OpenExplainer User Guide

OpenExplainer

0.2.0 (alpha 3)

Jason S. Wrench

Contents

1 Welcome to OpenExplainer 4
1.1 What OpenExplainer is for 4
1.2 What you need 5
1.3 How this guide is organized 5
1.4 How the pages are written 6
1.5 About this version and licence 7
2 Getting started 9
2.1 Installing OpenExplainer 9
2.2 Finding your way around the window 10
2.3 Finding a command in the menus 12
2.4 Searching for a command, and the list of keys 14
2.5 Making your first project 16
2.6 Saving and opening a project 19
2.7 Snapshots: a way back 21
3 Building your book 25
3.1 Chapters and topics 25
3.2 Writing a topic's page 27
3.3 Numbered steps 29
3.4 Writing boxed notes 30
3.5 Marking bold, italic and control names 31
3.6 Reusing the product name everywhere 32
3.7 Leaving notes and tracking progress 34
3.8 Links between topics and See also 35
3.9 Reusing text and building variants 37
3.10 Bringing in a Word manual 40
4 Taking pictures 42
4.1 Capturing a window 42
4.2 Capturing a rectangle of the screen 44
4.3 Capturing click by click 47
4.4 Reviewing your clicks 49
4.5 Showing where the reader clicks 51
4.6 Importing pictures you already have 53
5 Explaining your pictures 55
5.1 Describing a picture for readers who cannot see it 55
5.2 Pointing at part of a picture 57

5.3 Showing only part of a picture 61
5.4 Drawing marks on a picture 65
6 Checking your work 68
6.1 Finding what is still missing 68
6.2 Words to avoid and consistency 70
6.3 Sending the book out for review 73
7 Making your outputs 75
7.1 Previewing before you export 75
7.2 Making a PDF for print 78
7.3 Making a Word file 79
7.4 Making a help site 80
7.5 Exporting Markdown and JSON 81
7.6 Short step-by-step guides 81
7.7 Building the index 84
7.8 Setting the look of your book 85
7.9 Making an e-book 87
7.10 Translating your book 88
8 Using AI help 91
8.1 Knowing what AI help will and will not do 91
8.2 Turning on the built-in model 92
8.3 Reviewing what the AI drafts 94
8.4 Telling the AI how your book should read 96
9 Letting an assistant do the work 97
9.1 Turning on MCP access 97
9.2 Connecting an assistant 99
9.3 Knowing what an assistant can do 100
10 Reference 102
10.1 Using the keyboard 102
10.2 Where your files are kept 104
10.3 When something goes wrong 105
10.4 Glossary 107
Index 110

CHAPTER 1

Welcome to OpenExplainer

In This Chapter

- What OpenExplainer is for
- What you need
- How this guide is organized
- How the pages are written
- About this version and licence

This guide shows you how to use OpenExplainer 0.2.0 (alpha 3) to make how-to books and tutorials from pictures of your screen. OpenExplainer made this guide itself: every page and every export came out of the program, and the pictures of its windows show a copy of the program at work on a made-up book about brewing tea.

Start with Chapter 2, page 9 if you want to be making something in ten minutes. Come back to the other chapters when you need them.

1.1 What OpenExplainer is for

OpenExplainer helps you explain how to use a computer program. You take pictures of the program as you use it and say in plain words what each picture shows. OpenExplainer then lays everything out as a book.

You can print the book, send it as a PDF or Word file, or put it on the web as a help site.

It makes **still pictures only**. There is no video recording. A tutorial of pictures and short numbered steps is quick to scan, quick to search and quick to keep up to date. A screen reader can read it aloud, which a video cannot do.

> ✳ **BRIGHT IDEA**
>
> You do not have to be a writer or a programmer. If you can use a program and say in a sentence what a button does, you can make a good book.

What you can make

The main output is a **book for paper**: chapters, numbered steps, pictures with lettered callouts, boxed notes, a contents list and an index. It comes out as a PDF made for printing, or as a Word file with real styles.

The same book can also become a **help site**, an **e-book** (EPUB), a **Markdown** copy or a **JSON** copy. A help site is a folder of web pages with search, and each page also comes as Markdown for AI assistants. The JSON copy holds the pictures too, and you can bring it back into OpenExplainer.

You can also cut a **short step-by-step guide** out of the book: one task, a few pictures, one or two pages.

