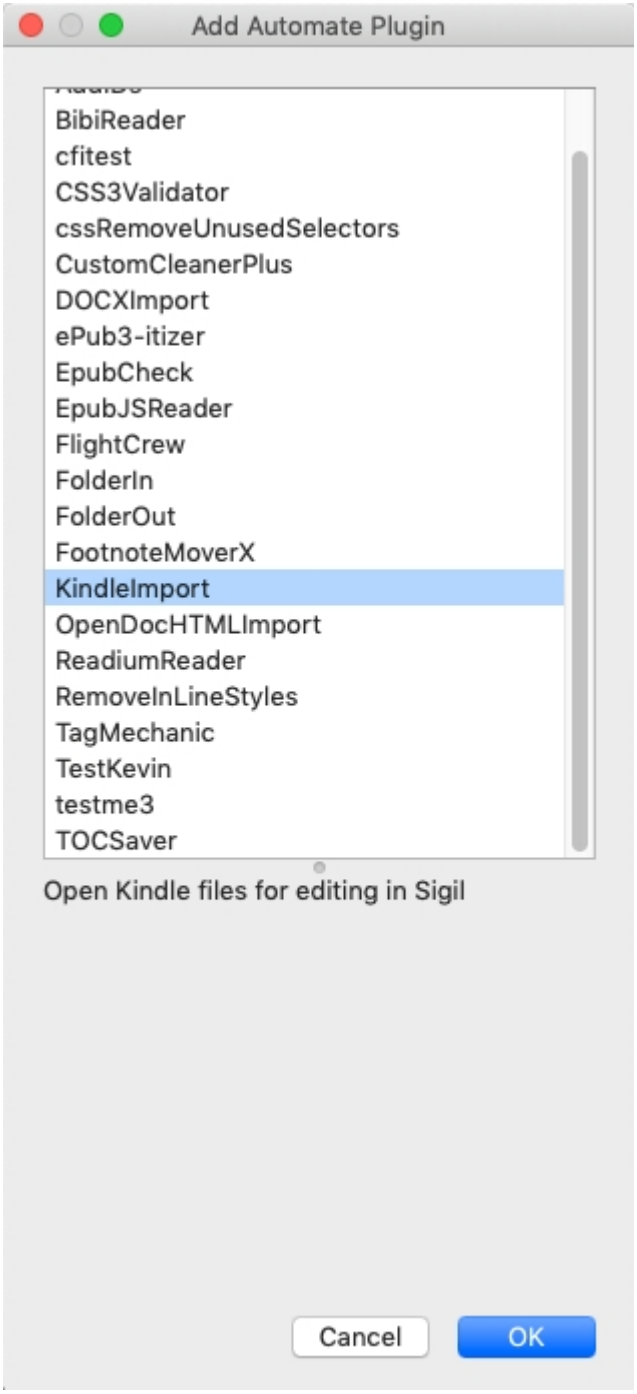


To add a Sigil tool to your Automate List, click the **Add Tool** button. This will bring up a menu of Sigil tools to choose from. Select a tool to see a description of what it does. Click **Okay** to add it to your Automate List. (Since many Sigil tools require user interaction to be effective, not all Sigil tools are currently supported.)



To run a Saved Search with Replace All, you must input the name of the Saved Search in the Parameter column. You can also run a Saved Search Group, by inputting the name of the group followed by a "/". For example, let's say you have a search group called "Automate Test" with two searches in the group named test1 and test2. If you input "Automate Test/", all searches in the group will run in sequence. If you input "Automate Test/test2", only test2 will run.

To add a plugin to your Automate List, click the **Add Plugin** button. This will bring up a menu of all of the plugins you have installed. (For more information on installing plugins, see Manage Plugins.) Select a plugin to see a description of what it does. Click **Okay** to add it to your Automate List.



Once you have created an Automate List, you can change the order of the tasks by selecting the command and clicking on the **up** or **down** buttons. To remove a command, select it and click **Remove**.

Since unexpected errors can always occur, it is a good idea to start your Automate List with a RepoCommit command to create a full Checkpoint of your existing EPUB. That way, if an Automate List fails, you can easily recover your entire EPUB using Sigil's Checkpoint Restore feature. (For more information on using checkpoints, see Using Checkpoints to Quickly Save/Restore EPUB State.

Automate Lists Are Simple Text Files

Automate Lists are simple text files located in the root of your Sigil Preferences directory. They are named: automate01.txt, automate02.txt and automate03.txt. As such, you can edit them using any text editor if you like. To jump directly to your Sigil Preferences directory, go to **Edit=>Preferences** (on Mac: **Sigil=>Preferences**) and click the **Open Preferences Location** button in the lower left corner.

Here's an example of an Automate List text file one might run as a final clean-up and validation check before publishing an EPUB:

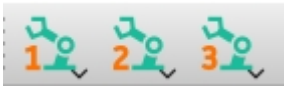
```
# start with Checkpoint save to prevent any data loss
RepoCommit

# run a basic well-formed test before running any advanced tools
WellFormedCheckEpub

# prettify any code and fix any remaining html errors
MendPrettifyHTML

# clean up
DeleteUnusedMedia
DeleteUnusedStyles

# make sure styles are valid
ValidateStylesheetsWithW3C

# make an epub3 that works with epub2-only readers
GenerateNCXGuideFromNav

# update manifest properties
UpdateManifestProperties

# validate epub and save
EpubCheck
Save
```

While there are only three Automate List toolbar buttons, you can create any number of lists and store them in your Preferences folder. To run a list, you will

need to assign it to one of the three toolbar buttons by renaming it automate01.txt, automate02.txt or automate03.txt.

Running an Automate List

To run an Automate List, simply click on the appropriate **Automate List** toolbar button.



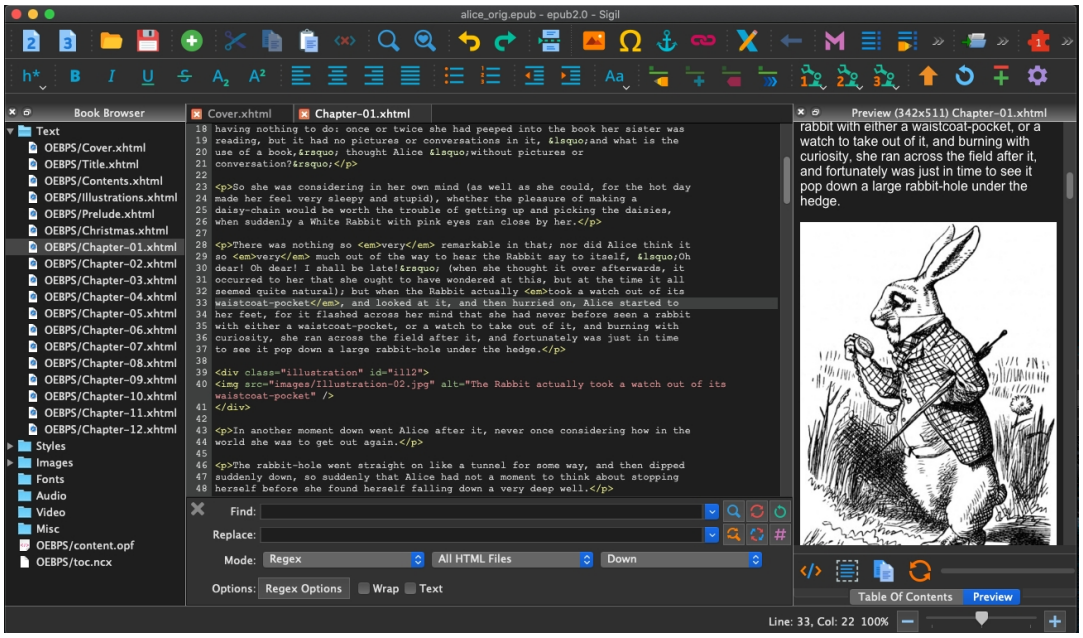
If any command fails to process, the Automate List is immediately terminated. Watch Sigil's status bar for messages as the list is executed. Once the list is completed, a dialog showing a complete log of your Automate List output is displayed so you can see what succeeded, what failed, and most importantly, why.

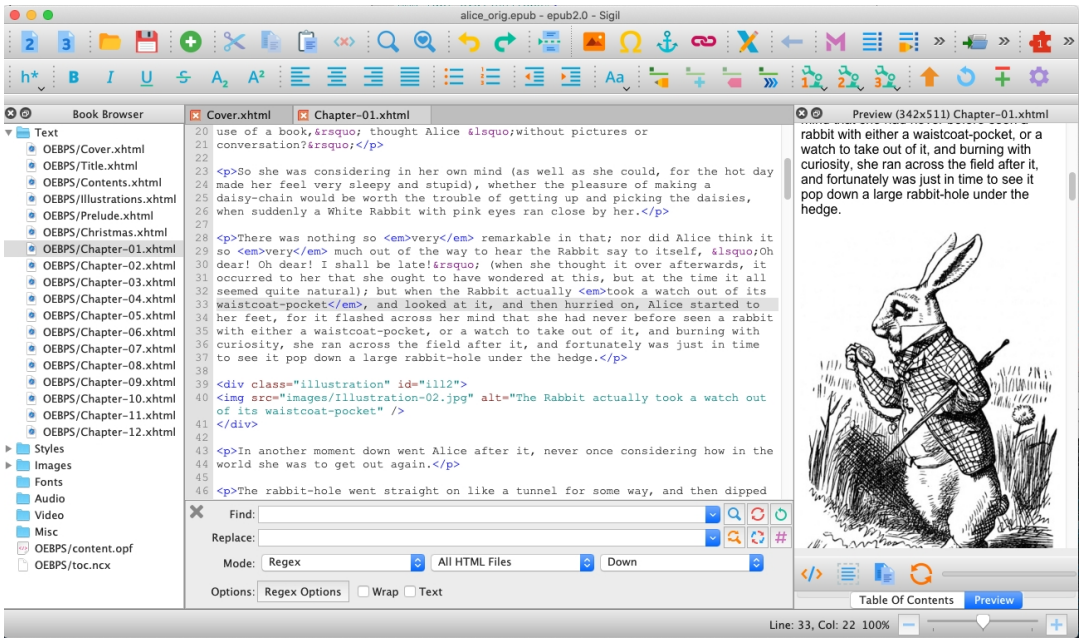
Customize the User Interface

— Make it Work the Way You Want —

The Sigil user interface is highly configurable.

Sigil supports both Dark and Light themes inherited directly from your System preferences settings.





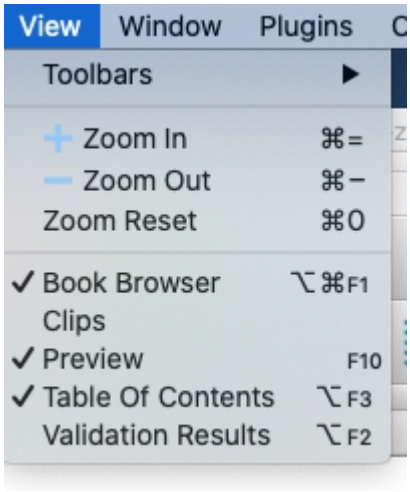
On macOS, your System Preferences can be used to change from Light to Dark themes at any time or even automatically based on time of day. On Windows and

Linux, your System theme is detected and set at Sigil's launch. You can override this setting using an environment variable and then restart Sigil to see the changes.

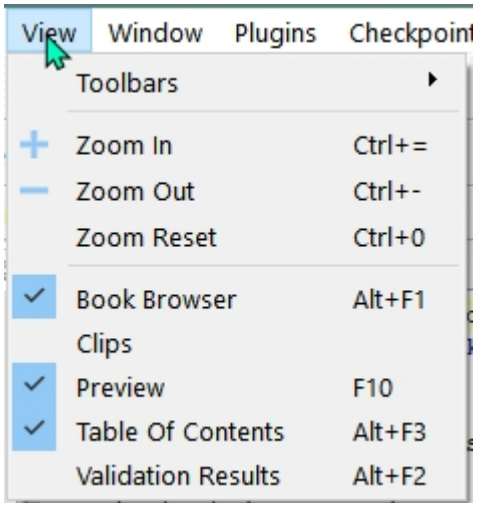
- set **SIGIL_USES_DARK_MODE** to **1** to force Sigil to use a Dark theme when your system uses a Light theme.

- set **SIGIL_USES_DARK_MODE** to **0** to force Sigil to use a Light theme when your system uses a Dark theme.

Sigil's primary interface consists of a main window in which are displayed several panels plus a set of toolbars. Which of these items you want to display are configurable using the View menu.



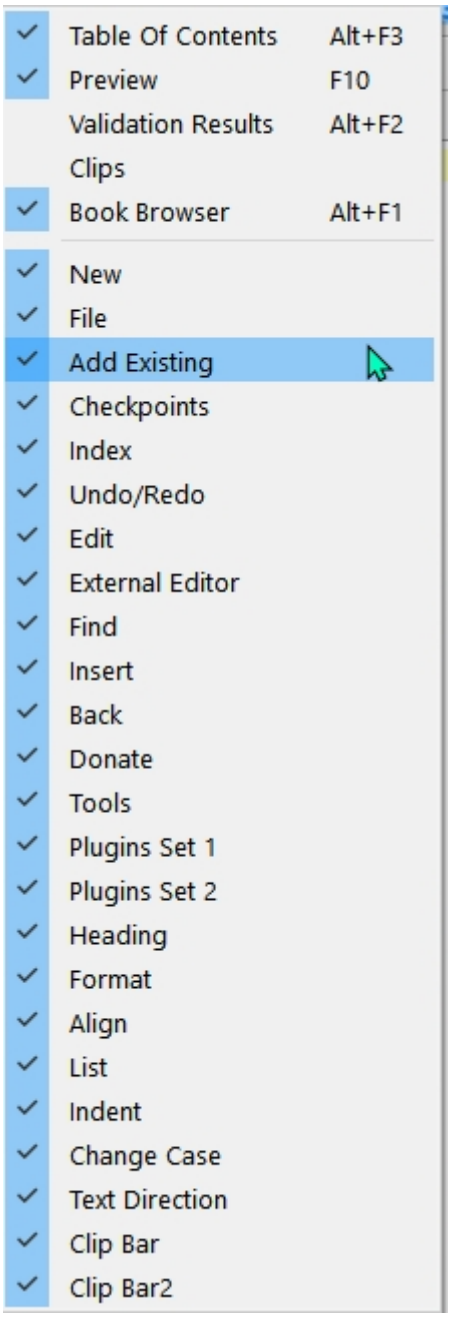
The panels you want to display can be customized in the main View menu.



The panels you can use are:

- **Book Browser:** For browsing the files in your book.
- **Clips:** Your defined clips
- **Preview:** Dynamically updated preview of your book.
- **Table of Contents:** The Table of Contents.
- **Validation Results:** The output from validating your book.

The Toolbars can be customized using the Toolbars menu, which is posted by hovering the mouse over the “Toolbars” item at the top of the View menu.



The rest of this chapter discusses three topics

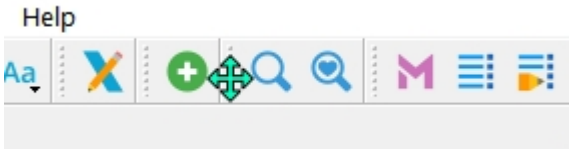
- Customizing the Toolbars

- Customizing the Panels
- Customizing the Icons

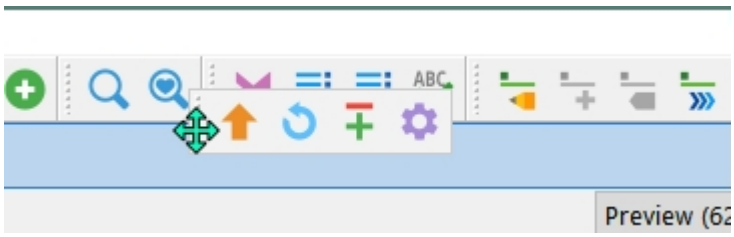
Customizing the Toolbars

Open the **View=>Toolbars** menu and select the toolbars that you want to see in the UI. Then, move them to the locations that best suit you.

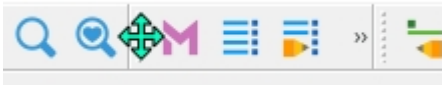
To move a toolbar to a different location, hover the cursor over the grab bar on the left side of the toolbar. When you do that, a move cursor will appear.



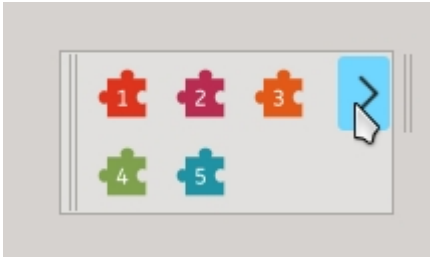
Drag the toolbar to a different location on the main window. An area will highlight where the toolbar will be placed when you release the mouse.



You can also resize the toolbar by dragging the move mouse cursor to the left and right.

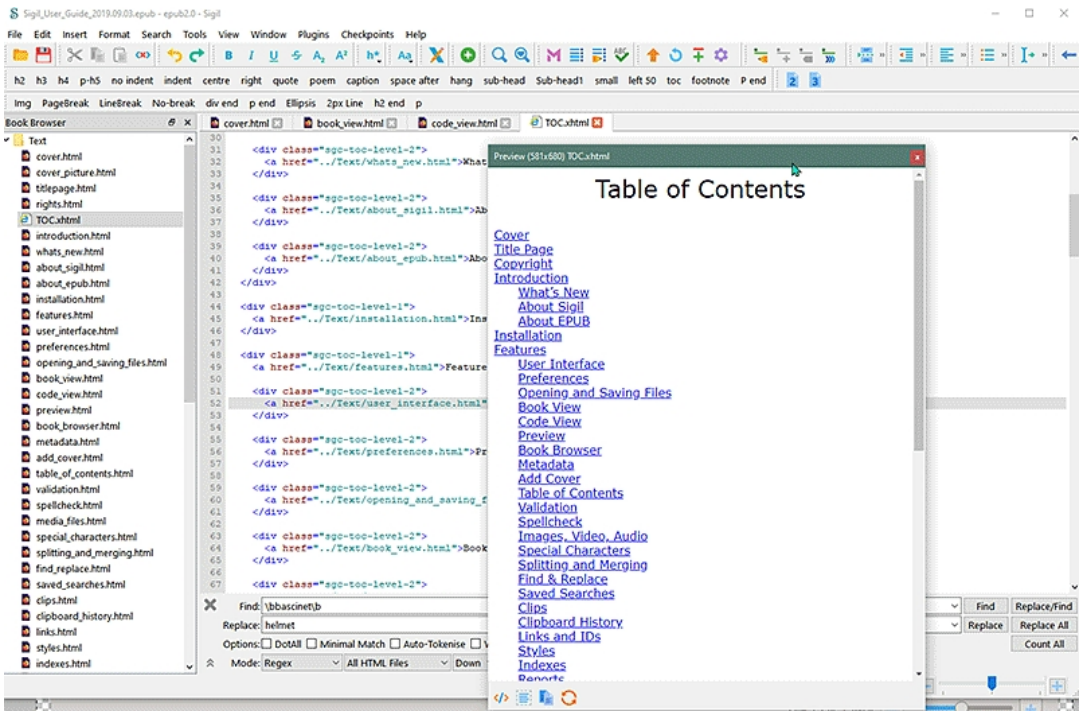


If you make the toolbar smaller than a size that will display all of the icons, an arrow will appear at the right side of the toolbar. When you click on that arrow, a submenu will appear allowing access to all the buttons in the toolbar.

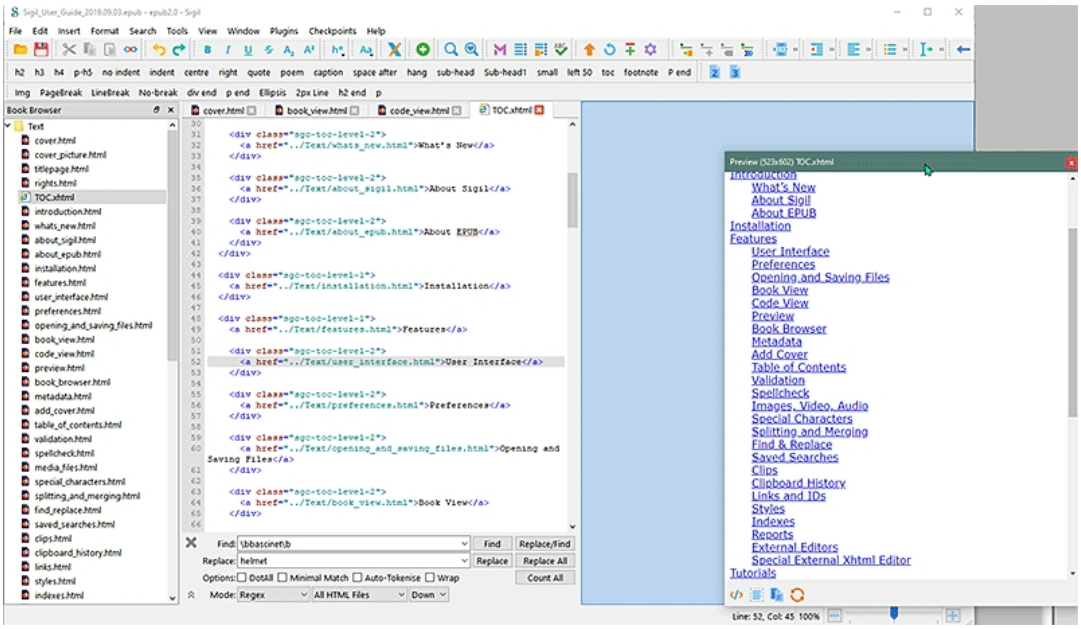


Customizing the Panels

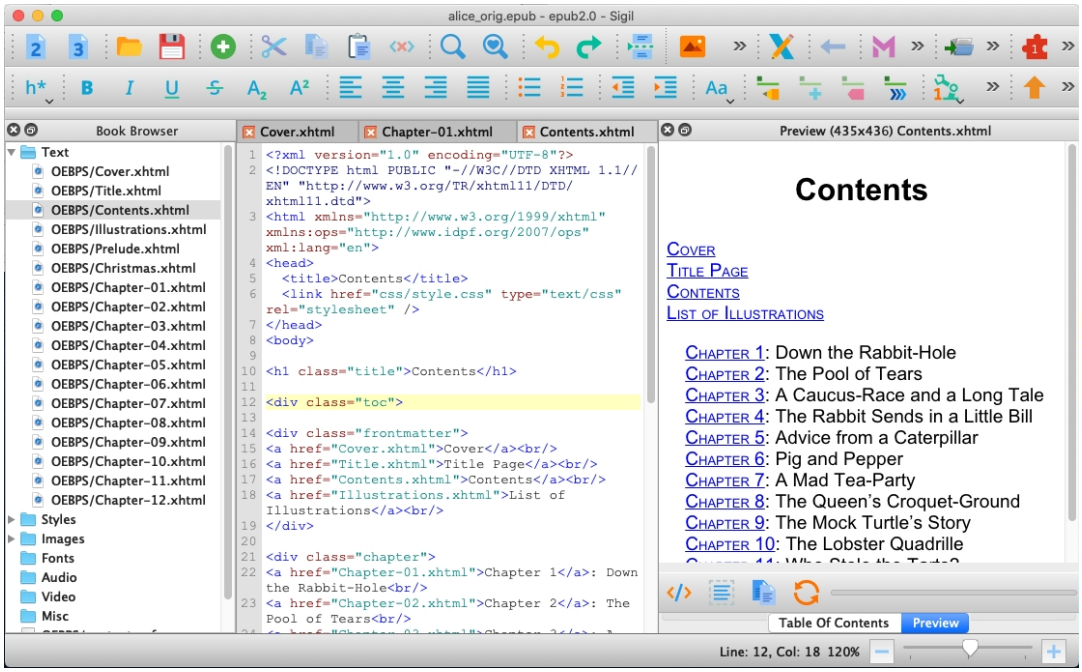
The panels are customized in a similar way that you customize the toolbars. If you move the mouse cursor to the top banner of a panel and press the left mouse button, the move cursor appears. You can then drag the panel out of the main window to undock it from the main window. The panel becomes a free floating window.