You do not have to start from nothing. You can bring in a manual that is already in Word. You can link topics to each other and reuse a warning or a step in many places. You can send the book out for review, send its words to a translator, and go back to an earlier copy if a change goes wrong.

Free, and it works by hand

OpenExplainer is free and open source under the Apache 2.0 licence. Everything in it works by hand. The AI help in Chapter 8, page 91 is optional and runs on your own PC if you turn it on. It never puts anything in your book until you say yes.

1.2 What you need

This topic says what your PC needs, and what you can do without.

Your PC

You need a PC with **Windows 11** and a 64-bit Intel or AMD processor. Other versions of Windows, and Windows on ARM, have not been tested yet. Display scaling above 100 percent has not been tried either.

You need about 60 MB of free disk space for the program. If you use the built-in AI model, add about 2.7 GB for the model file.

What you do not need

You do not need an account, a sign-in or an internet connection. The program uses the internet only when you choose to download the AI model.

You do not need a mouse. Everything can be done from the keyboard, and the program is built to work with a screen reader (see topic 10.1, page 102).

You do need the program you want to explain, installed and working, so that you can take pictures of it.

1.3 How this guide is organized

The guide follows the order in which you work.

Chapter 2, page 9, Getting started, installs the program, tours the window, shows where every command is (the menus, the command search and the list of keys) and makes your first project. It also shows how to go back to an earlier copy of your book.

Chapter 3, page 25, Building your book, covers chapters, topics and writing a page. It also covers links, reusing text and bringing in a Word manual.

Chapter 4, page 42, Taking pictures, covers capturing a window, a rectangle of the screen, and click-by-click capture.

Chapter 5, page 55, Explaining your pictures, covers descriptions, lettered callouts, views and drawing marks.

Chapter 6, page 68, Checking your work, covers the book check and sending the book out for review.

Chapter 7, page 75, Making your outputs, covers the previews and every output. It also covers the index, the look of the book and translation.

Chapter 8, page 91 is about the optional AI help. Chapter 9, page 97 is about letting an AI assistant use OpenExplainer while you are away.

Chapter 10, page 102 is a reference: the keyboard, where your files are kept, what to do when something goes wrong, and a glossary.

The way the pages are written is explained in topic 1.4, page 6.

1.4 How the pages are written

This topic explains how the pages of this guide look, so that you can read them quickly.

Names of buttons, menus and boxes appear in bold, exactly as they appear on screen, for example **Save as...**.

A command is in one of the nine menus under the navy bar. A step names the menu and then the command, for example: On the **File** menu, choose **Save as...**. The three dots are part of the command's name on screen. Keys are written like Ctrl+S, which means: hold down Ctrl and press S.

Steps you follow are numbered. A short bold sentence above each list says what the steps do.

A picture shows the program before you do a step. Where one picture marks several controls, its letters name them, and the steps say which letter to look for.

The pictures in this guide show OpenExplainer at work on a made-up book about brewing tea, with a made-up program called Tea Timer. A few pictures taken earlier show a made-up drawing program called Acme Draw instead. A picture shows the program's own page, so the Windows title bar is left out.

Some steps have a result line in italic. It appears only when you could not otherwise tell that the step worked, or when the next step needs what this one makes.

Boxed notes stand out from the text. A **Caution** warns you about something that can lose your work or cannot be undone. A **Design Tip**, **Power Tip**, **Time Saver** and **Bright Idea** pass on advice.

> **⚠ CAUTION**
> A Caution is the one kind of box you should never skip. Read it before you go on.

OpenExplainer also has a box called **New in this version**. This guide does not use it. Topic 1.5, page 7 lists what is new instead.

1.5 About this version and licence

This topic says which version of OpenExplainer this guide covers, when it was written and under what licence the program is shared.

This guide covers OpenExplainer 0.2.0 (alpha 3). It is dated 12 November 2026. An alpha is a version made so that it can be tried. It is not a finished release.

OpenExplainer is free and open source under the Apache 2.0 licence. Copyright 2026 Jason S. Wrench and OpenExplainer contributors.

What is new in this version

The look. OpenExplainer now has the look that the OpenStage programs share: a navy bar, nine menus that hold every command, panes with an **Add** button and a **...** button, Properties in sections that open and close, a command search and a list of keys (topic 2.2, page 10, topic 2.3, page 12 and topic 2.4, page 14). Every command has the name it had before, and every key works as before.

Windows that close. Every window that opens over the main window has a **Close** button in its corner, which does what Escape does, and its row of buttons stays in sight at the bottom.

What is new since version 0.1.1

Snapshots keep dated copies of your book, so you can go back (topic 2.7, page 21).

Links and **See also** lists take a reader from one topic to another (topic 3.8, page 35).

Snippets, **conditions** and **targets** let you reuse text and build variants of one book (topic 3.9, page 37).

Import from Word brings a manual you already have into your book (topic 3.10, page 40).

A **Word review copy** goes out to reviewers, and their comments come back (topic 6.3, page 73).

Look of the book sets the fonts, colours, title page and paper (topic 7.8, page 85).

The **e-book** (EPUB) is a new output (topic 7.9, page 87).

Translation sends the words of your book out and brings the translation back as a copy (topic 7.10, page 88).

What was tried for this edition

The pages, the pictures and the names of buttons and menus were checked on 12 November 2026 against the installed OpenExplainer 0.2.0 (alpha 2). OpenExplainer 0.2.0 (alpha 3), published a few minutes after it, is by its release notes the same program, built from the same

commit, with a signature added. It was not downloaded or run for this guide. A separate copy of alpha 2 was run with its own browser profile, on a made-up book. Every menu and every command in it was opened, and so were the command search, the list of keys and the windows that the commands open. The copy was driven through the browser engine's debugging port, so no real mouse or keyboard was used. The Capture window was opened, but nothing was captured, and the list of open windows in its pictures was replaced with made-up windows, because the real list would show whatever was open on the PC.

Not tried: the windows that Windows itself shows for choosing a file or a folder (Open project, Save as, Import pictures, Import from Word, Bring in review comments, Import a translation, Replace the picture file, Choose a logo picture and Choose the folder) and what comes after them. Also not tried: any capture of a window, a rectangle or clicks, so the small Stop and Cancel windows were not seen again; the AI help (no model was downloaded and nothing was drafted); connecting an assistant over MCP (this guide was edited through MCP, but the steps for connecting one were not followed); the installer; a screen reader; Windows on ARM; and display scaling above 100 percent.

The pictures of the selection screen (topic 4.2, page 44), the Stop window (topic 4.3, page 47), the Review your clicks window (topic 4.4, page 49) and the Bring in review comments window (topic 6.3, page 73) were taken on 1 November 2026, before the new look. A topic that rests on something that was not tried has the status **Needs recheck**.

CHAPTER 2

Getting started

In This Chapter

- Installing OpenExplainer
- Finding your way around the window
- Finding a command in the menus
- Searching for a command, and the list of keys
- Making your first project
- Saving and opening a project
- Snapshots: a way back

In this chapter you install OpenExplainer, take a short tour of its window and make your first project. It takes about ten minutes. It is the only chapter you need before you start work.

2.1 Installing OpenExplainer

OpenExplainer comes as one setup file with a name like **OpenExplainer_0.2.0_x64-setup.exe**. It installs for you alone, so Windows does not ask for an administrator's password.

A second small file, with the same name and the ending **.sha256.txt**, holds the setup file's SHA-256, so that you can check that your download is whole.

> **⚠ CAUTION**
> The first versions of OpenExplainer were not signed, and Windows asks you to confirm before it runs an unsigned setup file. Alpha 3 is signed by Jason Wrench, by its release notes, but Windows SmartScreen may still warn on a first download, because the signer is new to it. Run only a setup file that you got from a place you trust.

To install OpenExplainer:

1. Double-click the setup file.
2. If Windows shows a blue screen that says Windows protected your PC, click **More info**.
3. Click **Run anyway**.
4. Wait while the setup copies the program.

   Result: It takes a few seconds. When it finishes, the program is in your own AppData folder, and an OpenExplainer icon is in the Start menu and on the desktop.
5. Open OpenExplainer from the Start menu or from the desktop icon.

   Result: The OpenExplainer window opens with an empty project.