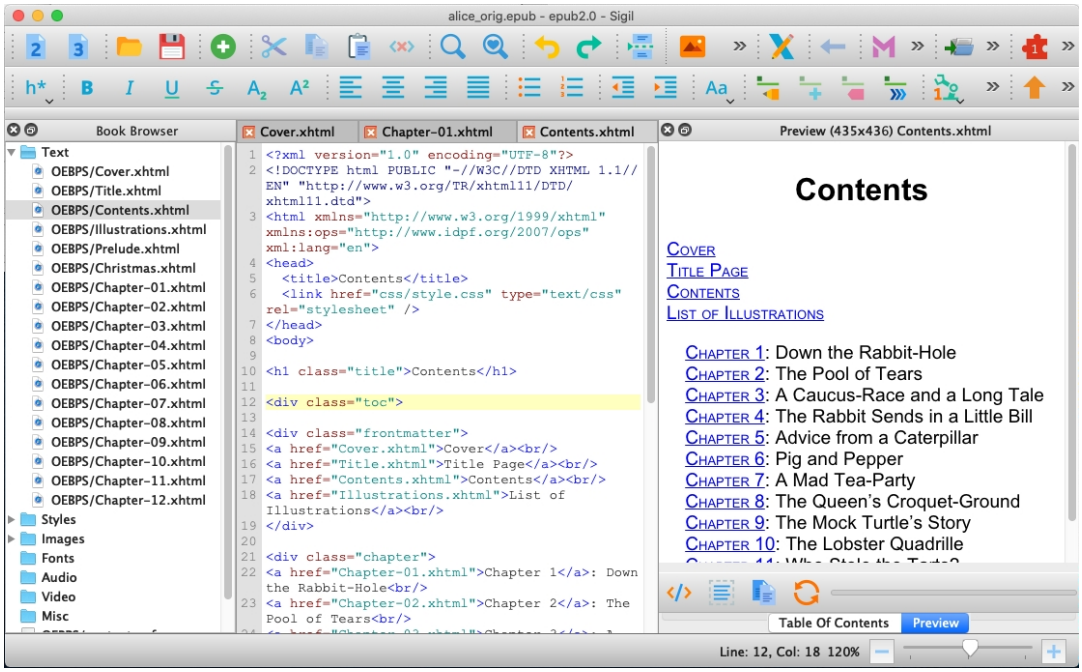
To re-dock the window back into the main window, drag it until the area where you want to place it is highlighted and release the mouse button.



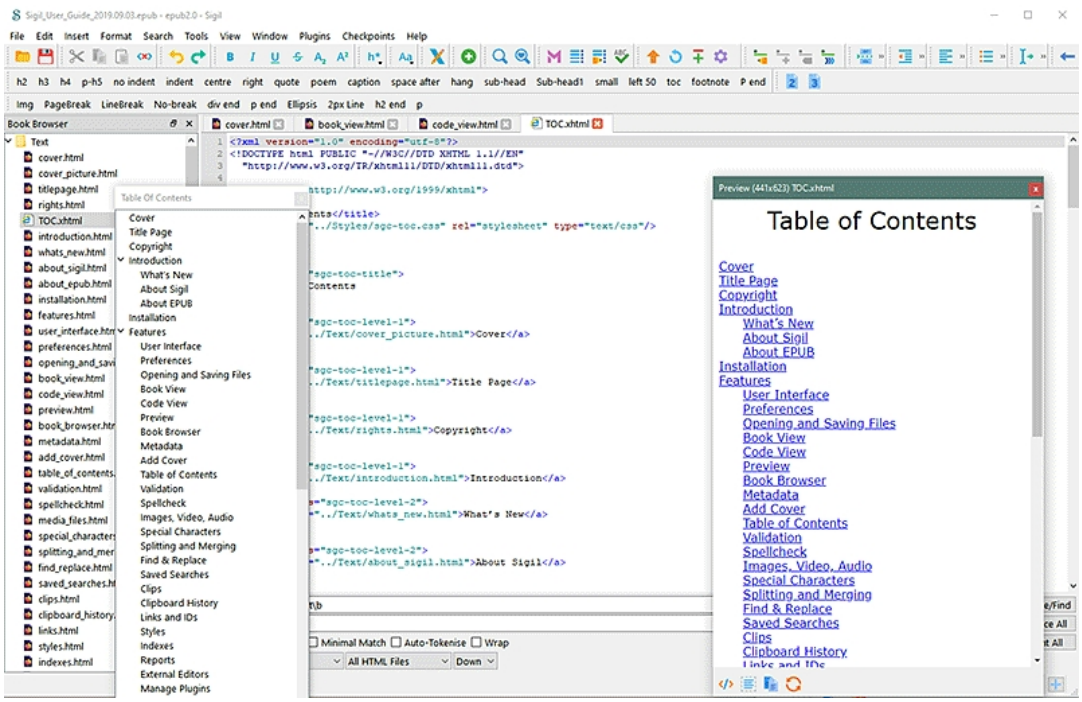
You can drop the floating window onto an existing panel to create a tabbed interface for the two panels. You do this by dragging it over the existing panel so that the existing panel is highlighted.



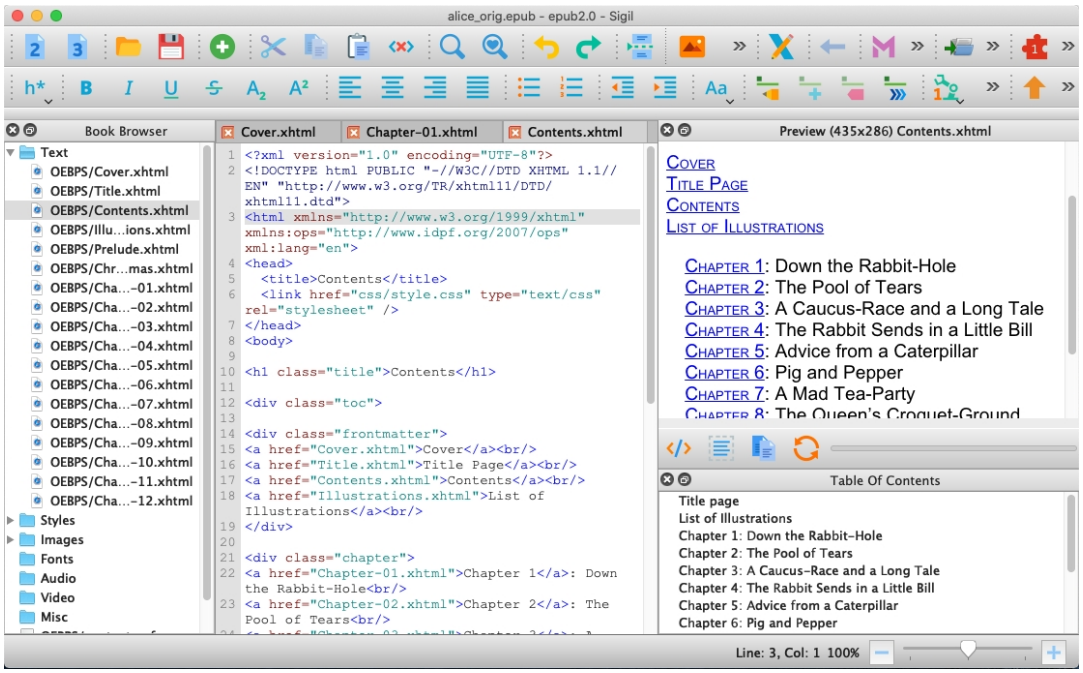
When you drop the window, a tab will be created so that you can switch between the two panels.



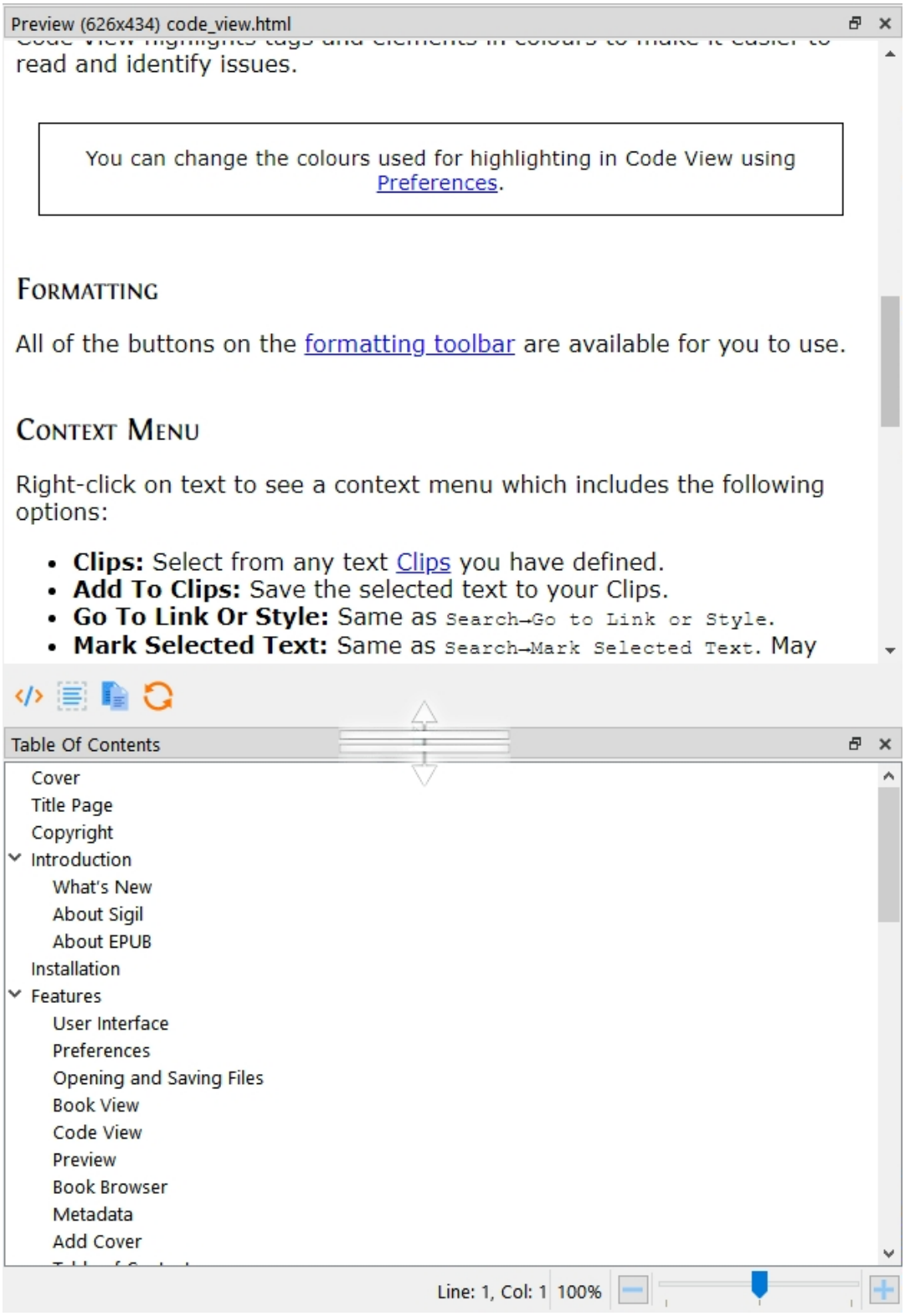
To dock the window back in as a separate panel next to an existing panel, drag it over the main window until an area is highlighted to the side, top or bottom of the existing panel, then release the mouse button.



The window will be redocked into the main window as a separate panel.



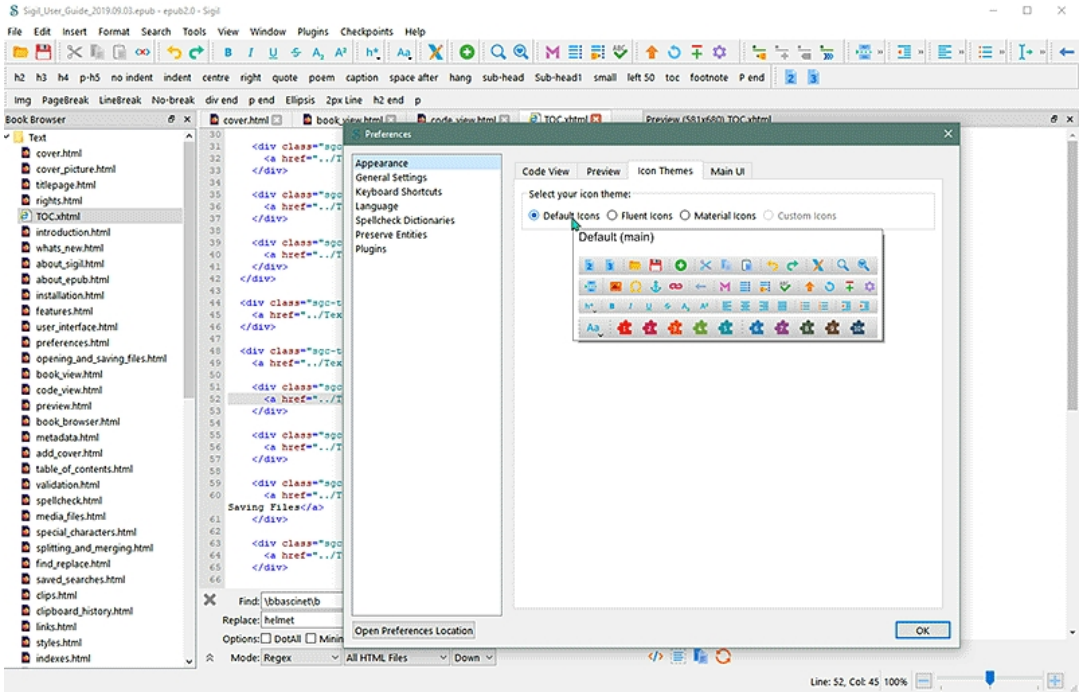
The relative sizes of the panels can be adjusted by hovering the cursor over the boundary between the two panels and dragging the boundary to a new location.



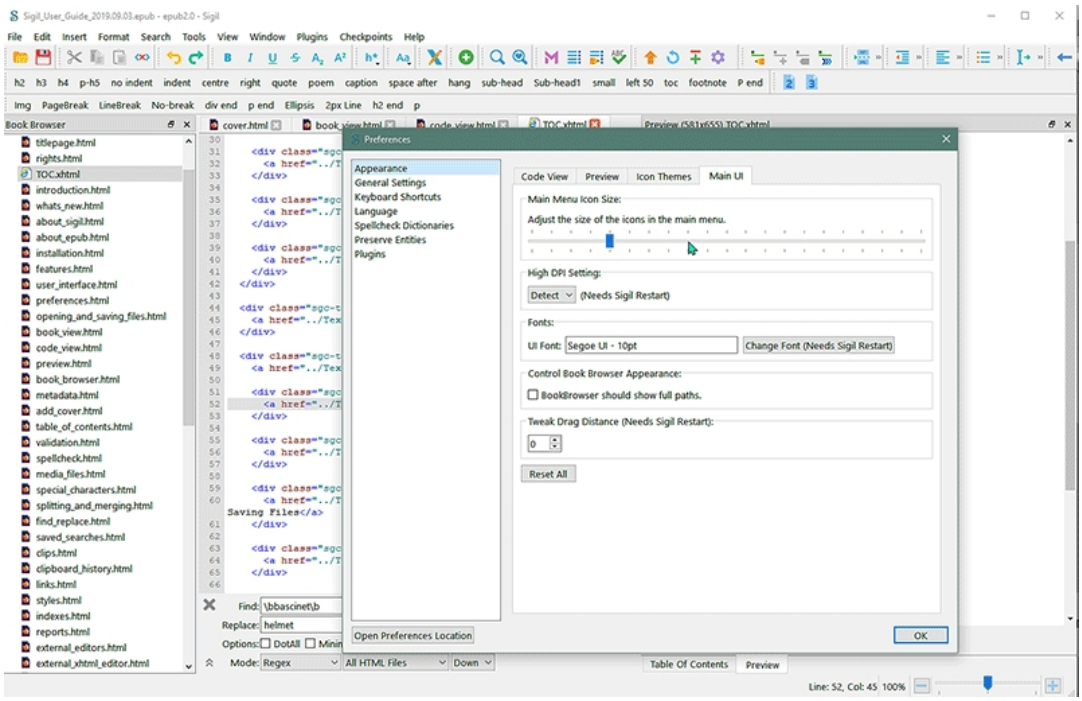
Customizing the Icons

The icons can be customized in the Preferences dialog.

To customize the icons theme, open the Preferences dialog using the **Edit=>Preferences…** menu. Click on the Appearance tab on the left and on the Icon Themes tab. If you hover the cursor over each of the selectable themes, a tooltip will display thumbnails of the icons in that theme.



Adjust the size of the icons by clicking on the Main UI tab and adjusting the slider.



TUTORIALS

— HOW-TO GUIDES —

Create your first EPUB ebook and learn about using Sigil with the tutorials in this section.

Details on all of Sigil's features can be found in the Features chapter.

To give you an idea of what is involved in making an EPUB, here is an overview of the steps you might follow:

- Convert your document to HTML (if it's not an EPUB already) so it can be loaded into Sigil.
- Enter book details including Author and Title.
- Add a cover image.
- Create a Table of Contents.
- Validate your ebook against the EPUB standards.
- Test everything looks and works ok in Sigil.
- Test on actual e-readers – what works in one may not work on another.

To see what an EPUB looks like and to try out Sigil, you can download the EPUB version of the Sigil User Guide and open it in Sigil.

Getting Started With EPUB

If you are going to create an EPUB ebook, it's probably a good idea to understand what EPUB is and how Sigil can help.

EPUB is a standard file format created to display books, and Sigil helps you create EPUBs. Sigil is designed to take the book you created in a word processor and convert it to a standard EPUB file that can be published and read on e-readers. It can also be used to clean up and reformat existing EPUBs.

How much you want or need to "clean up" your EPUB depends on you. If you are just formatting a book for yourself you may not be concerned with too many details and will be able to use Sigil just enough to get it to look right on your e-reader. But if you are formatting for publication then you'll find Sigil makes it easy to access and change any detail about your book.

When you edit a document in a typical word processor you usually end up with 1 file, e.g. mybook.doc. This file contains all the information about your document. An EPUB is slightly different. Instead of using just 1 file to store your information, it uses separate files to store the parts that make up your document: chapters, images, table of contents, etc. But because it would be difficult to share lots of separate files with people, all these separate files are actually stored into a single compressed zip file with the file extension .epub, e.g. mybook.epub. An EPUB file is actually just a zip file that contains all of the files needed by your book.

Sigil can open and create EPUB files, and it allows you to edit, rearrange, and format the individual files in the EPUB (like your text and table of contents).

Once you copy or import your document into Sigil, it becomes part of the EPUB file. And when you save your book in Sigil, your document is saved in the EPUB file.

Note that Sigil is designed to work equally well with the old EPUB 2 standard or the newer EPUB 3 standard. Many older e-readers only support EPUB 2, although by adding a Guide and NCX to the newer EPUB 3 version and restricting it to EPUB 2 features, a fully backwards compatible EPUB 3 can be generated that should work well on all e-readers.

EPUB 3 EPUBs are quickly becoming the international publishing standard as they support both Right-To_Left (RTL) and Left-To_Right (LTR) text layout, audio, video, internal and external resources, and better support Accessibility features by allowing multi-media overlays, improved semantic role labelling, and even limited javascript for interactive features.

Prepare Your File For Sigil

If you already have an EPUB or HTML file, or are writing from scratch in Sigil, you can skip this step.

If you are starting with a document created by a word processor, then the first thing you need to do is convert your document to HTML so that you can import it into Sigil. Some programs can generate EPUB files as output. And some programs have add-ons or features to clean up their HTML output. Whether it is best to use the HTML or EPUB (or even plain text) output of these programs depends on the program – you can always test different ways to see what works best.

> The "cleaner" your HTML file is, the easier it will be to turn it into an EPUB and the more likely it will display correctly on multiple devices. This means using styles for formatting and not using too many fancy formats and layouts.
>
> Do not expect an EPUB to look like a printed book – EPUB is designed so that the reader can control how the book is displayed.

Although conversion is not part of Sigil, here are some suggestions for converting files for importing into Sigil.

Word Documents

Use the MS Word menu item **File=>Save As Filtered HTML**. Or use the MS Word Macro @ MobileRead.

Both approaches try to reduce the amount of extra code exported by MS Word so that your HTML file is cleaner and simpler to edit in Sigil. You really want to have your code as clean as possible since Word adds lots of extra HTML tags that you will probably want to delete at some point.

When using Word or any word processor, you should always try to use styles to format your code instead of just making some text bold, some larger, etc. Styles are also an important concept when formatting EPUBs.

There are numerous sites on the Internet that deal with converting MS Word files for use in EPUBs. Here are just a couple:

- https://www.epubble.com/word-to-epub-tutorial
- https://justpublishingadvice.com/how-to-convert-word-to-epub/

There are also free apps available from both Microsoft and Apple that automate the process for you.

Windows: Word-to-epub-converter

Apple iOS: The Document Converter

LibreOffice ODF/ODT

The LibreOffice writing application has the same capabilities as MS Word when it comes to saving an ODT file to html – **File=>Save A Copy** (setting the filter to HTML Document (Writer) (.html)) OR **File=>Export** (setting the filter to XHTML (.html;.xhtml))

For EPUB use it is preferable to use Save a Copy as this creates a cleaner output than Export with less formatting data in the document. Save a Copy also links to any images, whereas Export embeds the image data in the file.

Whilst not specifically EPUB related some information on this subject can be found in the LibreOffice on-line documentation – Saving Documents as HTML

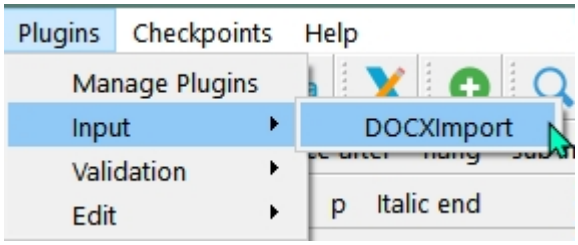
DOCXImport Plugin

To create a new EPUB book directly from a MS Word file you can use the DOCXImport plugin developed by DiapDealer. By adding this plugin you can directly import a docx into Sigil as html.