Updating and removing

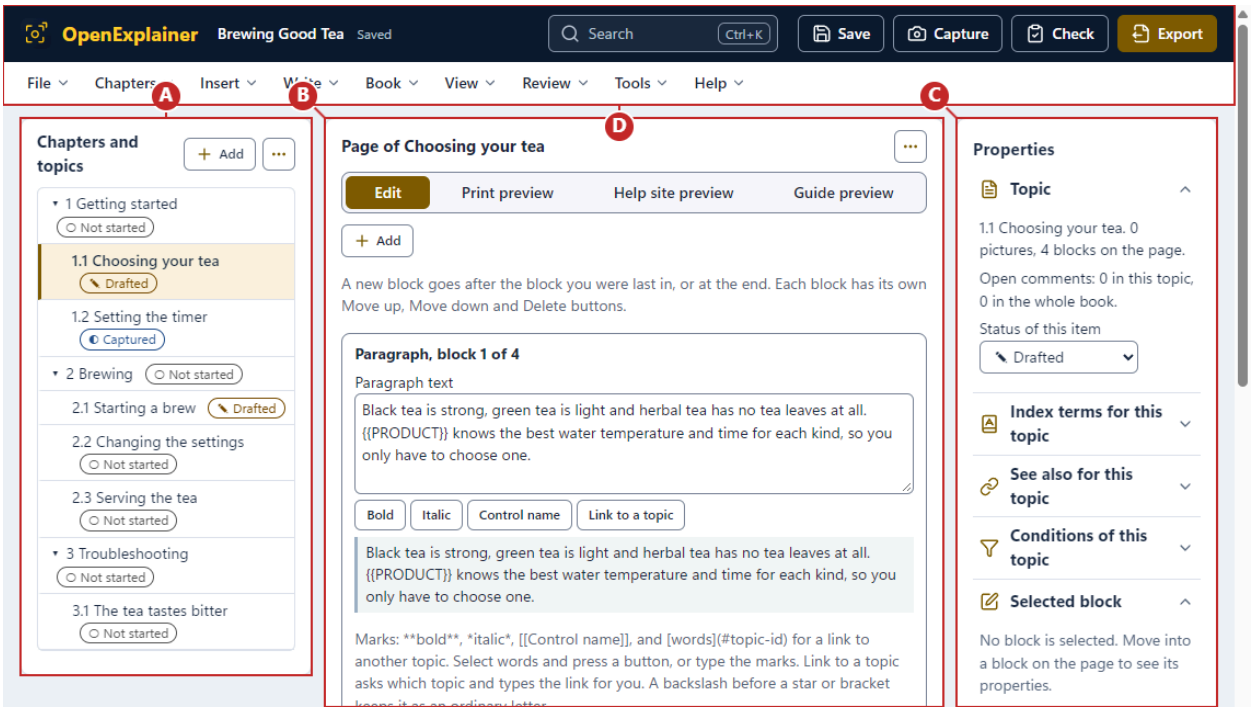
To update, run the newer setup file. It replaces the program and leaves your projects alone. It also keeps your settings and a downloaded AI model. Your projects are files that you keep wherever you saved them.

To remove OpenExplainer, open Windows Settings, choose Apps, then Installed apps, find OpenExplainer and choose Uninstall. If the uninstaller offers to delete the program's data, leave that unticked. Tick it only if you want to remove the downloaded AI model as well, which is about 2.7 GB.

The installer's own windows were not run again for this edition (topic 1.5, page 7). The steps above are as the release notes describe them.

2.2 Finding your way around the window

The OpenExplainer window has a navy bar across the top, a row of menus under it and three panes below. This topic names each part, so that you know where to look.



A. Chapters and topics: Your book as a tree, with an Add button and a ... button.
B. Page: The page of the chosen topic, with tabs for editing and for previews.
C. Properties: The topic's status, its index terms and the block you are working in.
D. The navy bar and the menus: The name of the program and the project, whether it is saved, the Search box, four buttons and the nine menus that hold every command.

The **navy bar** has the name of the program and, beside it, the **project's name**. After the name come the words *Saved*, *Unsaved changes* or *Not saved yet*, so that you can tell in words, and not by colour, whether your work is kept. Then come a **Search** box, which finds any command

(Ctrl+K), and four buttons: **Save**, **Capture**, **Check** and **Export**. Check is short for Check the book.

Under the bar is a row of nine **menus**: **File**, **Chapters**, **Insert**, **Write**, **Book**, **View**, **Review**, **Tools** and **Help**. Between them they hold every command (topic 2.3, page 12).

> ✳ **BRIGHT IDEA**
>
> You do not have to learn all of these now. The three panes, the Add buttons and the Save button are enough to start. The rest are explained when you need them, in the chapters that follow.

The three panes

The three panes sit side by side below the menus. Press F6 to move to the next pane, and Shift+F6 to go back.