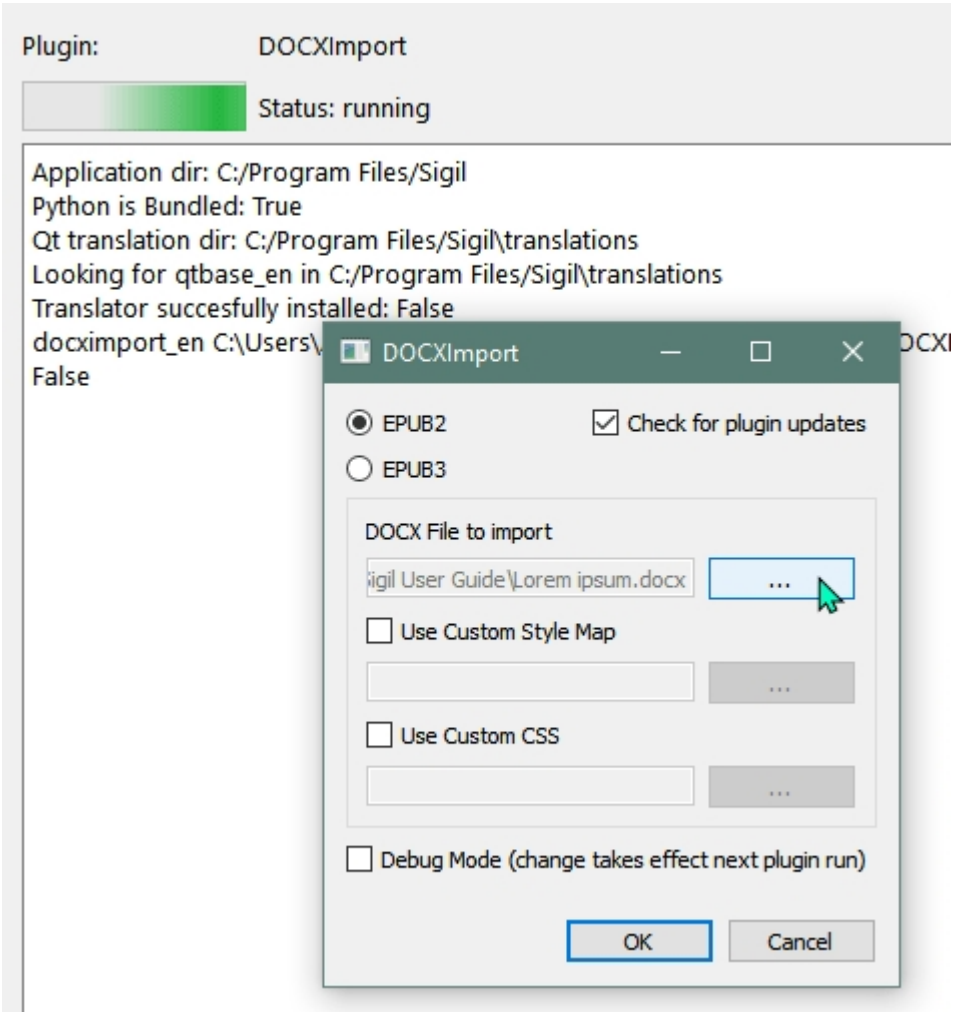
DOCXImport plugin (See the Manage Plugins chapter on how to install a plugin).

> Note: Using this plugin will create a new EPUB. If you already have an existing EPUB open in Sigil this will be replaced and any unsaved work will be discarded.

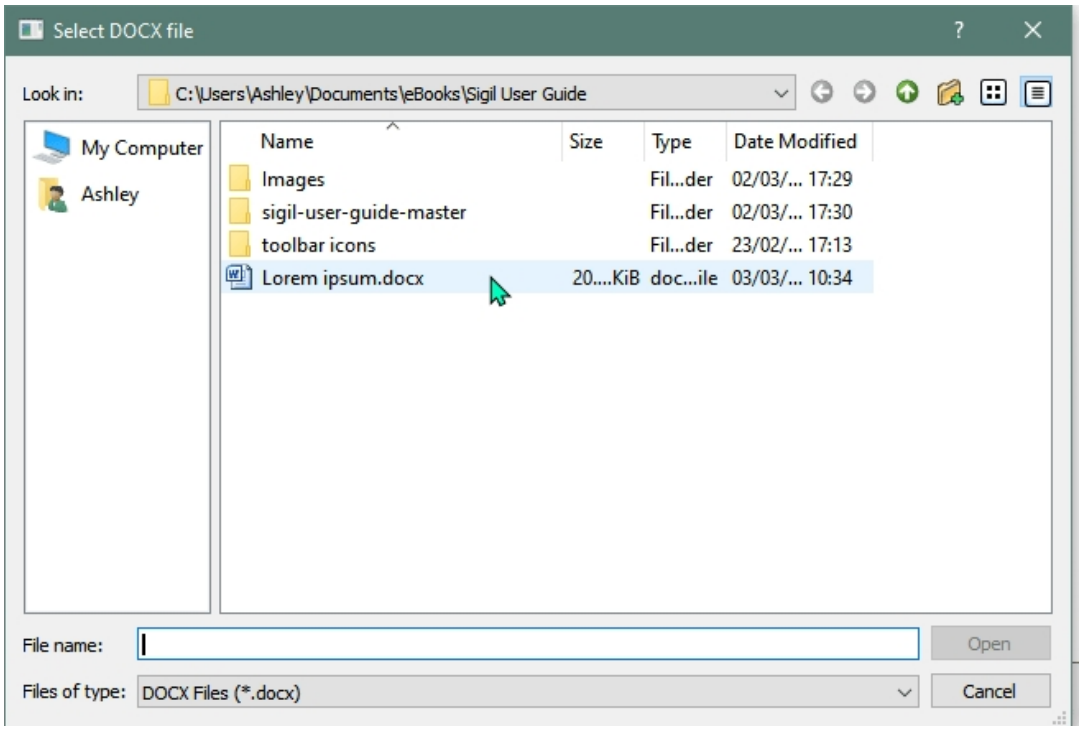
Click on **Plugins=>Input=>DOCXImport** to run the plugin.



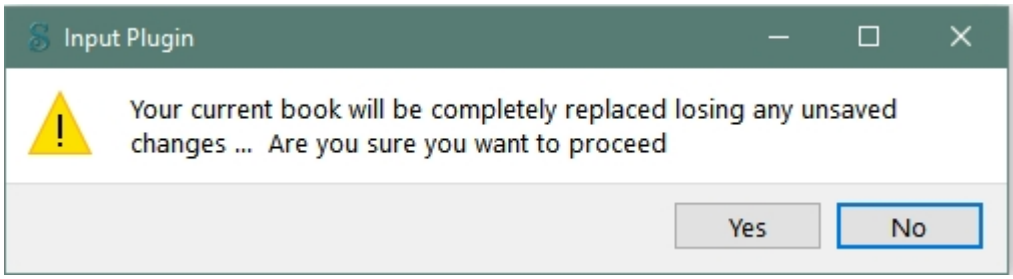
Prior to selecting the file to import you have the option of choosing which version of EPUB your book will be, and also whether to use a custom Style Map or Stylesheet.



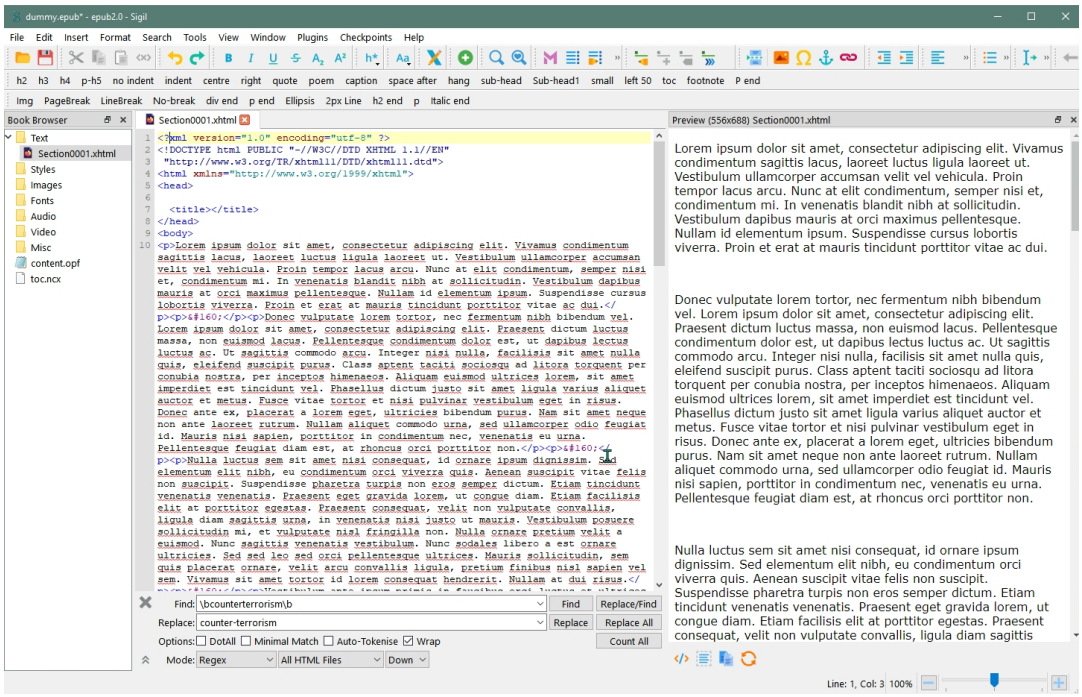
Click on DOCX File to import and then select the file you wish to import.



Heed the warning!



The docx file after successful importation and conversion ready for editing.



Text Files

It is not strictly necessary to convert text files to HTML. Sigil can directly open .txt files.

> Text files must have a blank line between paragraphs so that Sigil can split them into separate paragraphs.

Text files are not an ideal source, but they can still be better than some formats as they will have fewer unnecessary HTML tags. They may be worth using if you have lots of issues with other formats. Sometimes it is faster to add styles to a clean document than to try to clean up a converted file.

Other Formats

Many other word processors, such as LibreOffice, also have a save-as or export to HTML or even save-as EPUB function that you can try. As with MS Word good advice on this process can be found on the Internet.

You can also convert other formats to HTML or EPUB using additional software such as Calibre which handles most ebook formats, runs on all platforms, and has an ebook library built-in, as well as, Pandoc, a dedicated document conversion application that supports a huge number of document formats including EPUB2,

EPUB3, docx, odt and most Markdown variants. Pandoc is free, opensource, cross-platform, and written in Haskell (GHC).

Note that any automatic conversion of your files will introduce lots of extra code. If you are just formatting a book for personal use you may not be concerned with this. But if you are formatting books for professional purposes or want as little extra code in your EPUB as possible, then it should be avoided.

Aside from the EPUB format, Sigil primarily supports importing HTML (with any linked in images, css, and media files), and Text files. Since converting from other formats can be quite specialised, it is often best left to plugins and external tools. Free Sigil import plugins do exist to directly read in MarkDown, Kindle, docx, and odt formats.

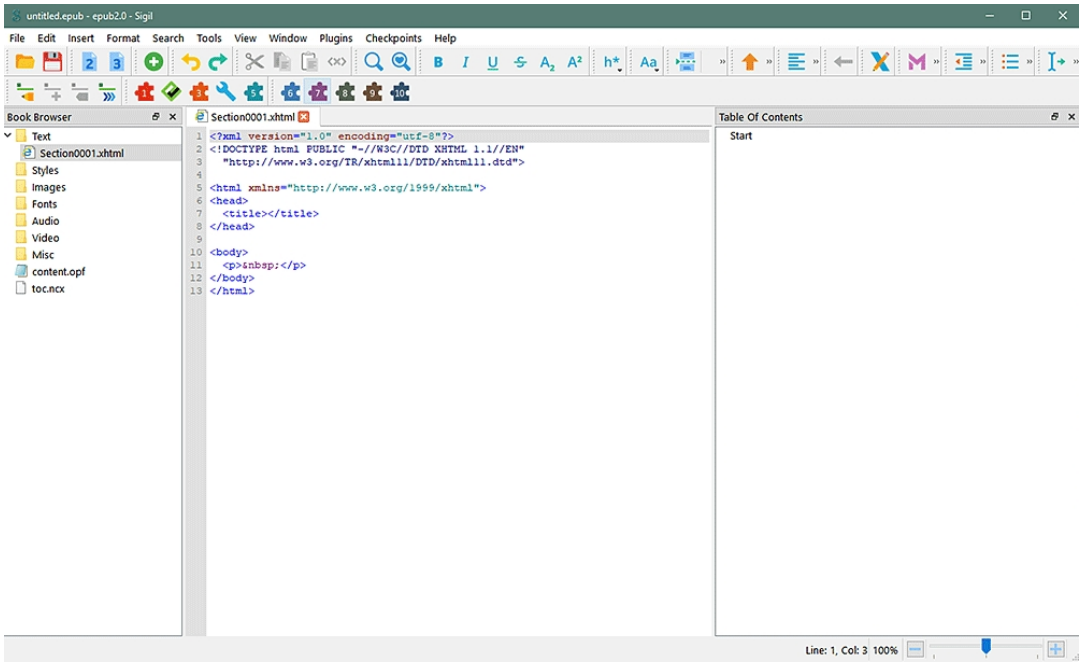
Open Your File With Sigil

Once you have an HTML or EPUB file, it's time to load it into Sigil. Even if you have an EPUB file already you should follow these steps for a quick introduction to using Sigil.

> If you are starting with an EPUB file, make a backup copy of the file before opening it in Sigil.

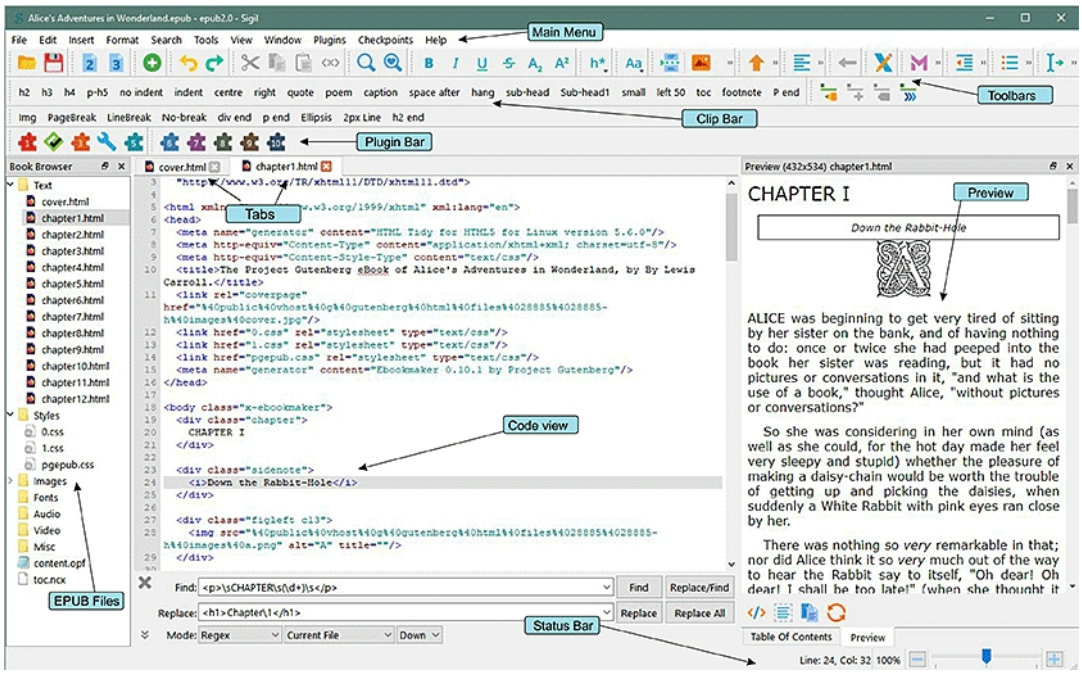
Start Sigil

When you first start Sigil you will be shown a screen with several windows:



The Book Browser window on the left shows you all of the files in your EPUB, the Code View editing Window is in the center.

This is how Sigil might look if an EPUB was loaded:

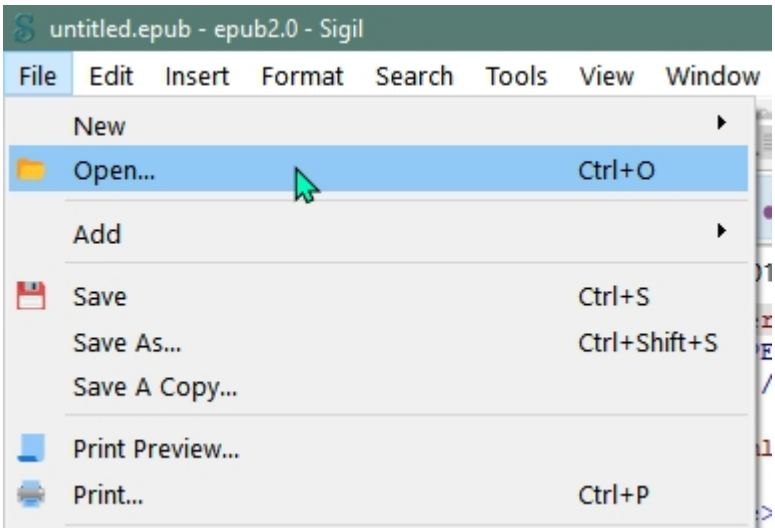


You can open individual files in your book for editing by **double-clicking** on their names in the Book Browser.

> See the User Interface chapter for more details.

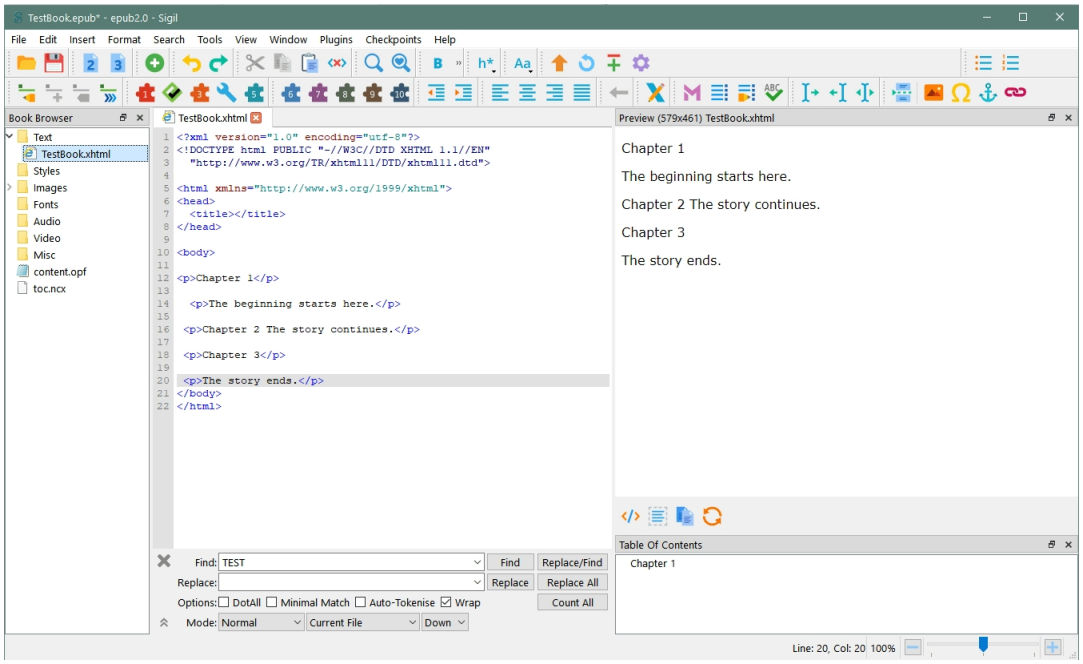
Load Your File

To load your HTML or EPUB file into Sigil, click the Open icon or use the menu item **File=>Open**: or the keyboard shortcut Ctrl+O



Select the file you want to open from your file manager, e.g. TestBook.html.

You will now see your HTML file(s) in the Book Browser window under the Text folder on the left. The file will be opened in the Code View editor in the middle, with its filename showing on the tab header:



Your HTML file is now part of an EPUB. From now on you are editing your EPUB and not just an HTML file. You may have noticed that when you opened your file in Sigil it created a couple of other files needed by the EPUB format, which you can see alongside your original document in the Book Browser.

If your file includes any images or stylesheets, etc., Sigil will attempt to load them as well and store them in the appropriate folder in the Book Browser.

When converting your file to EPUB or loading an EPUB Sigil may add some necessary details and organise your files into standard folder names.

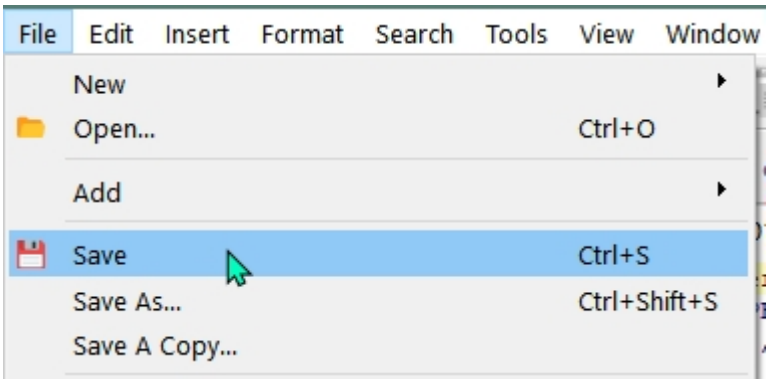
Save Your EPUB File

Now that you've loaded your file into Sigil, you should save it.

> You should always save right after loading a new file – even if it was an EPUB to start with – to make sure there are no problems with saving.

Since the file you are editing is now an EPUB (even if you loaded an HTML file to start with), when you save in Sigil you will be saving an EPUB formatted file (although it might be missing some details that you will need to add later).

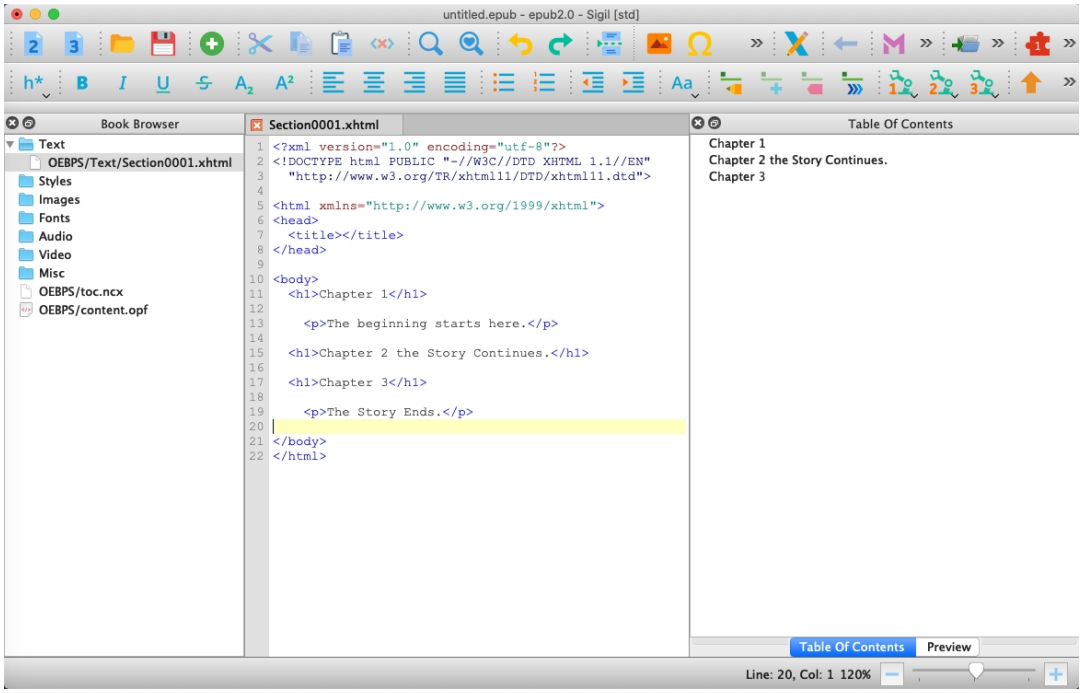
To save your EPUB, click the Save icon or use the menu item **File=>Save:** or the keyboard shortcut – Ctrl+S [1]



If you loaded an HTML file, your EPUB won't have a name yet – so you will be prompted to enter a name for your file. It will be given a .epub extension since it will be saved as an EPUB file.

> You can click the Save icon to save your EPUB at any time – and you should save often.

Notice that the title of your book – at the top of your window – is updated (e.g. to TestBook.epub) and that the status bar briefly shows "File saved":



You now have an EPUB file – but don't stop yet!

To make sure your ebook conforms to EPUB standards, you need to add a bit more information as shown in the following tutorials.

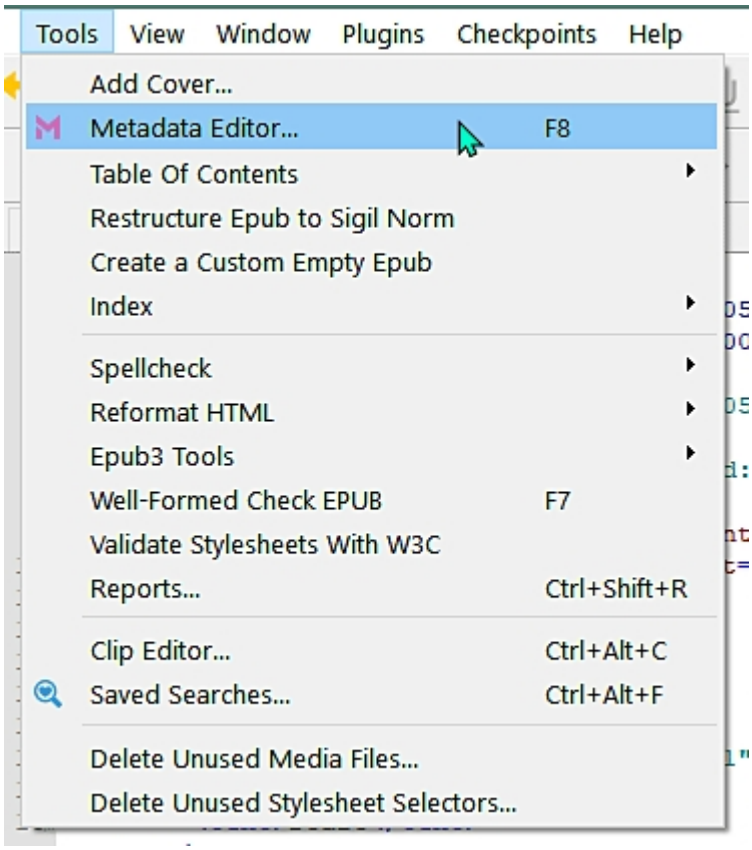
---

1 – If you wish to retain the original version of your ePub unmodified there are two further save options – **File=>Save As** which creates a copy with a different name, and **File=>Save A Copy** which creates a copy of the EPUB with _copy appended to the name.

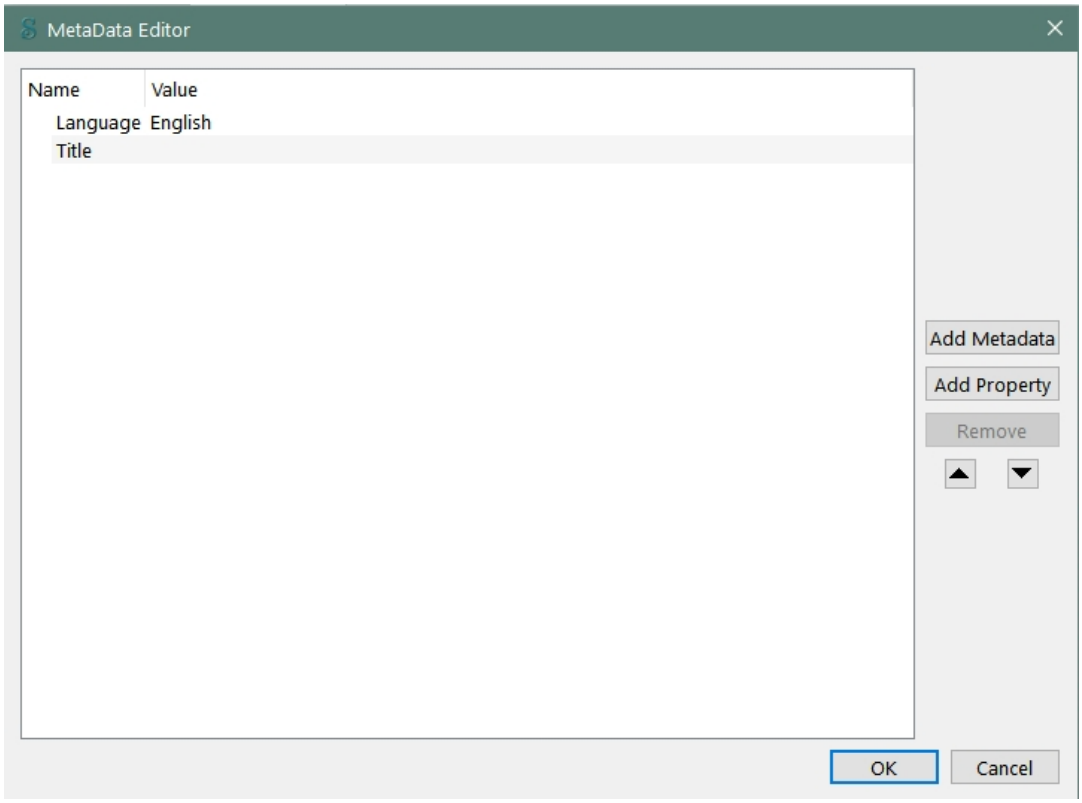
Add the Author and Title

In addition to the actual content of your book, EPUB files also contain book details like Author and Title. These can be added with Sigil.

To add details (called metadata) about your book to your EPUB, click the Metadata Editor button or the menu item **Tools=>Metadata Editor:**

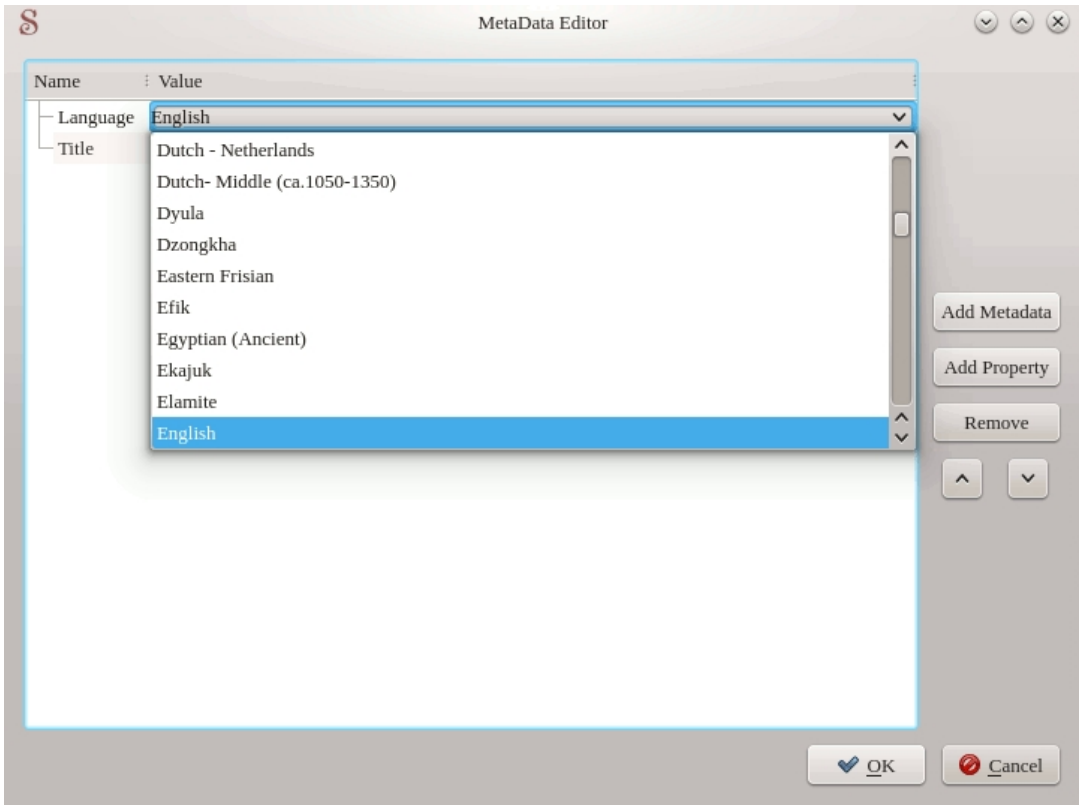


The Metadata Editor window will be displayed:

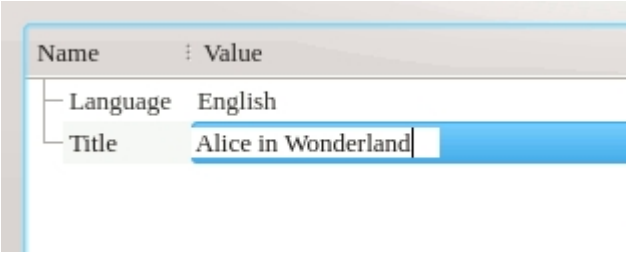


The entries required by an EPUB are Title, Author, and Language. By default, the metadata editor contains the Language field and the Title field as shown above.

To change the language, double-click on the Value item in the Language row. A dropdown menu appears in that space that allows you to select your desired language.

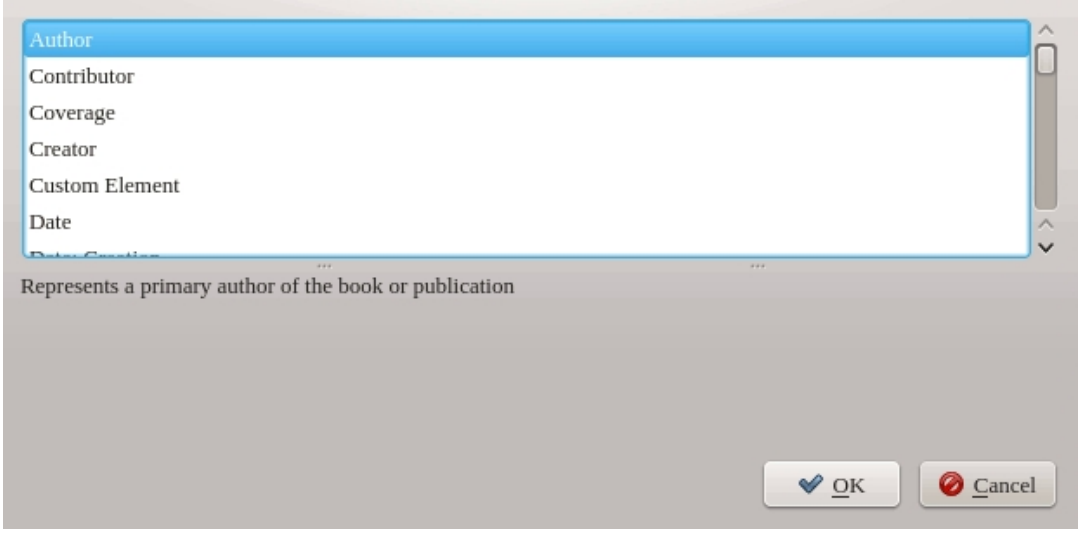


Fill in the Title field by double-clicking on the Value item in the Title row. That opens the field for editing.



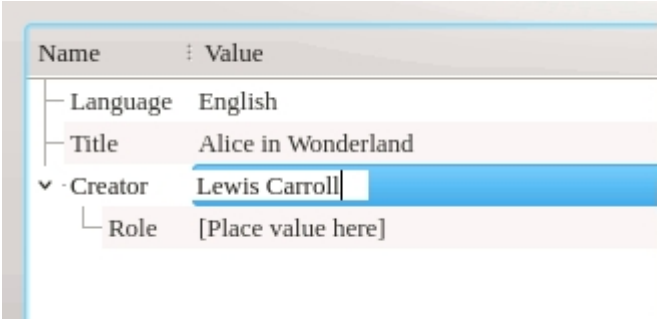
Type the full title of your book, for example enter Alice in Wonderland.

Add the Author to your book using the **Add Metadata** button. That button pops up a dialog that allows you to add a metadata item. Select Author from the list.

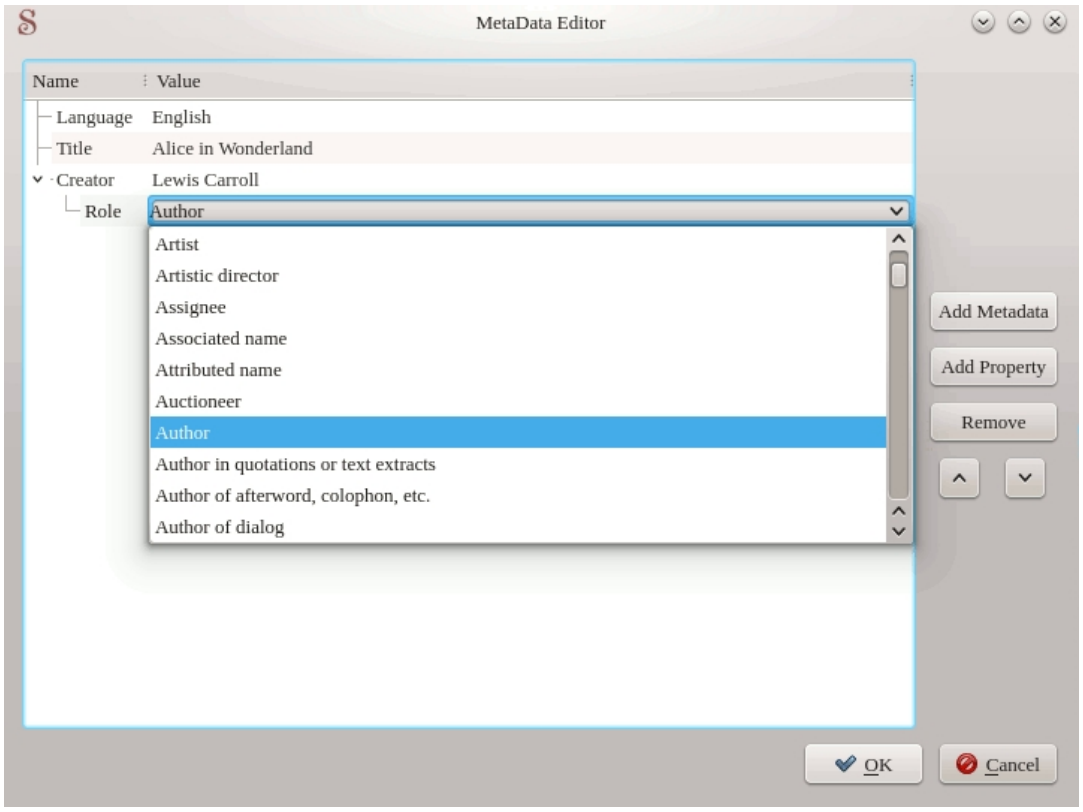


That adds the metadata item “Creator”.

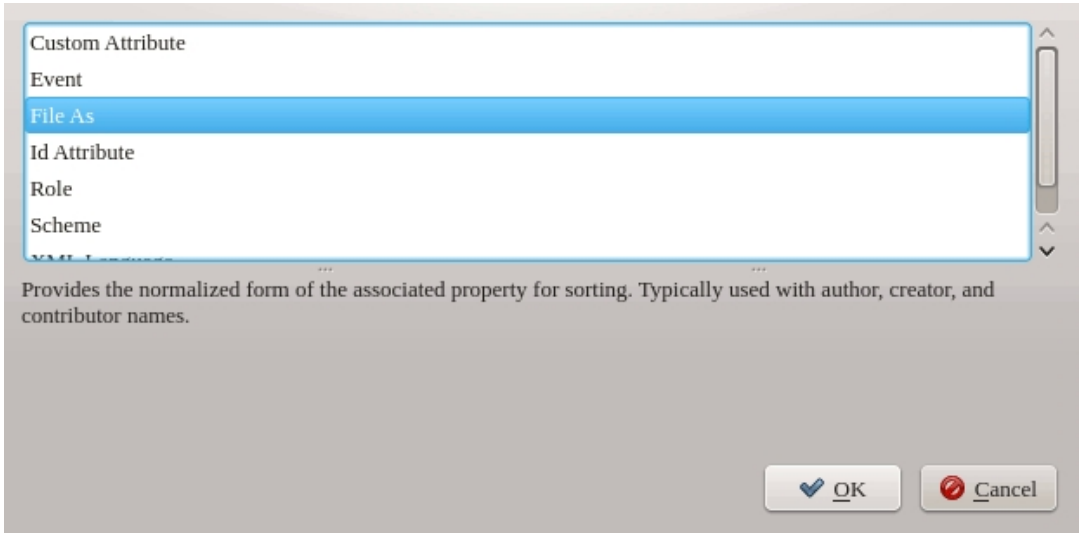
Enter the author’s name, for example enter Lewis Carroll.



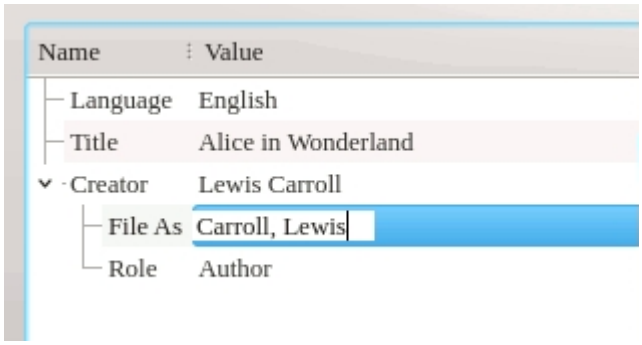
Specify that this Creator is the Author. Double-click on the Role value item. That presents a dropdown menu that allows you to select the Author entry.



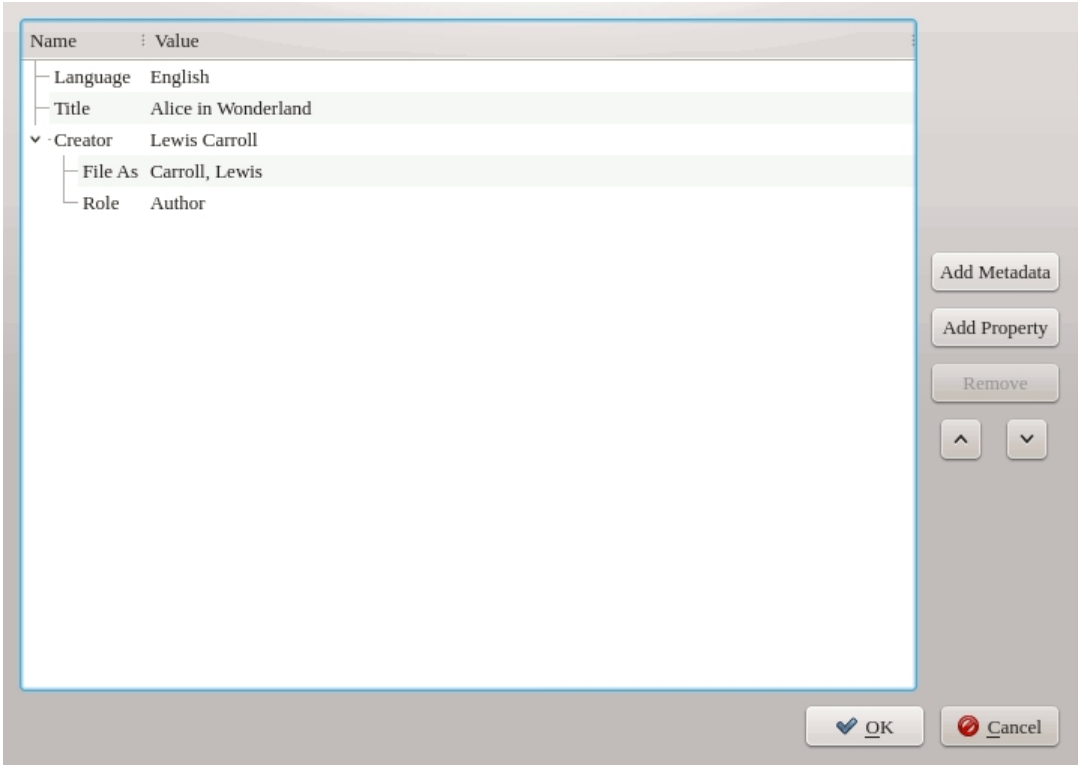
Add the property File-As to the Creator item. This is actually optional. But it is usually included as it controls how the author's name is sorted. Click the **Add Property** button. That button posts a dialog that allows you to select the property File-As.



Select the "File As" entry, then double-click the Value field and enter Carroll, Lewis.



This is how your data might look when filled in:



Once you have updated the information, you can click **OK** and then save your EPUB again.

> You may have noticed that there is a lot of room for other values in the Metadata Editor. Only the Title, Author, and Language are required for EPUBs, but if you want to add more values using the **Add Metadata** or **Add Property** buttons you can.
>
> See the Metadata Editor chapter for more details.

Add a Cover Image

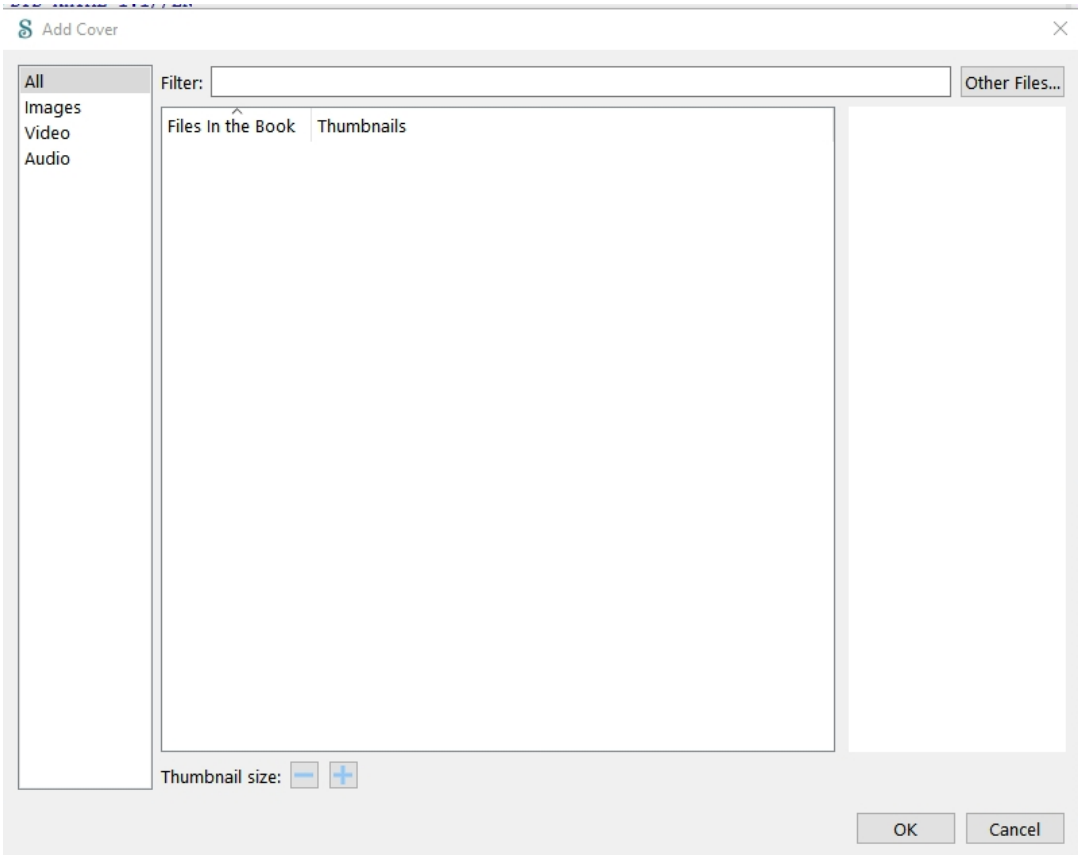
Every book should have a cover image, and Sigil can make adding one to your EPUB easy. You will need to have an existing cover image available to use (and have permission to use it). The exact size of the cover is up to you and the guidelines for your device, however here are some general ideas about what is expected.