Chapters and topics is on the left. It lists your book as a tree: chapters, and the topics inside them. Its **Add** button adds a chapter or a topic, and its **...** button holds the rest: rename, move, indent, outdent, find and replace, and delete. A topic is one subject that gets its own page, such as Saving your work.

Page is in the middle. It shows the page of the topic you chose. Its tabs switch between **Edit**, **Print preview**, **Help site preview** and **Guide preview**, so you can see your book as it will look. In Edit, the **Add** button under the tabs puts a paragraph, a heading, a picture, a numbered procedure, a boxed note or a snippet on the page, and the **...** button above it holds the unresolved comments and the AI drafts for a page. Under the page is the list of the topic's pictures, with its own **Add** button (import and capture) and **...** button.

Properties is on the right. It is a column of sections that open and close: **Topic**, **Index terms for this topic**, **See also for this topic**, **Conditions of this topic** and **Selected block**. Click a section's heading to open or close it. The Topic and Selected block sections start open.

The line at the bottom

The line at the bottom of the window says where the project is saved, with the words *Saved file:* and the file's name, or *Not saved to a file yet*. At its other end it reminds you that F6 moves between the panes and Ctrl+K finds a command.

Windows that open over the window

A window that opens over the main window, such as Capture or History, has a **Close** button in its corner. It does the same as pressing Escape: nothing you changed in the window is kept. A window is never wider or taller than the program's window. If it holds more than fits, scroll inside it.

2.3 Finding a command in the menus

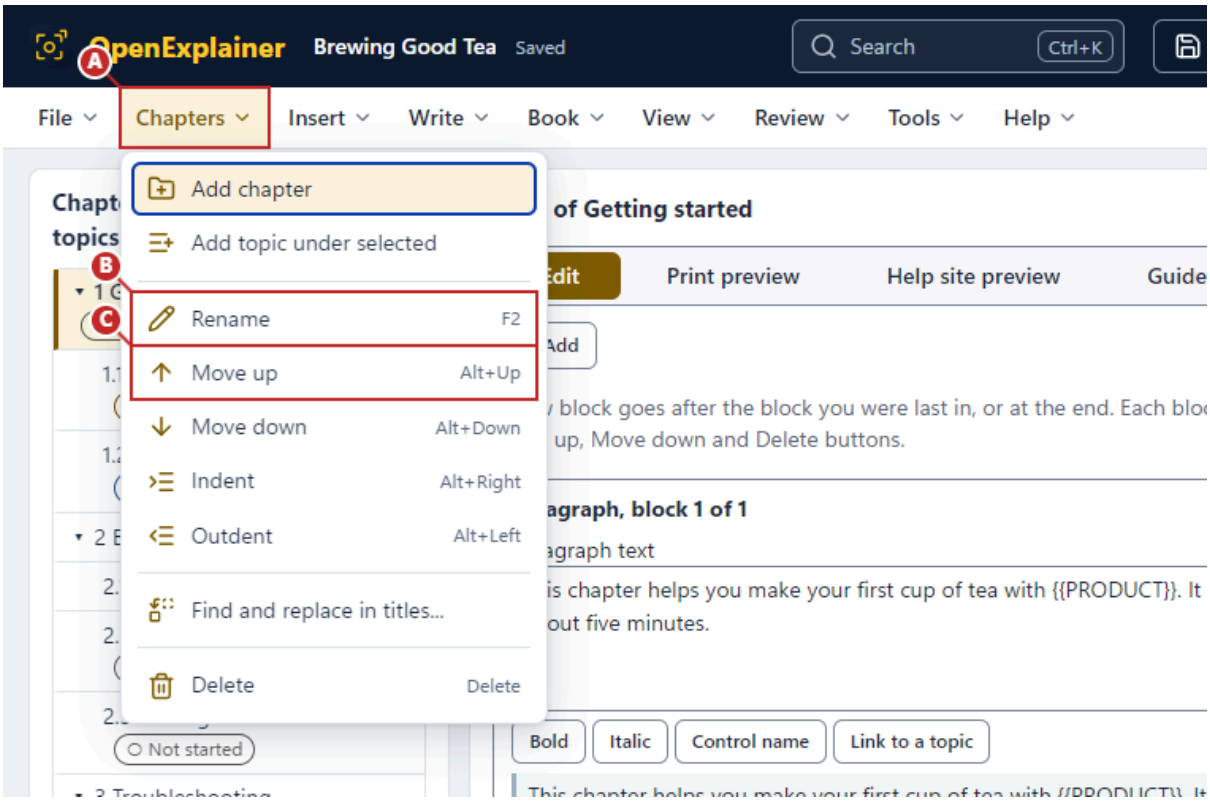
Every command is in one of nine menus, in the row under the navy bar. A menu shows each command with its icon and, where it has one, its keys.