- **Kindle Direct Publishing** – 2,560 × 1,600 (1.6:1 aspect ratio)
- **Novels and Non-Fiction** – 2,560px × 1,600px (1.5:1 aspect ratio)
- **Illustrated Books** – 2,800px × 3,920px (1.4:1 aspect ratio) or 3,000px × 3,600px (1.2:1 aspect ratio)
- **Audiobooks** – 3,200px × 3,200px (1:1 aspect ratio)

You should check with your book vendor and with the specifications for your targeted devices for the details on proper cover sizes.

To add a cover, just select the menu item **Tools=>Add Cover**

That menu item will display the Add Cover Window.



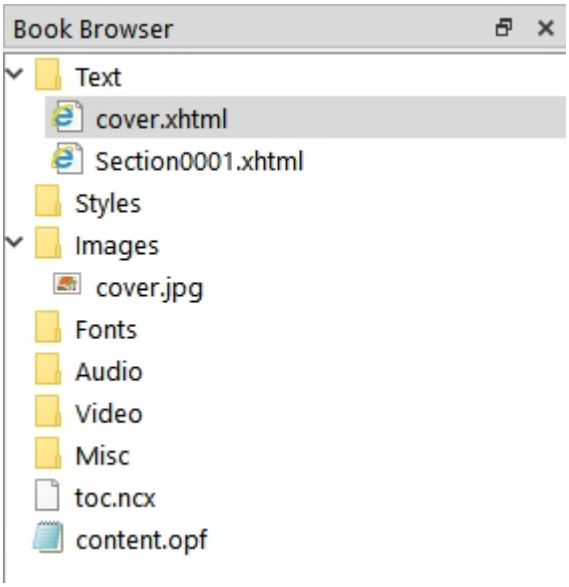
The Add Cover window shows you a list of images already in your EPUB. But since you haven't loaded any images yet, it's empty.

You can insert the image into your EPUB and make it the cover in one step by clicking on the **Other Files** button.

Select your cover image from your filesystem and click OK.

And that's it. Sigil has added your image to the EPUB and marked it as the cover image. It has also created an HTML file at the start of your book and inserted your cover image into the file so that it will be displayed at the start of the book:

The contents of your Book Browser show that the cover.xhtml file has been added to the Text folder, and the cover.jpg image has been added to the Images folder.



You should Save your book.

Create Files For Each Chapter

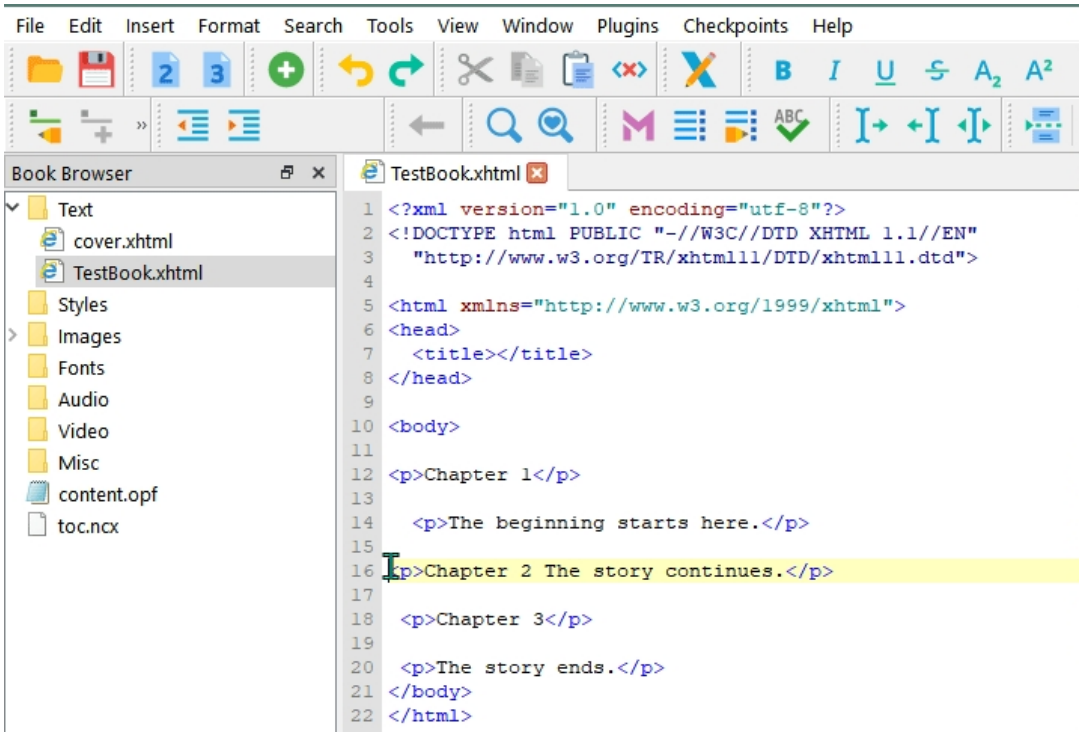
If you loaded an HTML file into Sigil, this usually means all of your text and chapters are in one large file. This is allowed, but the standard practice with EPUBs is to put each chapter in your book into a separate HTML file. The smaller file sizes are needed for some e-readers to read your book, but it also makes your EPUB much faster to display and easier to edit.

> Since most e-readers start a new file in the EPUB on a new page, it means your chapters will also start on a new page.

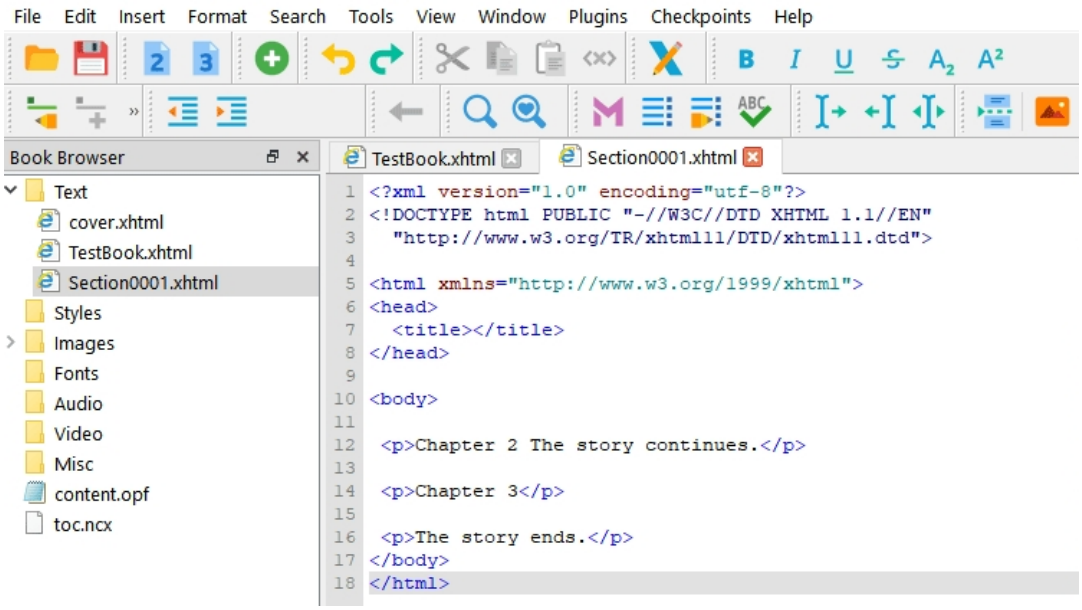
Create One File Per Chapter

The easiest way to split your book into separate files is to use the Split At Cursor button wherever you want to start a new file.

For example, to start a chapter on a new page, click just to the left of the chapter heading so that your cursor is at the start of the line:

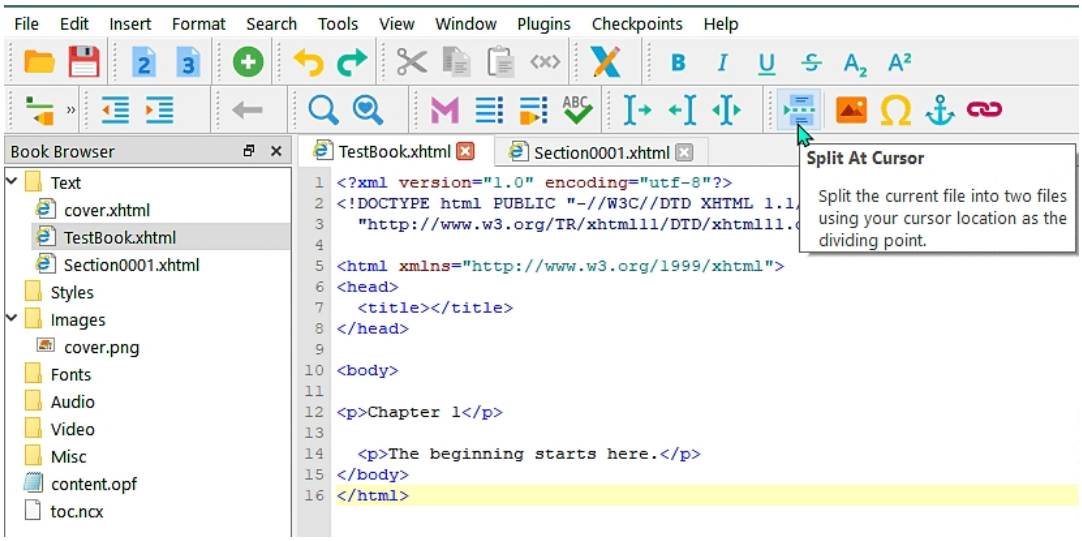


Now click the Split At Cursor button to split the text into two parts:



You can see that everything after where you split is now in a new file called "Section0001.html".

If you look at the original file you can see it only contains the text before where you split:



You can continue splitting every chapter in your book, and putting any section that should start on a new page in a separate file.

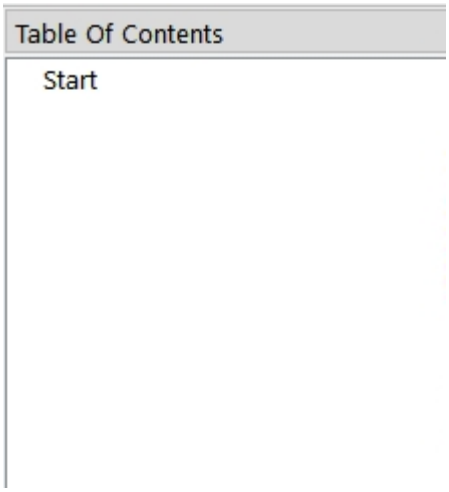
If you want to rename your files, see the Rename section. This includes how to rename many files at once, e.g. Chapter_1, Chapter_2, etc..

See the Splitting and Merging chapter for more details – including how to insert Split Markers and split every file at once.

Identify Your Chapters

EPUBs can provide a Table of Contents (abbreviated as TOC) to help readers find their way around your ebook. Sigil can display and generate a Table of Contents automatically from your ebook – you just need to tell it what your chapter names are.

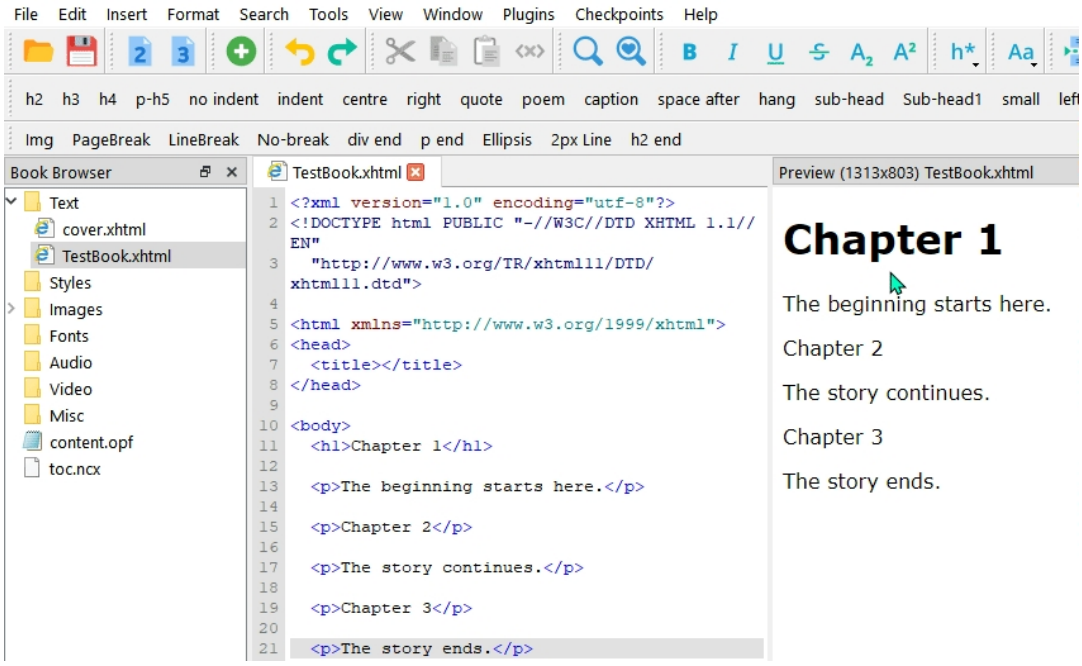
Sigil shows the Table of Contents in a separate window – on the right by default:



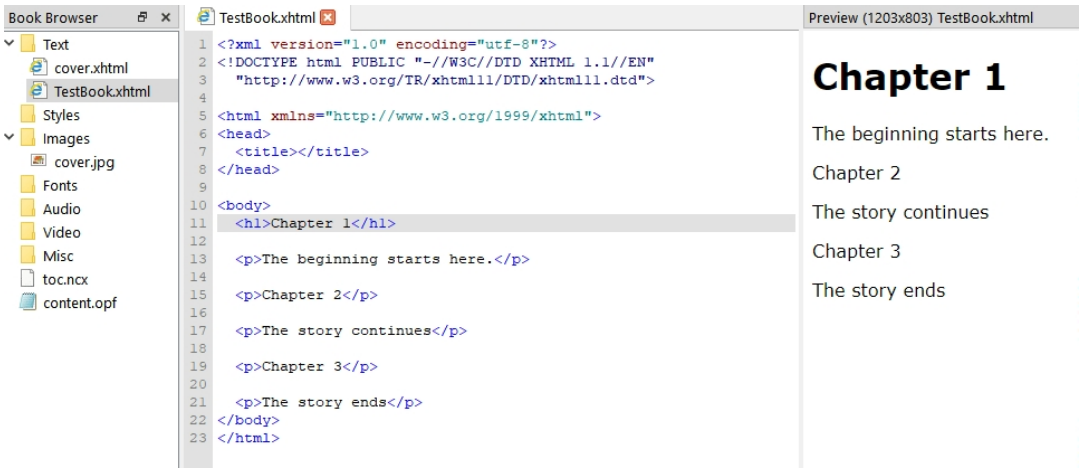
Mark Your Chapters

In order to create a Table of Contents you need to tell Sigil what your chapter names are. To do this just select each chapter name in your book and use one of the toolbar buttons to mark it as a chapter heading:

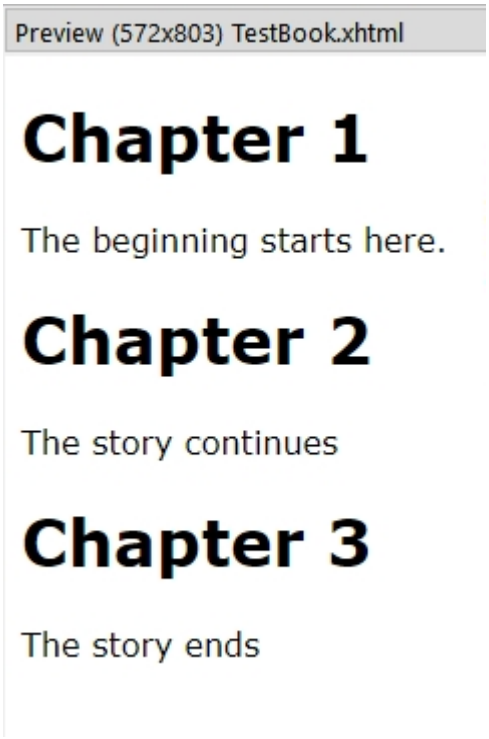
Just click anywhere in the paragraph that you want to be a chapter:



Click on the Heading Level 1 button to mark the text as a chapter heading:



You can then mark the remaining chapter headings in your book:



When you click on a heading in your text the Heading buttons will highlight the heading level.

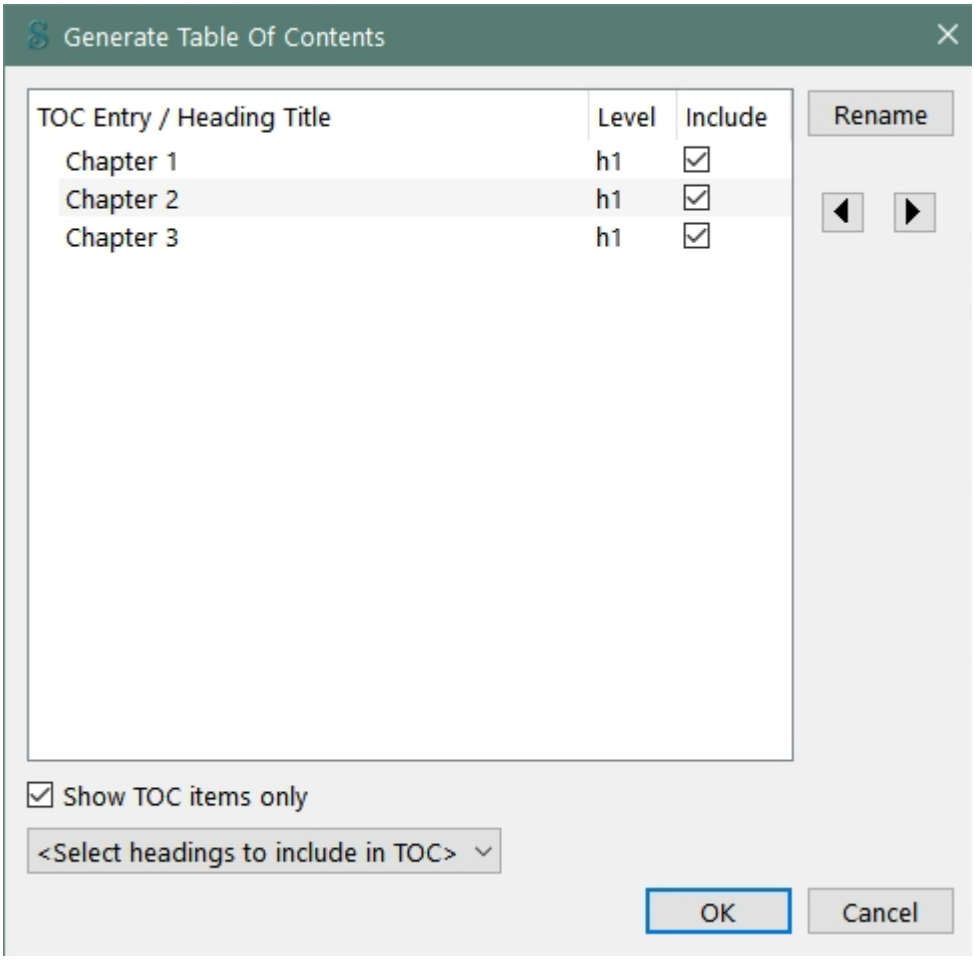
The other heading levels (h2, h3, etc.) allow you to create sub-chapters. The paragraph button p can be used to change your text back to a normal paragraph instead of a heading.

If you have multiple HTML files in your book, just double-click on them in the Book Browser to open them.

Generate A Table of Contents

In order for Sigil to generate a table of contents for you, you must have already identified the chapter names. See the previous tutorial (Identify Your Chapters) if you have not done this already.

Once you have marked all your chapter headings you can click the Generate Table Of Contents button or use the menu item **Tools=>Table Of Contents=>Generate Table Of Contents** to open a new dialog:

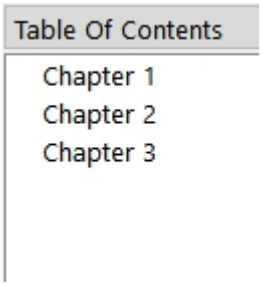


The dialog allows you to mark which headings to include or not include in your Table of Contents.

> Marking a heading to be excluded from the TOC is persistent. The next time you generate the TOC, that heading will remain excluded. See the Table of Contents chapter for more details.

Click **OK** when you are finished.

The Table of Contents window will be updated to show your current TOC – which includes the headings you selected:



If you click on an entry in the Table of Contents window you will jump to its location in the book.

You should re-generate the TOC any time you change your headings.

> If you also generate an HTML TOC as well, you should do that after you generate the EPUB TOC because the HTML TOC gets its information from the EPUB TOC, not directly from the HTML files.

Save your book if you haven't already.

> See the Table of Contents chapter for more details.

Create Links and Notes

— Linking your Text —

Sigil allows you to easily link text to other parts of your book, such as chapter names and notes, so that readers can select the link and jump to the identified text.

IDs

In order for Sigil to create a link from certain words in your book to other locations in your book, you first need to give an ID (effectively a unique name) to the text you want your link to go to.

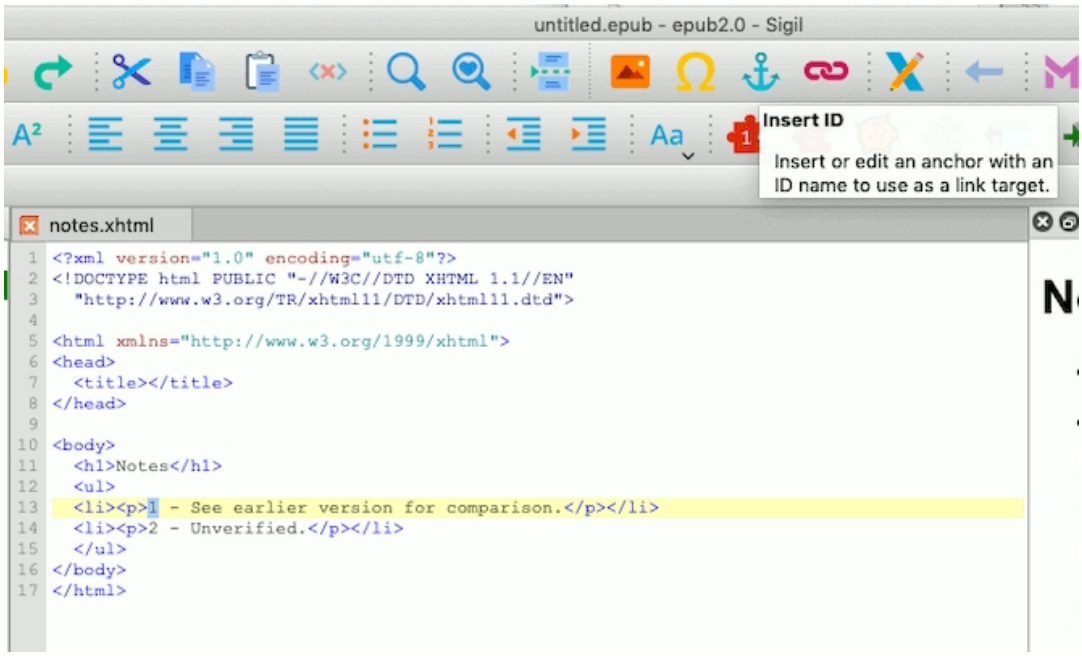
If the target for your link is just the top of a file, then you don't need to do anything since you can use the filename as the target ID.

But if you want to point to specific words in your book, then you need to assign an ID to the text.