To use a command from a menu:

1. Click the name of a menu, for example **Chapters** (A in the picture).

   Result: The menu opens under its name.

2. Click a command, for example **Rename** (B).



A. Chapters: The name of the menu that is open.

B. Rename: The key for a command, here F2, is shown at the right of its row.

C. Move up: Moves the chosen chapter or topic above the one before it.

Result: Some commands open a window. Others do their work at once.

The key for a command is at the right of its row. In the picture, F2 is the key for **Rename**, and Alt+Up is the key for **Move up** (C).

What is in each menu

File holds **New project...**, **Open project...**, **Save**, **Save as...**, the four commands that bring something in (**Import an export...**, **Import from Word...**, **Bring in review comments...** and **Import a translation...**), **Export...** and **History...**.

Chapters holds **Add chapter**, **Add topic under selected**, **Rename**, **Move up**, **Move down**, **Indent**, **Outdent**, **Find and replace in titles...** and **Delete**.

Insert holds **Import pictures...**, **Capture...** and the six kinds of block for the page: **Paragraph**, **Heading**, **Picture**, **Numbered procedure**, **Callout box** and **Snippet**.

Write holds **Drafts...**, the five ways to draft (**Draft callout text for this topic**, **Draft a procedure from the clicks**, **Draft an opener for this chapter**, **Draft a chapter summary here** and **Suggest boxed notes for this topic**) and **Writing style...**.

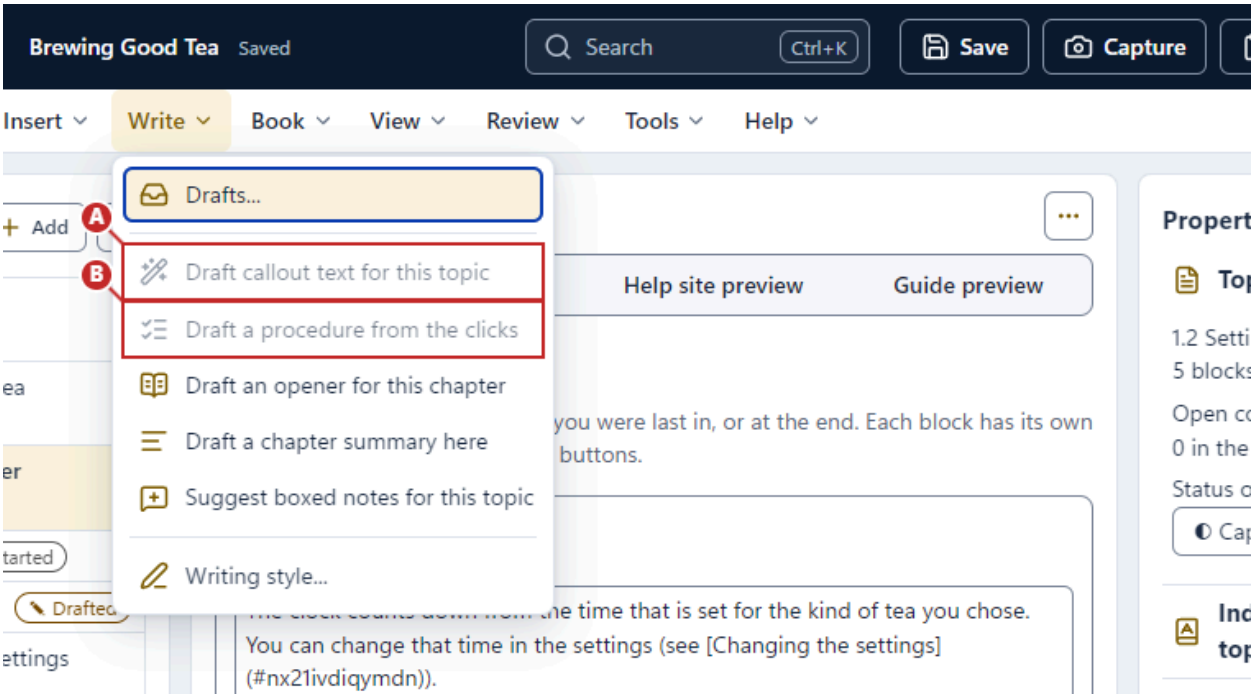
Book holds **Project variables...**, **Look of the book...**, **Reuse...**, **Index...**, **Guides...**, **Click circle settings...** and **Zoom settings...**.