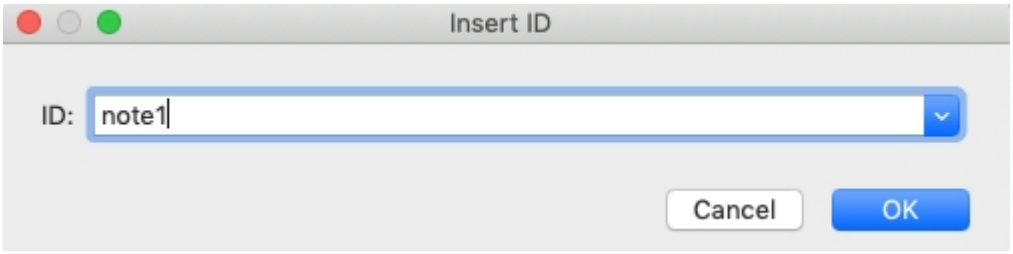
> An ID is just a unique name, like note1 that you will assign to a some of your text so that you can refer to it. ID names must be unique within an HTML file (and start with a letter), but its best if they are unique throughout the whole EPUB in case you move text around.
>
> IDs and links are usually created with the HTML anchor tag ("a") – so the ID is often considered the anchor ID for a link, hence the button used for inserting IDs.

Create an ID

To create an ID, select some text in your book and click on the Insert ID button :



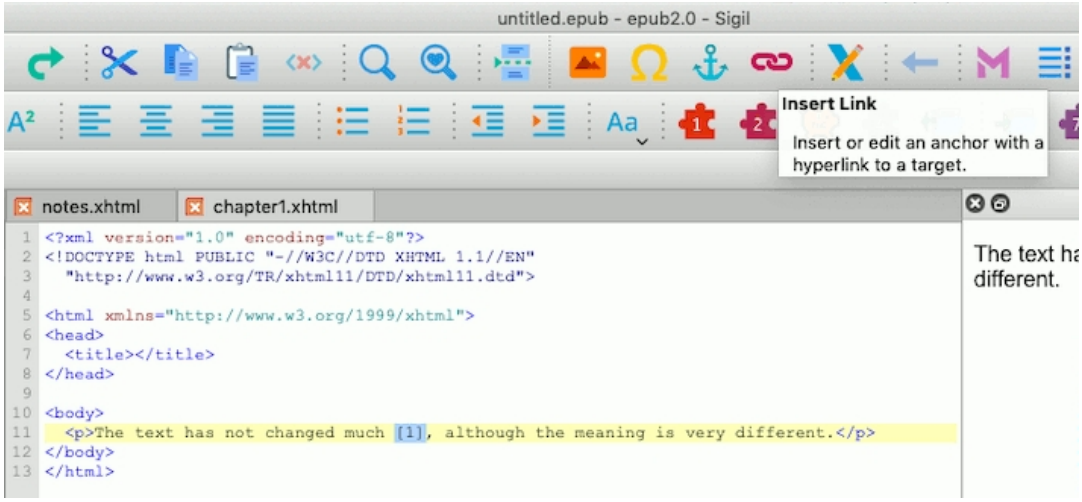
This will open the Insert ID dialog where you can type the name of your ID:



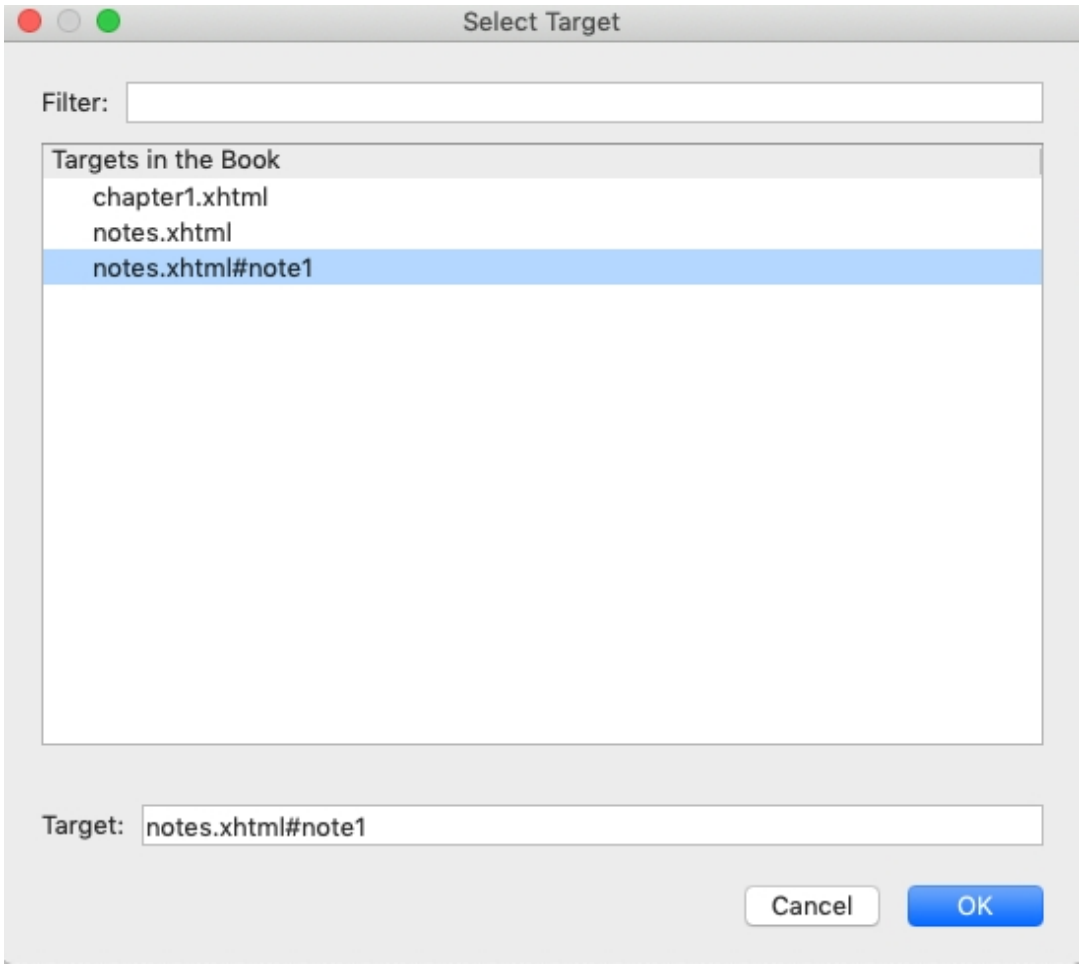
LINKS

A link is a way to point to your ID, and when selected by the reader it “jumps” the reader to the location of the ID.

To create a link in Sigil, select some text to use for the link and click on the Insert Link button :



This will open the Select Target dialog which shows you the IDs in your book that you can link to:



You will notice a number of different IDs in the list. At first it might not look like yours is there, so to find it, it helps to understand what a link looks like:

- **Filename.html:** This points to just a filename as the target of your link and tells the link to jump to the top of the file. This is what is used for the Generate Table Of Contents links.

- **Filename.html#ID1:** This points to a specific ID in a file. When you use this as the target of your link the reader will go to the exact location of the text with the ID in the file listed.

- **#ID1:** This is just the ID name, and implies that it is in the same file as your link so no filename is needed.

When you select a link you will see that the Target text box shows the actual link that will be used. Click **OK**, or double-click on an entry, and the link will be created for your text.

```
 9
10 <body>
11   <p>The text has not changed much<a href="notes.xhtml#note1">[1]</a>, although the
   meaning is very different.</p>
12 </body>
```

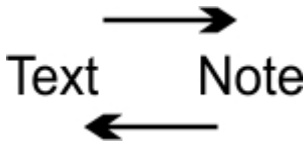
If you click on the newly created link in your text you will jump to where the ID is.

You can then click the Back button to return to where you clicked the link.

Reverse Linking

Most e-readers have a Back button that you can select once you follow a link to take you back to the original linked text. But for those that don't, you can create a link from your destination back to your original source text. This is often called reverse linking.

Reverse linking is just like any other kind of linking – it just means you have to create a link from the source text to your destination note, and also from your destination note back to your source text – so both source and destination will need IDs and links, and will be mirror versions of each other:



For example, in the typical case of creating endnotes:

- Select your note at the end of your book (or chapter) and assign an ID to it, e.g. note1.
- Select your text in the book and assign an ID for it, e.g. text1.
- Select the same text again and create a link to the note's ID.
- Select your note again (click on the link to jump you to the note and to test the link), and create a link pointing back to your text's ID.
- Click on the note to jump you back to your text to test the link.

A Quick Introduction To Code View

Code View

That's quite a lot of text, but it's not as jumbled as it appears. First, ignore everything at the top of the file up to and including the <body> tag on line 11 in this example. The "header" is just information required by HTML files and not something you normally need to change.

Your text starts after the <body> tag:

```
11 <body>
12   <h1 id="introduction">Introduction</h1>
13
14   <p class="h1subheading">— Sigil: The EPUB Editor —</p>
15
16   <p>Welcome to the Sigil User Guide - completely updated for Sigil 1.5.0.</p>
17
18   <p>Sigil 1.5.0 is the result of countless hours of volunteer effort to create the fastest,
     most stable and powerful version of the best free EPUB editing software available.</p>
19
20   <p>Jump right to the <a href="tutorials.html">Tutorials</a> to learn how to use Sigil step by
     step. Or view <a href="whats_new.html">What's New</a> for changes in recent releases.</p>
21
22   <p>Download Sigil from:</p>
23
24   <ul>
25     <li class="indent"><a href="https://github.com/Sigil-Ebook/Sigil/">The Official Sigil
     Website</a></li>
26   </ul>
27
28   <p>Help is available from other users and the developers of Sigil at MobileRead:</p>
29
30   <ul>
31     <li class="indent"><a href="http://www.mobileread.com/forums/forumdisplay.php?f=203">Sigil
     Forum @ MobileRead</a></li>
32   </ul>
33
34   <p>The Sigil User Guide includes documentation that will help you get the best out of editing
     your EPUB, including:</p>
35
36   <ul>
37     <li><span class="listheading"><a href="installation.html">Installation</a>:</span>
     Installing Sigil on Windows, MAC and Linux.</li>
```

You can see all the same words that are shown in Preview – the chapter name "Introduction" and several paragraphs of text. But you can also see the HTML code tags that tell readers how to display that text.

HTML Basics

A full description of HTML is beyond this tutorial, but understanding the basics will help you edit your text.[1]

If you look at the HTML version of your file in Code View you will see lots of words like <p> or <h1>, etc. These are the HTML tags that tell e-readers what to do with your text.

For instance, the HTML code below tells the e-reader that the text Chapter 8 is a level 1 Heading:

```
<h1>Chapter 8</h1>
```

The <h1> tag is the start of the text to be treated like a heading, and the </h1> tag indicates the end of the text included in the heading. This is what Sigil enters into your document when you click the icon. E-readers will typically make the text of a heading larger and center it on the screen.

The HTML code below tells the e-reader that the text This is a paragraph is an ordinary paragraph. Since it's a paragraph the e-reader will apply a certain formatting, perhaps indenting the first line.

```
<p>This is a paragraph.</p>
```

There are many more tags in HTML. The important point to remember is that the HTML tags tell e-readers how to format and display your content. They allow you to keep your content separate from your formatting which makes it easier to change how your book looks.

HTML tags can get more fancy since they can include attributes that modify how they are interpreted. For example, here is a paragraph with some tags that define an ID used for linking as shown in the last chapter:

```
<p><span id="note1">This is a note</span> for linking.</p>
```

The span tag is used around small sections of text that need to be formatted differently or to contain information. In this case the information is that the words "This is a note" has an ID of "note1". The span tags with the ID attribute are the code that Sigil inserts when you use Insert ID.

These are just a few examples of HTML code, but they give you an idea of what the majority of HTML is all about.

What is important to remember is that in Code View you have access to both your text and the formatting codes that control the look of your book – so you can find and change them.

See the Introduction, Code View and Stylesheets chapters for more details.

Editing in Code View

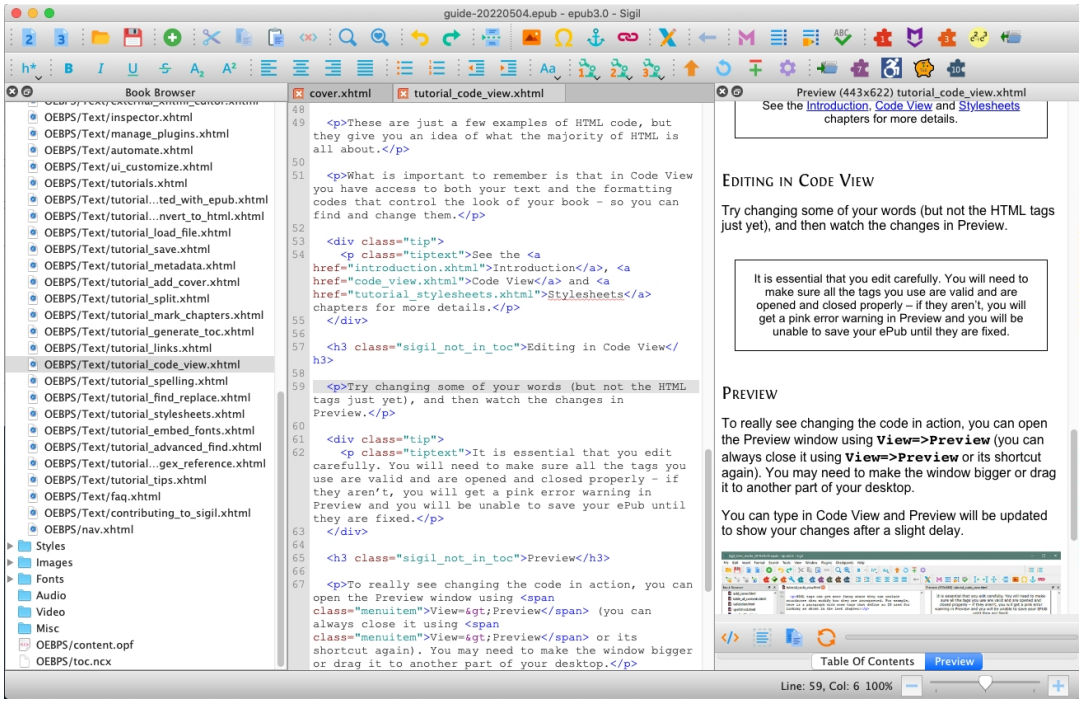
Try changing some of your words (but not the HTML tags just yet), and then watch the changes in Preview.

> It is essential that you edit carefully. You will need to make sure all the tags you use are valid and are opened and closed properly – if they aren't, you will get a pink error warning in Preview and you will be unable to save your ePub until they are fixed.

Preview

To really see changing the code in action, you can open the Preview window using **View=>Preview** (you can always close it using **View=>Preview** or its shortcut again). You may need to make the window bigger or drag it to another part of your desktop.

You can type in Code View and Preview will be updated to show your changes after a slight delay.



> Save often to have your code checked for any errors to catch them early.

[1] Many good HTML tutorial sites can be found on the Internet, W3Schools.com being one.

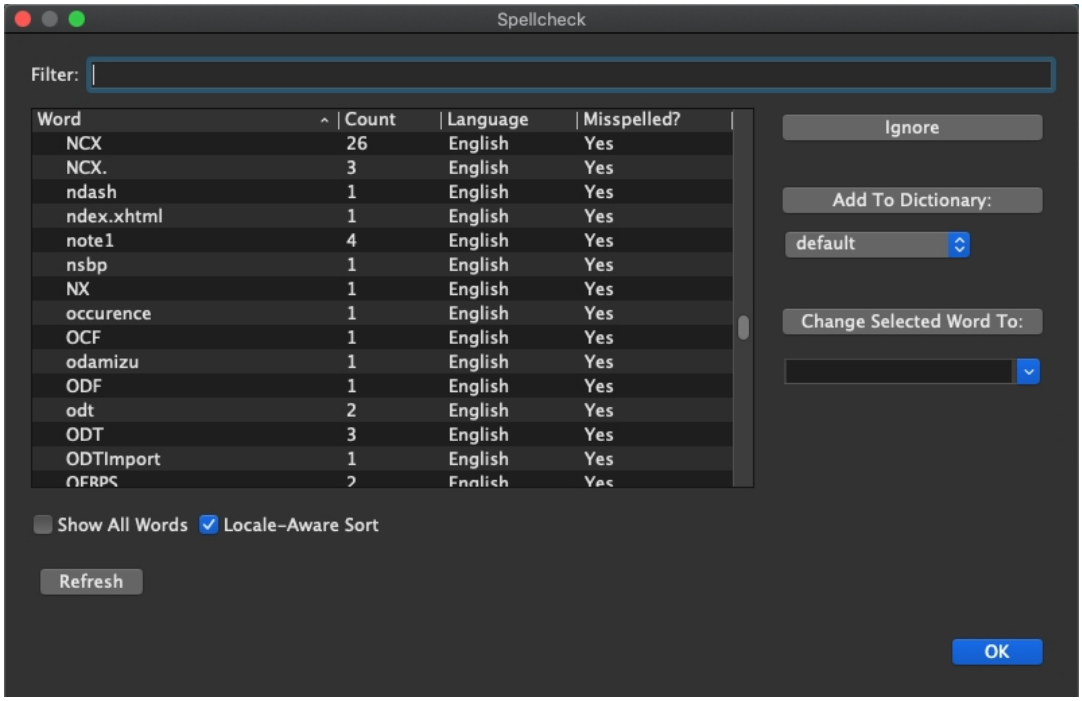
Check For Spelling Mistakes

Sigil can check the spelling in your book using standard dictionaries, and dictionaries you create.

Before checking the spelling in your book, you should first configure a dictionary using **Preferences=>Spellcheck Dictionaries**. You will need to select a primary dictionary (English, French, etc.). You can just use the "default" personal user dictionary to store any words you want to remember, or create a new user dictionary for each book.

Spellcheck

To check the spelling in your book click the Spellcheck button or use the menu item **Tools=>Spellcheck=>Spellcheck:**



This will open a dialog that shows you every word Sigil has identified as misspelled.

If you select a word you will notice that Sigil will find the first occurrence of that word in your book – and display it in Code View. This is useful to see the word in context, but can be a little confusing if you are not used to Code View, but is necessary to ensure Sigil can find your misspelled word accurately.

Once you have selected a word, you have several choices:

- Use **Ignore** to skip marking this word as misspelled until you switch to a new book or restart Sigil. Sigil will not remember to ignore this word if you restart.
- Click on the button **Add To Dictionary** to add the word to the dictionary you select from the list below the button. You will start with just a "default" dictionary in the list unless you created additional user dictionaries in Preferences.
- Use **Change Selected Word To** to change every occurrence of the selected word in your book to the word you enter below. The dropdown list contains a list of suggested alternatives, or you can type your own word.

The other options available to you are:

- **Refresh** can be used if you edit your book while Spellcheck is open. You do not need to use it if you are just ignoring, adding, or changing a word.
- **Show All Words:** Check this to show all unique words in the book and not just the misspelled words.
- **Case-Insensitive Sort:** Enable this to sort words as AaBbCc instead of ABCabc.

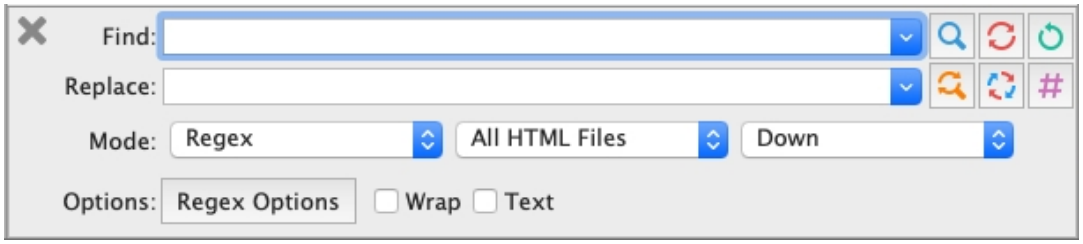
You may also notice that in Code View, misspelled words are optionally underlined. This is another way to check spelling with Sigil.

See the Spellcheck chapter for more details.

Edit With Find & Replace

In Sigil, Find & Replace can be used to find any text in any of your files. And even more important, it can be used to change text – but only in Code View.

To open the Find & Replace, use **Search=>Find & Replace** or its shortcut. The Find & Replace window will open just below your editing window, and will stay there until you close it:

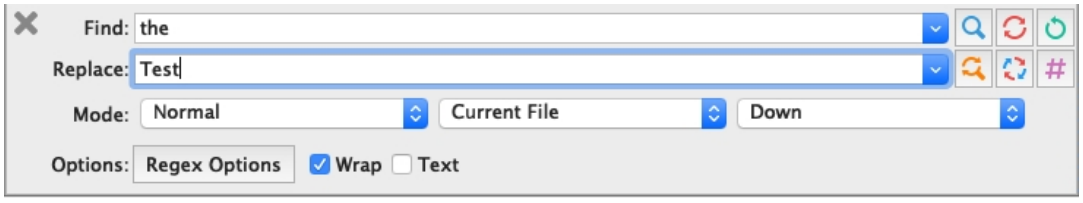


The Find and Replace text boxes are where you enter what to look for and change. The buttons on the right (Find, Replace, Restart, Replace/Find, Replace All, and Count All) are used to run your search. The Mode dropdown boxes on the bottom allow you to change what to search for, where to search, and which direction to search in.

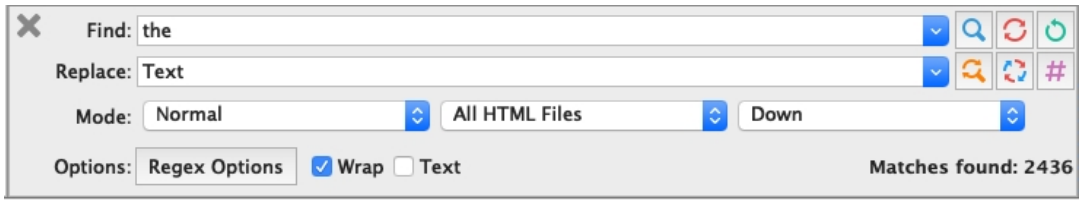
Each dropdown and button has a tooltip with more details.

Using Find & Replace

To make a simple change enter some text to search for in the Find box, and in the Replace box enter the new text to replace it with:



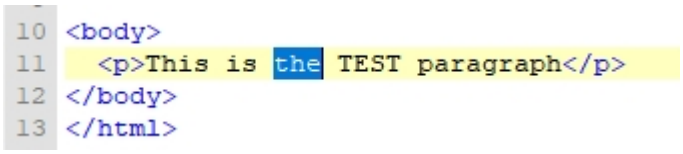
Click on **Count All** to show how many matches there are for the text in the Find box (Count and Find ignore what is in the Replace box):



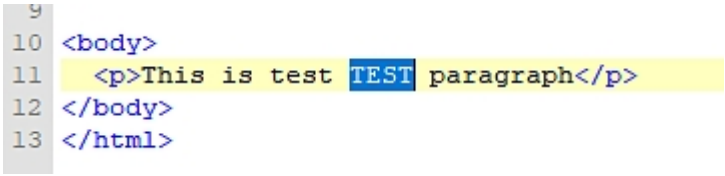
You'll notice that you switched to Code View if you weren't already using it! This is because all major searches are done in Code View. You should stay in Code View for this tutorial.

Note that the number of matches is shown on the bottom right.

Now click on **Find**. The next occurrence of the word in the Find box will be highlighted:



Click on **Replace** – this changes the text to what is in the Replace box and leaves the new text highlighted:

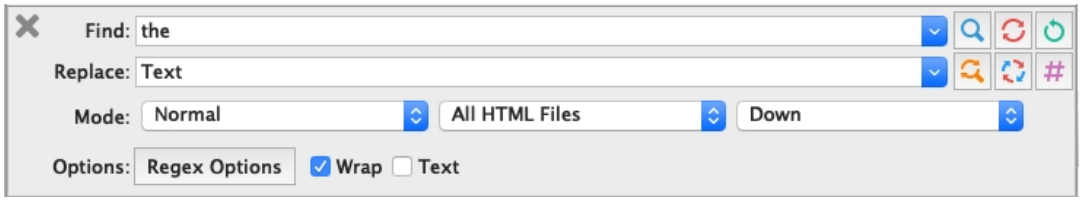


To make it a little easier to find and replace one occurrence after another, you can use the **Replace/Find** button. When you click on this button Sigil will replace the selected text if it matches, and then find and highlight the next match.

If you are satisfied that the Replace is working okay, you can use **Replace All** to update all the matching text in your document at once instead of one at a time.

> Always use Count All and possibly a few Replace commands before using Replace All – to make sure you have entered your find and replace strings correctly.

You may have noticed the selection box containing "Current File". If you change "Current File" to "All HTML Files", your Find or Replace will be done across every HTML file in your EPUB:



This is why editing in Sigil can be done quickly.

The last main part of Find & Replace is its ability to use Regular Expressions – but see the Advanced Find & Replace chapter for more details.

Use Stylesheets

— Formatting with Style —

This tutorial describes stylesheets and how to use them in your EPUB.

A stylesheet (or CSS – Cascading Style Sheet) is a list of instructions that tells e-readers how to display the text in your book. More specifically it tells the e-reader how to style text using HTML tags used in your book. For example, this is a style for a paragraph of type (or class) "author":

```
p.author {
    font-size: 1.5em;
    font-weight: bold;
    text-align: center;
    margin-top: 4em;
    margin-bottom: 2em;
}
```

Understanding and using CSS is the best and most flexible way to layout your book exactly as you want.

You can find other introductions to CSS online, including:

- W3Schools CSS Guide

> Using CSS can get quite involved, but if you become familiar with the basics it will make it much easier to understand why your book looks the way it does and how to modify its layout.
>
> This chapter assumes you are familiar with Code View since you need to use Code View to see the details of styles.

What is a Style?

Styles are used in a book to do things like "make this text blue", or "make that text centered", or "leave a little more space above this text". A style definition is just text that looks like:

```
color: blue;
```

which says make the color of the text blue.

But styles aren't very useful unless they are applied to something. In the case of an EPUB styles are applied to the HTML elements in your document, tags like `p` (paragraphs), `h1` (level 1 headings), `div` (divisions), `span` (spans), etc.

There are 3 ways styles can be used in an EPUB:

- **Style Attribute:** A style applied directly to one specific HTML tag in the document.
- **Style Element:** A group of styles in one document that can only be used by that document.
- **Stylesheet:** A separate CSS or stylesheet file that can be used by many HTML files and is therefore the most flexible approach.

A separate file is the recommended way to use styles. The descriptions below of each approach will help you understand why.

Style Attributes

To apply a style directly to an HTML tag, you can use the `style` attribute in the tag. For example, to apply the style `color: blue` to a paragraph you would change

```
<p>Hello</p>
```

to:

```
<p style="color: blue">Hello</p>
```

This works perfectly well if you just want to change a few things and rarely update the formatting. But if you later want to change from using bold to italic for certain words then you will have to edit every place in the book that set the specific style.

Styles attributes are rarely used unless they were created by WYSIWYG software or from an automatic conversion and should be avoided.

There is a much better way to manage styles: using style selectors.

Style Selectors

A style selector is a way to give a particular style a name, and then use the style in the document just by referring to its name. More specifically the style selector is used to select which elements in your document should use the style you define.

For example, if you wanted to style all paragraphs in your EPUB to be centered, you could define a rule for a style selector as follows:

```
p {
    text-align: center;
}
```

This defines a style selector p that will match (or select) all paragraph elements p.

Just using the tag name as a selector is useful for changing every tag, but you can also select specific tags.

One of the most common ways to select specific tags is by using a class name in the tag and the selector. A class is just a name that you choose (like "firstparagraph" or "example").

For example if you wanted to style the first paragraph in a chapter slightly differently than other paragraphs, you could define a style selector as follows (an "em" is one character width):

```
p.first {
    text-indent: 0em;
}
```

This defines a style selector that will match (or select) paragraph tags that use the class name first. You could then use that style in your document for a paragraph by assigning the class name first to the paragraph as follows:

```
<p class="first">First paragraph in a chapter</p>
```

The great advantage of this abstraction is that you can now update the values of the style (to change the indentation or other settings) without having to update your text.

> It is common practice to use class names that refer to the purpose of the style and not what the style does in case you change the styling later. For instance, use span.placename instead of span.bold in case you want to change your place names to italic later.

Now that you know what a style selector is and how to use them in your text, you need to actually put your style selectors somewhere in order to allow them to work.

In an EPUB you can store your stylesheet in the same file as your text (using the style element), or in separate files (external stylesheets or CSS files). You should avoid storing the styles in the HTML files, but it's explained here in case you see it in your files.

Style Elements

If you store style selectors and rules in the same file as your text they can only be applied to the HTML file they are in and cannot be shared by other HTML files.

When using these internal styles, the style definitions are stored inside a style tag at the top of your HTML files. If you open a file in your book and switch to Code View, you will see that the top of the file contains text between two head tags. This section of your document contains information about the file, and one of the things you can put here is the style tag that defines your internal style rules.

Here is a typical use of internal styles (they are surround by the style tag):

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
 "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html >
<head>
<title></title>

<style>
	h1 { text-align: center; }
	p.boldtext { font-weight: bold; }
</style>

</head>
```

Internal styles are a way of grouping styles together and using style selectors, but when you have many files in an EPUB it becomes a bit confusing as you have to have separate style rules in each file.

The best way to manage styles is to use a separate stylesheet file to hold your style rules.

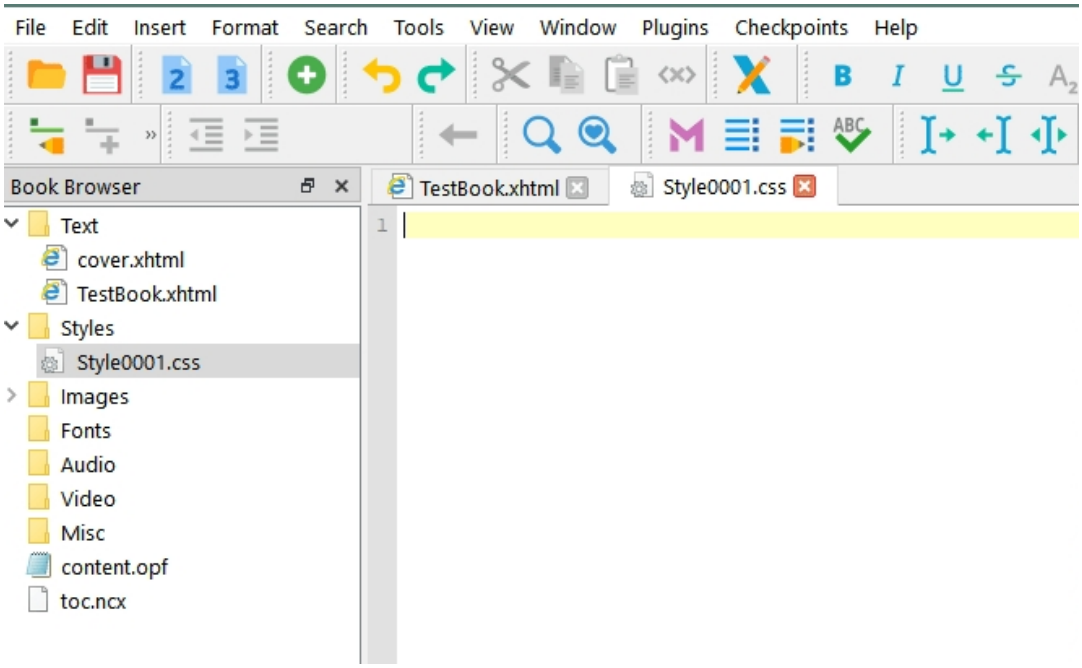
> You can convert or copy your inline styles and your style elements into a separate CSS file. Just remember to link your stylesheet to your HTML file and to delete the old style attributes and elements.

External Stylesheets – CSS Files

If you create a separate file to hold all your style rules, they can apply to the entire book. They are also much easier to find and edit, and updates are much easier to manage as you add and remove files. It also helps to ensure your book is formatted consistently as you can see all your styles in one place.

In Sigil, external stylesheets are stored in the Book Browser in the Styles folder.

To add a new blank stylesheet to your EPUB, use the menu item
File=>Add=>Blank Stylesheet:



As you can see the stylesheet is empty.

You can copy or type the styles that you want into the file, and edit it at any time. For example, if you open the styles.css file in an EPUB you might see something like this:

styles.css

```
    text-align: center;
    margin-top: 4em;
}

div.tip {
    text-align: center;
    margin-left: 1em;
    margin-right: 1em;
    margin-top: 2em;
    margin-bottom: 2em;
    border-style: solid;
    border-width: 1px;
}

div.toolbar {
    margin-top: 2em;
}

img {
    max-width: 100%;
}
```

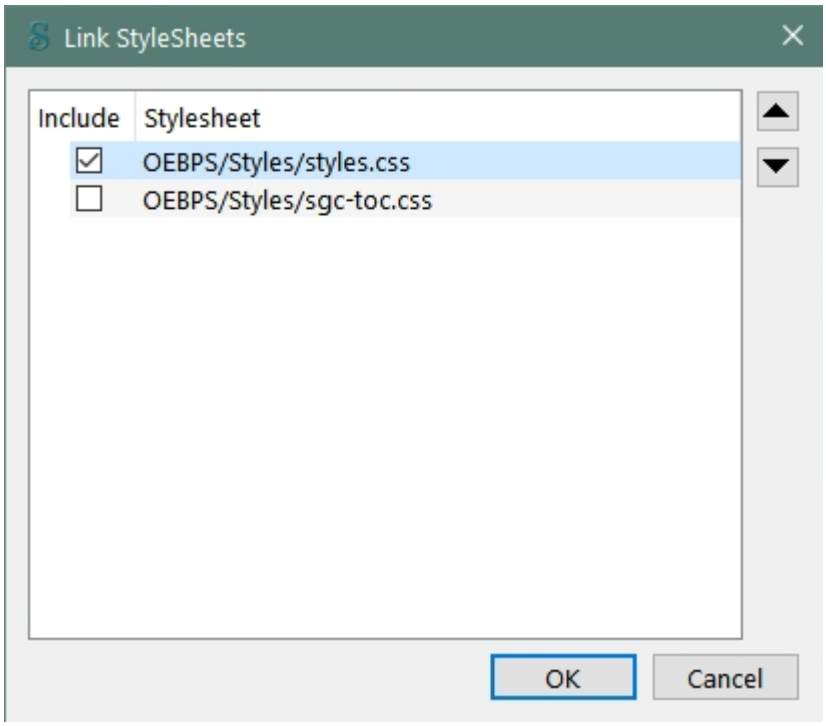
Even though you have created the external stylesheet, it's not actually used by anything yet.

To use it, you need to tell your HTML files to use the stylesheet – by linking to them as described below.

Link your HTML files to a Stylesheet File

To tell your HTML files which CSS file to use you need to link the stylesheet to your HTML file:

- Select one or more HTML files in the Book Browser.
- Right-click on the selection.
- Choose Link Stylesheets from the context menu:



- Click on the stylesheet(s) you want to be used for the files and select **OK**.

You can always check to see what what, if any, stylesheet is being used by a particular html by looking at the ‹head› section of the file.

```
<head>
  <title>Use Stylesheets</title>
  <link href="../Styles/styles.css" rel="stylesheet" type="text/css"/>
</head>
```

If you link multiple stylesheets they will be applied in order, so the styles in the last stylesheet will override the styles in the preceding stylesheets.

What Styles Should Be Used?

The styles you use for your text are entirely up to you.

The best way to learn how to use styles is to find an EPUB that contains formatting you like and look at what its CSS files contain and how it uses those styles in the book.

The styles will be applied in the order in which they appear in the stylesheet.

It is important to remember this as an error in one particular style definition results in all the following styles to be ignored.
It is advisable to check your stylesheets carefully as any errors may not be apparent until the book is viewed in an external e-reader.
You can use the menu item **Tools=>Validate Stylesheets With W3C** to check for errors.

Include Custom Fonts

— Embedding Fonts In Your Book —

This tutorial shows you how to use custom fonts in your EPUB.

> One of the benefits of e-readers is that you can choose your own font, so it's usually best to only use embedded fonts to highlight specific types of text – such as chapter headings, quotes, etc. – and not to change the font of the entire book.

To use a font in an EPUB you need to load the font into the EPUB, define a style for the font, and then apply the style to your text.

> You will need to know about stylesheets to embed fonts so you should read through the Stylesheets chapter first.

Adding Fonts to your EPUB

To add fonts to your book, use the Add Existing Files button . This will display a dialog allowing you to select one or more files from your computer. Select as many font files as you want.

> Many fonts are not free. Before you include fonts in your EPUB please make sure that you have a license or permission to use them. There are many free fonts available online if you search for them.
>
> A number of different font types are supported by the EPUB specification including OpenType (.otf), TrueType (.ttf),and Web Open Font Format (.woff or .woff2) formats. Almost all e-readers support at least TrueType and OpenType formats.

The fonts will be added to the Fonts folder in the Book Browser. However, they are not yet actually used anywhere in your book – they are just available to use.

Defining Styles for Your Font

Once the fonts are in your EPUB, you must define a style that includes them.

Add or open a stylesheet that is linked to your HTML files (see the Stylesheets chapter for more details).

Add a definition of the font, e.g.:

```
@font-face {
    font-family: 'Garamond';
    font-weight: normal;
    font-style: normal;
    src: url('../Fonts/Garamond.ttf');
}

@font-face {
    font-family: 'Garamond';
    font-weight: bold;
    font-style: normal;
    src: url('../Fonts/Garamond Bold.ttf');
}

@font-face {
    font-family: 'Garamond ;
    font-weight: normal;
    font-style: italic;
    src: url('../Fonts/Garamond Italic.ttf');
}
```

Font names are case-sensitive!

You must define one @font-face rule for each style of font (normal, bold, italics, etc.).

When using the name of a font-family the font name must either be quoted or unquoted every time you use it – you cannot mix quoted and unquoted names. Always using quotes around the font name makes things more consistent and avoids having to check if a name is valid without quotes.

The font still isn't actually being used in the book yet, that's the next step.

Apply Your Font to Text

To make use of the font you need to create another style that you will then assign to sections of your text.

So, for example, to use your new font for headings, define the style for the heading tag as follows:

```
h1 {
    font-family: 'Garamond', serif;
}
```

Or to certain paragraphs:

```
p.longhand {
    font-family: 'Garamond', serif;
    font-style: italic;
}
```

Or if you want, although it's not recommended, you can apply the font to all text in the book:

```
body {
    font-family: 'Garamond', serif;
    font-weight: bold;
}
```

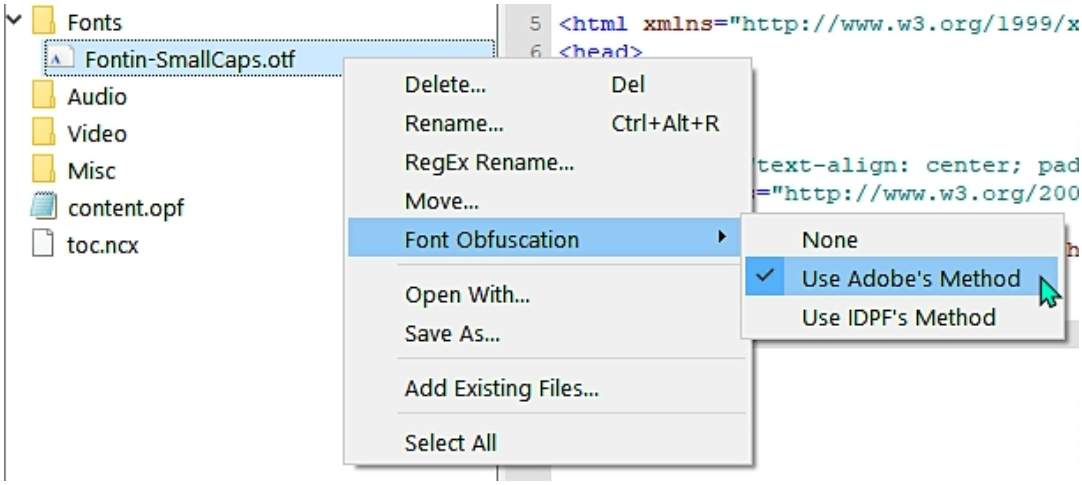
> The font-family rules above instruct the e-reader to use your font if possible for the text, but if it's not possible use a generic serif font.

Font Obfuscation

When you use a font in an EPUB, the font file could be extracted by someone and used somewhere else. If your fonts are licensed you may want to prevent this.

You can use Sigil's font obfuscation menu to scramble the font file using one of two standard methods, Adobe or IDPF – which one is up to you, but you should test on your target e-reader to make sure the font is still readable. Obfuscation is not encryption – it just modifies the font file making it hard to use elsewhere while still being useable by readers to display fonts.

To use font obfuscation, right-click on your fonts in the Book Browser window, right-click and select Font Obfuscation from the menu and choose a method to use for obfuscation.



Font Subsetting

A better alternative to Font Obfuscation might be to use Font Subsetting tools to restrict the font descriptions to only the exact characters used in your text. The act of subsetting a font, renders the font file useless for any other more general use. There are plugins available for Sigil such as the **"ePUB Optimizer"** and **"Subset Fonts"** plugins that can subset a font. You can use the Characters Report of Sigil to provide the list of characters actually used in your EPUB.

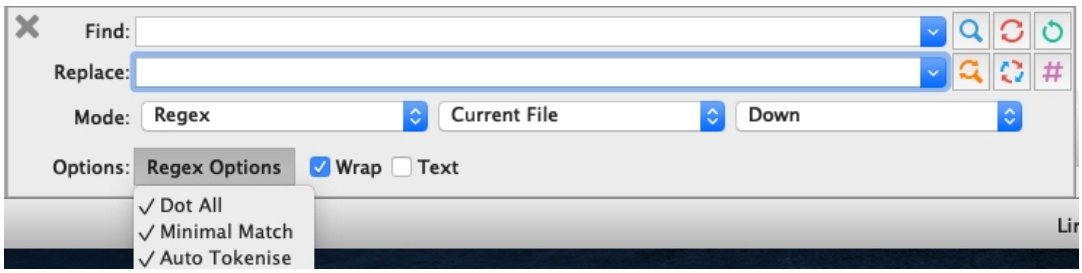
Testing Your Font

Not all devices will show embedded fonts. Even Sigil may not show all your embedded fonts. Its best to test on your target device to check if your fonts are displayed as you expect.

Advanced Find & Replace – Regex

The real power of Find & Replace in Sigil is provided by its support of Regular Expressions (also called regex). Regex is basically using patterns instead of plain text for searches to make matching text more flexible.

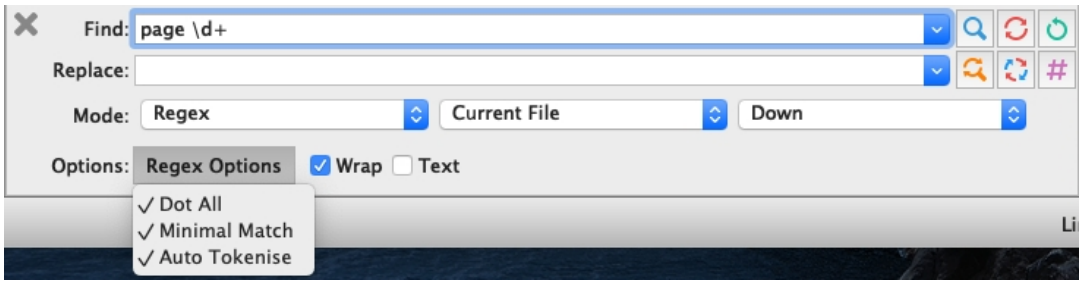
You can use regex in Find & Replace by changing the mode of the search from Normal to Regex.



Regex is a very important tool in Sigil, but only a very brief introduction can be given in this tutorial. See the next chapter (Regular Expression Reference) for a reference list of regular expressions.

For example, instead of searching for very specific text to replace or delete, such as "Page 123", it would be much more useful to search for text that contains "Page" followed by a number – that's exactly what regular expressions can provide.

Instead of putting your actual text into the Find or Replace boxes, you can substitute it with special regex codes. So in the example above you could search for "Page \d+" instead of "Page 123" because "\d+" is a regex code that means any digit (the "\d") repeated at least once (the "+").



Using Regex to Replace Text

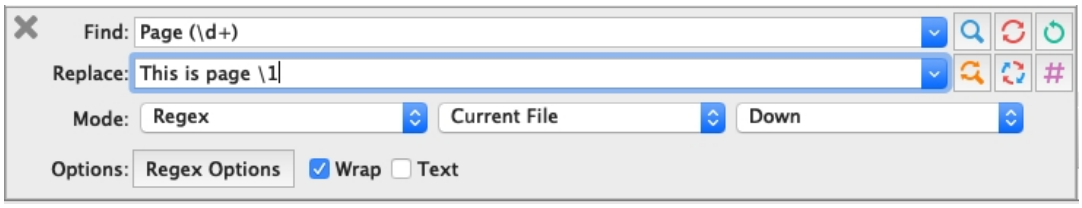
But even more useful is that you can use regex to help with replacing text as well. In the example above, you might want to keep the page numbers but replace the word "Page". You could use then use something like:

- **Find:**

```
Page (\d+)
```

- **Replace:**

```
This is page \1
```



This will replace any occurrence of "Page" followed by a number with "This is page" followed by the same number:

- **Before:**

```
Page 418
```

- **After:**

```
This is page 418
```

The first point to note about this replacement is the use of the regex code (\d+) in the Find box. The \d+ code tells the search to look for numbers as above. But the use of parenthesis around the code tells the search to remember what those numbers were – to remember anything matched in the parenthesis. The other point to note is the use of the regex code \1 in the Replace box, which tells Replace to substitute the characters remembered in the Find statement for the string \1 wherever it finds it.

Using Regex to Change Formatting

As a further example of regex, this is how you might change the formatting of certain text into chapter headings.

Let's say that you have an imported HTML file that contains lots of chapter headings, but none of them are marked using the h1 heading tag. Instead they are all marked as paragraphs like this:

```
<p>CHAPTER 7</p>
```

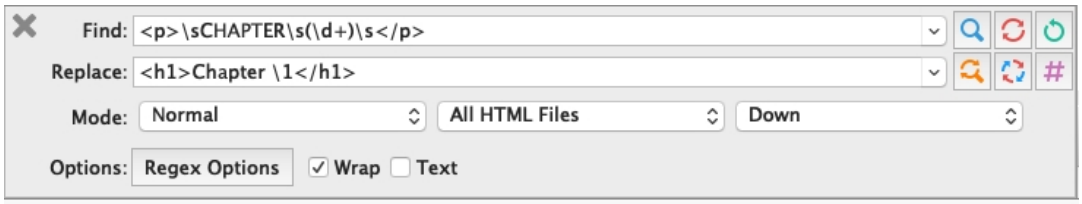
Assuming every paragraph like this is a chapter heading, you could use this regex:

- **Find:**

```
<p>\sCHAPTER\s(\d+)\s</p>
```

- **Replace:**

```
<h1>Chapter \1</h1>
```



That's quite a lot to digest, but if you look carefully you can see that it's very similar to the Page number example above in that it's remembering the digits in the chapter name and using them in the replace.

It's the Find that is the most interesting. It breaks down like this:

- – Look for a starting paragraph tag.
- \s – Regex code to match any white space (blanks, tabs, etc.).
- CHAPTER – Match the word CHAPTER (Regex is case-sensitive by default).
- \s – Regex code to match any white space.
- (\d+) – Regex code to match any number of digits in a row and remember them.
- \s – Regex code to match any white space.
- – Look for an end paragraph tag.

You could just use a space instead of \s but \s is more flexible since it will match any number of blank spaces and tabs.

So the results of that search could be:

- **Before:**

```
<p>  CHAPTER    14</p>
```

- **After:**

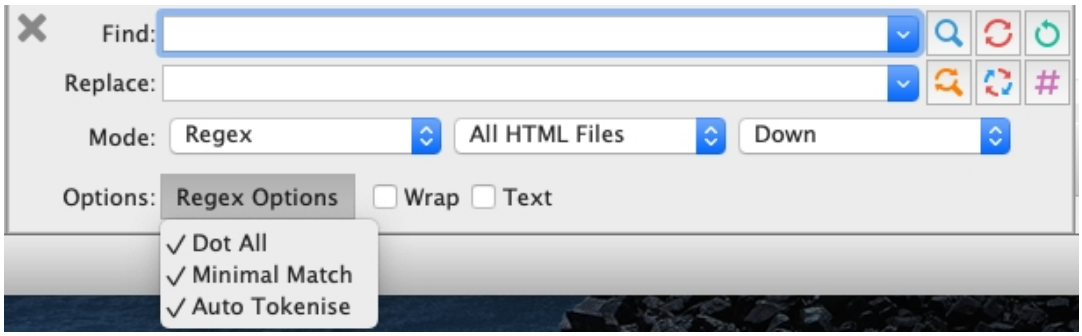
```
<h1>Chapter 14<h1>
```

It becomes more difficult to use regex if the formatting used in a file is not consistent – in these cases you may have to do several Find & Replace passes and even resort to making the updates one by one manually.

This is only a very basic description of what regex is, but it gives you an idea of what is possible. The trick with regex is getting the right find and replace strings, and that only comes from reading, asking questions, and testing.

Helpful Options

You may have noticed the extra Options in Find & Replace: DotAll, Minimal Match, and Auto-Tokenise. All of these options are useful when using Regex. If you are starting out using regex, then the recommendation is to check each of the boxes:



This is what each of the options is used for:

- **DotAll:** This regex option prepends (?s) to all regex searches and is used when you want the ".*" pattern to match across separate lines.
- **Minimal Match:** This regex option prepends (?U) to all regex searches and is used when you want a pattern to match the shortest occurrence instead of the longest match.
- **Auto-Tokenise:** When using Ctrl-F on selected text to add text to Find & Replace, this will convert spaces to \s and escape certain characters so they are suitable for regex.

See the Regular Expression Reference chapter for more details.

Regular Expression Reference

— Regex Reference —

This tutorial is basically a quick reference for Regex.

Regex is a very powerful way for you to edit and cleanup your EPUB when using Find & Replace. However, when you first start using regex it can be confusing because there are a lot of commands you can use. This is a brief summary of some basic regex commands.

You can find more details about regex at these links (or ask for help in the Sigil forum):

- Regex overview at MobileRead
- An Online Regex tutorial/reference
- A one page command summary
- Regex as used by Python
- Official PCRE details – start at the PCRESYNTAX(3) section

Overview

When using regex commands in Find & Replace the purpose is to allow you to search for patterns instead of exact text. This means that you can tell Find & Replace to search for "a number" instead of "8", or to match any words between certain tags and then replace it with new tags or new words.

In order to allow you to search for patterns, you need to use regex commands in place of the specific text you want in your search. For instance, instead of searching for `8` you could search for `\d` (which means one digit) or `\d+` (which means one or more digits).

An example Find & Replace command using regex might be to change all level 1 headings to level 2 headings:

- **Find:** `<h1(.*?)</h1>`
- **Replace:** `<h2\1</h2>`

where the Find command searches for any text between the h1 start and end tags, and using `()` to save the text. (The start tag is left open in order to catch any class definitions, and the `?` tells `.*` to do minimal matching). The Replace command is used to replace the matched text with the saved text (indicated by `\1`) surrounded by the h2 start and end tags.

The following sections give a sample of common regex commands that you may find useful.

General Modifiers:

You can put these commands at the start of any regex to modify how the expression behaves:

- (?s) – search across lines when using .* since by default .* only matches within a line of text. This is done by the DotAll option in Find & Replace.
- (?U) – match the first occurrence of your string (also called minimal or ungreedy matching). This is very useful when using .* to avoid matching too much. This is done by the Minimal Match option in Find & Replace.
- (?i) – ignore case when matching.
- (*UCP) – enables unicode properties for the expression that follows.

You can combine modifiers, e.g. (?sU) is commonly used.

Single Characters:

You can use these regex commands to represent single characters:

- . – matches one character
- \d – match one digit
- \t – match one tab
- [abc] – any character that is a b or c
- [^abc] – any character that is NOT a b or c
- [0-9] – any digit from 0 to 9
- [[:alpha:]] – any letter, upper or lower case
- [^[:alnum:]] – any character that is NOT a digit or upper or lowercase letter

Multiple Characters:

These regex quantifiers or codes allow you to represent multiple characters:

- * – use after a character to mean 0 or more, e.g. .* means 0 or more of any character
- + – use after a character to mean 1 or more, e.g. \d+ means 1 or more digits
- ? – use after a character to mean 0 or 1 instance, e.g. \d? means none or 1 digit
- \s – matches whitespace, including space, tab, end of line carriage returns or newlines

And there is a special character that tells your search to stop at the first match it finds instead of the largest match which you can use instead of the general modifier (?U):

- ? – use after a quantifier to do a minimal search, e.g. .*? or \d+?

GROUPS:

In some cases you need to group your expressions together so that, for example, you can save them for a later replace, or to indicate a choice of words:

- (atext|btext) – atext or btext
- (group) – group the words together for later retrieval – you can use multiple groups

REPLACING:

When replacing text, one of the most useful features is to be able to use text that was matched by storing it in a group. Once you have grouped text you can reference it in your replace command:

- \1 – use in Replace to retrieve the value of a saved group (use \2 for the second group, etc.)

Tips

Here are some useful suggestions for checking that your EPUB is complete:

- Use the Metadata Editor button to set your Title, Author, and Language. And if you want, also set the Copyright, Publication Date, Subject, Description, and ISBN if there is one.

- Create an HTML file for your cover, and set the file and your cover image as type Cover.

- Right-click on your first text file and select **Add Semantics=>Text** (for EPUB2) or **Add Semantics=>Body Matter** (for EPUB3) to tell e-readers that this is where the book should open to.

- Use the Generate TOC button if you generated your TOC, and re-create your HTML TOC if there is one, to make sure you have the latest updates.

- Test your TOC links work as expected.

- Test your links, footnotes or endnotes work as expected.

- Check your spelling with the Spellcheck button , and consider creating a separate dictionary for your book if it has lots of unique words.

- Use the FlightCrew and/or EpubCheck plugins to find EPUB errors then fix them – see Validation for typical error messages and what to do about them.

- Check your stylesheets are valid using **Tools=>Validate Stylesheets with W3C**.

- Link your stylesheets to all files that need them.

- Remove unused images, audio, or video files or unused style classes using **Tools=>Delete Unused**…

- Run **Tools=>Reports** and look at the reports for anything unexpected (extra files, broken links, classes used but not defined, etc.).

- If you are using CSS files for your styles, search for, <style> tags used in inline CSS, or style= attributes used by inline styling and convert them to CSS stylesheet files.

- If you are really keen, search for and replace code like , <br />, empty paragraphs   , empty tags, empty attributes like src="", href="", or ="", change – to –, change ' to ’, etc. – unless of course you want to keep them.

- Proofread on actual e-readers. Sigil can only give an approximation of what your book will look like. Despite EPUB being a standard, there are a lot of e-reader devices and software and they all have their own quirks and bugs.

And in case you missed it – test your book on actual devices and e-reader software!

FAQ

— Frequently Asked Questions —

Common Questions

Does Sigil Support EPUB 3?

Yes, Sigil fully supports reflowable EPUB2 and EPUB3 EPUBs. In fact, Sigil was designed to make it easy to create EPUB2-compatible EPUB3 ebooks. Simply select **Tools=>Epub3 Tools=>Generate NCX/Guide for epub2 e-readers** when you are finished editing your ebook.

Why does my book look okay in Sigil but not on my device?

Every device or program that displays your book uses slightly different software. Despite EPUB being a standard, this means there may be differences in how your book appears on each device. In some cases, although it looks okay in Sigil, it may not look okay on a specific device. And in other cases, it might not look ok in Sigil but it might display fine on your e-reader. You must test your book on as many devices as you can.

Where to Get Help

Where can I get help with Sigil?

If you have any questions about Sigil not covered in the Sigil User Guide, then visit the Sigil Forum at MobileRead where the users and developers of Sigil will try to help.

Where can I ask questions about EPUB?

You can post on the EPUB Forum at MobileRead if your questions are more about how to format and style something in an EPUB rather than how to do something in Sigil.

Where can I report bugs or request features?

Actual bug reports should be submitted via our Github Repository at https://github.com/Sigil-Ebook/Sigil. If you so desire, bug reports can also be posted to the MobileRead Sigil forum. Please include Sigil Version number, operating system (Windows 10, macOS, Linux Mint) and specific directions on how to recreate the bug or a sample EPUB that shows the bug.

Using Sigil

Sigil does not start on MAC OS 10.11?

Sigil on the MAC requires MAC OS 10.12 or later.

Is there a tutorial or manual for Sigil?

Yes – in Sigil use **Help=>User Guide** to see an HTML version of the Sigil User Guide which also includes tutorials. Or visit the Sigil home page for a link to the HTML and EPUB versions.

Can I write my book using Sigil?

Yes, you can, but Sigil is primarily an editing program for books that have already been written in one of the many existing word processors. You may prefer to write using your favourite editor, export it to HTML, and then use Sigil to format it for EPUB. But if you really want to write and edit it in Sigil, then you certainly can – the Sigil User Guide was written completely using Sigil.

Are DOC, MOBI, KF8 formats supported?

Sigil is an EPUB editor. It can open EPUB files and import HTML files, but all other formats must be converted to HTML first before importing.

There are a lot of formats out there, and converting from one format to another is not a trivial issue. Since there are already excellent tools for converting from one format to another, Sigil will remain focused on being the best EPUB editor available.

See the Prepare Your File For Sigil chapter for more information.

To convert EPUB to MOBI or KF8 you should use Kindle tools.

How do I use Regex?

See the Regular Expressions chapter in the User Guide for more details and tips on using Regex.

Sigil won't import my file?

Sigil can only import or open valid HTML and EPUB files. It will try to recover from some common problems in files, but if there is a significant issue (missing standard code or files) then it may be unable to open the file.

If your Preferences have Mend Code turned off, try setting it to Mend On Open, as it will correct certain errors. Sometimes Sigil will give you an error message that might identify the problem with the file. In this case you may be able to open your file with a Zip or archive program and edit the files to correct them. But often the cause of the problem is not clear.

If you want to report a problem with a file that cannot load and crashes Sigil, then you should attach the EPUB file to your bug report. But in most cases you will need to correct the file.

Spellcheck does not handle quotes or dashes correctly?

If you install your own hunspell dictionary and it isn't recognizing curly quotes, em-dashes, etc. then you may need additional hyphenation dictionary files. See Spellcheck Dictionary preferences in the User Guide for more details.

Text pasted into Sigil does not look right?

Pasting rich text into Sigil may not work very well. You can try to export (or convert) your text to HTML and then import the new file directly. This may preserve the formatting.

If you want to paste something as plain text, you may need to paste it into Code View to avoid preserving formatting tags.

Formatting and Styles

Why does my book display random question marks?

This may be because your original file has soft hyphens. ADE (the Adobe software that is used to display EPUB files on many systems) does not handle soft hyphens. So just remove the soft hyphens from your file and the question marks will go away. Look for something like or ‑. Basically anything "strange" that exists between those two characters. You have to look in Code View to see the raw HTML, but you can use the Reports feature to show what characters are used in the document.

It may also be because ADE cannot perform font substitution.

In Sigil, if the font you are using does not have the required glyphs to render a character present in the text, glyphs from a different font that *can* render it are used. This is called font substitution. So as long as *some* fonts on your computer can display these characters, so can Sigil.

ADE on the other hand only displays a small subset of Unicode characters by default. The supported characters are listed in the Adobe PDF Reference v1.7, Appendix D, tables D.1 and D.3.

If you want other characters than these to display in ADE, you will have to embed the required fonts.

How do I change the style of my text?

Read the Stylesheet tutorial in the User Guide – you'll need it to make use of many of the answers in this section.

Note that e-readers may support displaying formats or styles differently and how text is display in one reader, or in Sigil, may be slightly different to how it's displayed in another e-reader.

What are stylesheets and CSS?

CSS stands for Cascading Stylesheets. See the Stylesheet tutorial in the User Guide for more details on CSS and why you should use stylesheets to format your text.

Can I embed my own fonts using Sigil?

Yes – see the Fonts tutorial in the User Guide.

What does sgc-1 mean in my document?

sgc indicates a Sigil class that is used to set the formatting style for certain parts of your text. sgc classes are automatically created for certain span tags and “style=” attributes.

How do I add blank lines?

Don’t. It’s much better to use stylesheets to define the amount of space above or below your special sections using a style definition. For instance to leave 1.5 lines of space above a paragraph you can use this style class:

```
p.blankabove { margin-top: 1.5em; }
```

and in your text use the class with your paragraph:

```
<p class="blankabove">My paragraph</p>
```

If you really want to, you can insert a break:

```
<br />
```

or an empty paragraph:

```
<p> </p>
```

For multiple blank lines, it’s best to use a style to define the amount of space you want to leave.

How do I start a new page?

If you use a separate file for your page, then almost all e-readers will start a new page for your text.

How can I use an image for a Chapter heading in the TOC?

Insert your image where you want it – it will look something like this in Code View:

```
<img src="../Images/myimage.png" />
```

Add the heading tag you want to use for the image and give it a title to display in the TOC, e.g.

```
<h1 title="CHAPTER 1"><img src="../Images/myimage.png" /></h1>
```

The TOC will now have an entry called “CHAPTER 1” which links to your image.

You can even use heading tags with no text in your document if you just want a Chapter entry:

```
<h2 title="Chapter Subsection"></h2>
```

But it’s probably easier just to link the TOC to the file if the image is at the top.

How do I add margins to my text?

Add this text with your preferred left and right margins to your CSS definitions:

```
@page {
  margin-left: 5px;
  margin-right: 5px;
}
```

How do I remove the white space between paragraphs?

To remove the white space between all paragraphs, add this to CSS definitions:

```
p {
   margin: 0;
   padding: 0;
}
```

This will affect *only* the paragraph spacing (the *p* means paragraph). To remove *all* the whitespace between the various elements and make the document look more “book like”, use this code:

```
body, div, p, h1, h2, h3, h4 { margin: 0; padding: 0; }
```

How do I add a “drop cap”?

Drop caps are not supported in all readers, but there are several ways to do a drop cap. If your book is formatted just right, it can be done just with CSS.

Add this code to your CSS definitions, which will set the text using the dropcap class in a span tag to use certain values:

```
span.dropcap {
    float: left;
    font-size: 4.7em;
    line-height: 0.8em;
    margin-right: 3pt;
    margin-bottom: -0.1em;
}
```

In Sigil, find the paragraph you want to start with a drop cap, and switch to Code View. Change the text from something like this:

```
<p>Hello</p>
```

to:

```
<p><span class="dropcap">H</span>ello</p>
```

Look at Preview and you should see the drop-cap.

Add around all the letters at the start of paragraphs where you want a drop cap.

You might need to adjust the values for font-size, line-height and margin-bottom to get the effect you want. Sigil's Preview may not show exactly how it'll appear on all e-readers, but it's pretty close.

Why doesn't my font show up in iBooks?

If you embed a font in your EPUB, iBooks requires a non-standard file in the EPUB to tell it to use your font. This cannot be done in Sigil.

Create a file called com.apple.ibooks.display-options.xml with a text editor and put these lines into it:

```
<?xml version="1.0" encoding="UTF-8"?>
<display_options>
<platform name="*">
<option name="specified-fonts">true</option>
</platform>
</display_options>
```

Then use an archive manager to unzip or access the EPUB file. Add the xml file you created to your EPUB in the META-INF directory, then re-zip your EPUB.

Why doesn't my file pass EPUBCHECK due to plist?

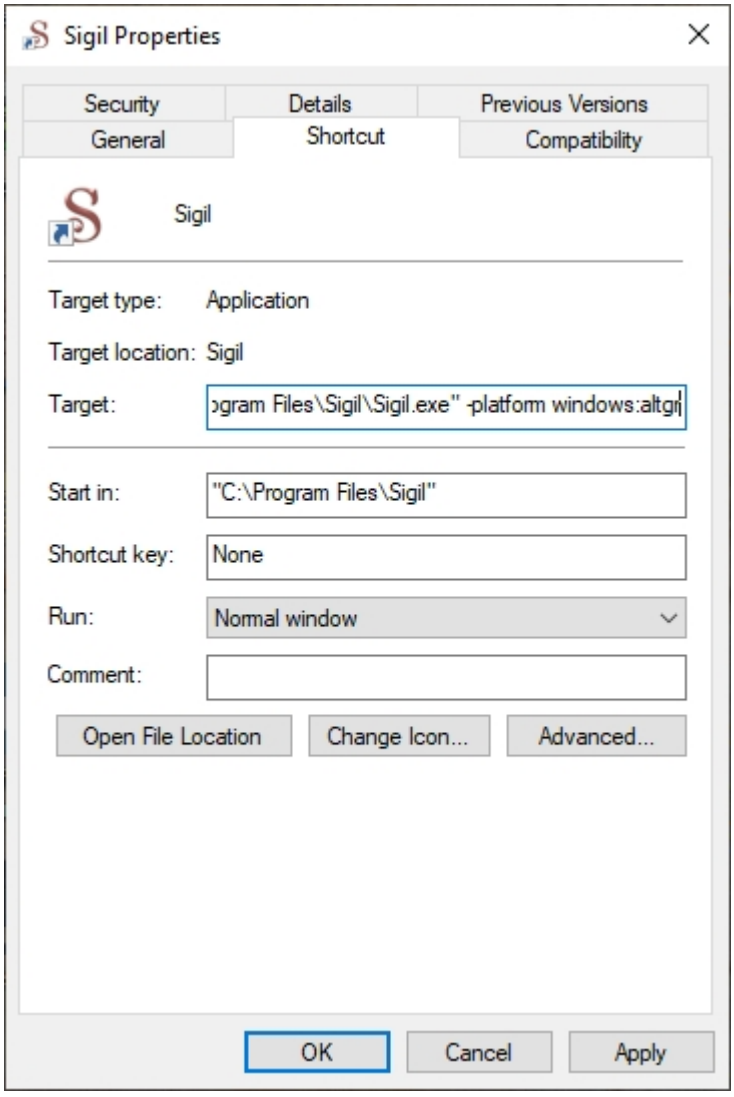
If you open your EPUB in iTunes it will automatically add an extra .plist file to the top of the EPUB. This will cause it to fail EPUBCHECK. Do not open your EPUB in iTunes if you intend to use it elsewhere. You can unzip the EPUB and remove the file, then re-zip the file.

Why doesn't my international keyboard AltGr key work the way I expect?

If you use an international keyboard with the AltGr In order to enable keyboard shortcuts when entering national characters or other symbols, it may help to launch Sigil with an additional parameter:

```
"C:\Program Files\Sigil\Sigil.exe" -platform windows:altgr
```

To make this change persistent, you can add it to the Windows properties window in the shortcut that starts the Sigil application.



If you want the AltGr key to work also when opening an EPUB file directly, you will have to add the additional parameter to your Windows System Registry in a manner similar to the following:

```
"C:\Program Files\Sigil\Sigil.exe" -platform windows:altgr "%1"
```

Contributing to Sigil

— Helping Development —

Sigil is completely free and open-source software, and it is developed and supported entirely by volunteers who love books and want well-edited ebooks to be easily available. Your contributions, bug reports, and testing help others as you help yourself. Feel free to just say hi on the Sigil forum and let others know how useful Sigil is to you.

Translating Sigil

Translating Sigil into other languages is highly encouraged. There is an active community of volunteers who translate Sigil already, and new languages can always be added.

The best way to translate Sigil is to use the web based translation system provided by Transifex. Translations done using Transifex are downloaded just before releasing a new version:

- Sigil translations @ Transifex

Alternatively you can use local Desktop tools for translation. Sigil uses Qt's Linguist tool for creating and editing translations. You can either download Qt and compile the tools manually or use the pre-built packages:

Updating and creating new translation files using the Desktop tools is the same process:

- Open a terminal and change to the src/Sigil/Resource_Files/ts directory. This holds all of the raw translation files.
- Run lupdate ../../* -ts out.ts to either generate a new translation file or to update an existing one with any new strings that have been added.
- Use Linguist to write translations for the strings in a given ts file.

All translations files should have the form sigil_lang.ts. Where lang is a two letter language code. For example the Polish translation will have the filename sigil_pl.ts.

Assisting Users

If you are an experienced Sigil user, EPUB expert, or working your way there, you can help by answering questions or commenting on your experiences – primarily at

the official Sigil Forum at MobileRead. You can also check issues on our github development website and add comments.

Coding

If you are a C++ programmer and have suggestions for improvements or bug fixes you are welcome to submit patches or github pull requests for consideration.

The Sigil User Guide is also free. If you have found it useful, you can say thanks by helping to update and improve the user's guide. We always need help keeping the Sigil User Guide up to date.

Table of Contents

- Title Page
- Copyright
- Introduction
  - What's New
  - About Sigil
  - About EPUB
- Installation
- Features
  - User Interface
  - Preferences
    - Code View
    - Preview
    - Icon Themes
    - Main UI
    - Basics
    - Security
    - Validation
    - Advanced
  - Opening and Saving Files
  - Code View
  - Preview
  - Book Browser
    - Delete
    - Rename
    - ReGex Rename
    - Move
    - Merge
    - Sort
    - Link Stylesheets
    - Link Javascripts
    - Add Semantics
    - Open With
    - Save As
    - Add Blank HTML File/Stylesheet/SVG Image
    - Add Copy
    - Add Existing Files
    - Select All
    - Validate With W3C (CSS stylesheets only)
    - Cover Image (image files only)
    - Font Obfuscation (font files only)
    - Renumber TOC Entries (.ncx files only)
  - Metadata
  - Add Cover
  - Table of Contents

- Validation
- Spellcheck
- Images, Video, Audio
- Special Characters
- Using Checkpoints to Quickly Save/Restore Epub State
- Splitting and Merging
- Find & Replace
  - What to Search For
- Saved Searches
- Clips
  - Context Menu
  - Clip Editor
  - Clip Bar
  - Clips Window
  - Context Menu
  - Keyboard Shortcuts
  - Main Menu
- Clipboard History
- Links and IDs
- Styles
- Indexes
- Reports
- External Editors
- Preferred External Xhtml Editor
- Using the Page Inspector
- Manage Plugins
  - Running Plugins
- Automate Lists
  - Creating and Editing Automate Lists
    - Automate Lists Are Simple Text Files
  - Running an Automate List
- Customize the User Interface
  - Customizing the Toolbars
  - Customizing the Panels
  - Customizing the Icons

Tutorials

- Getting Started With EPUB
- Prepare Your File For Sigil
- Open Your File With Sigil
- Save Your EPUB File
- Add the Author and Title
- Add a Cover Image
- Create Files For Each Chapter
- Identify Your Chapters
- Generate A Table of Contents
- Create Links and Notes
- A Quick Introduction To Code View
- Check For Spelling Mistakes

- Edit With Find & Replace
- Use Stylesheets
- Include Custom Fonts
- Advanced Find & Replace – Regex
- Regular Expression Reference
- Tips

FAQ

- Common Questions
- Where to Get Help
- Using Sigil
- Formatting and Styles

Contributing to Sigil

Landmarks

- Cover
- Text
- Table of Contents