View holds **Edit**, **Print preview**, **Help site preview** and **Guide preview**. A tick shows which one is on.

Review holds **Check the book...**, **Unresolved comments...** and **Drafts...**. The menu shows how many comments are open, as in Unresolved comments (1).

Tools holds **AI help...** and **MCP access...**. **Help** holds **Search for a command** and **Keyboard keys**.

A command that cannot be used now stays in its menu, dimmed. For example, **Draft callout text for this topic** is dimmed until the topic has a picture with callouts, and **Draft a procedure from the clicks** is dimmed until it has a picture taken click by click.



A. Draft callout text for this topic: Dimmed, because the chosen topic has no picture with callouts yet.
B. Draft a procedure from the clicks: Dimmed, because the chosen topic has no picture taken click by click.

Using the menus from the keyboard

Press F10, or press Alt and let go of it on its own, to go to the menus. Left and Right move to the menu beside it, and Down opens a menu. Up and Down move in a menu, Home and End go to its first and last command, and a letter jumps to the next command that starts with it. Enter chooses a command. Escape closes the menu and puts the keyboard back where it was.

The Add and ... buttons

Each pane has its own menus for the commands that belong to it: **Add** and **...** for the chapters and topics, **Add** and **...** for the page, and **Add** and **...** for the pictures. They hold the same commands as the menus under the navy bar, so you can use them without leaving the pane.

> **TIME SAVER**
>
> If you know what a command is called, Ctrl+K is quicker than looking through the menus. Topic 2.4, page 14 shows how.

2.4 Searching for a command, and the list of keys

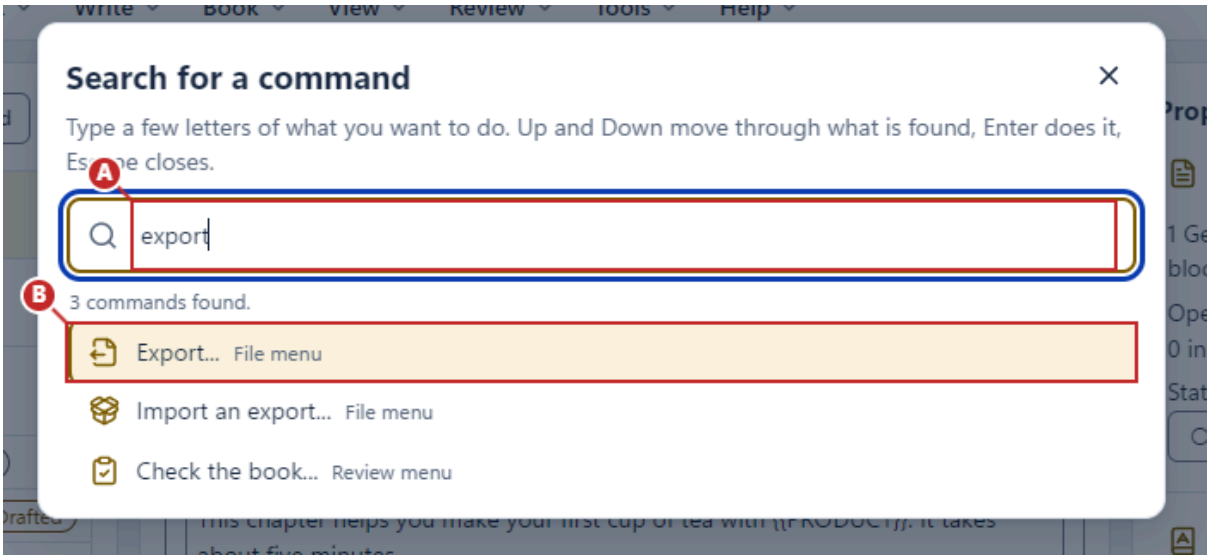
Ctrl+K finds any command by its name, by the menu it is in, or by what it is for. It is the quickest way to a command when you do not remember which menu holds it.

To search for a command:

1. Press Ctrl+K, or click the **Search** box in the navy bar.

   Result: A window called *Search for a command* opens, with the keyboard in its box.

2. Type a few letters of the command, for example export (A in the picture).



A. Search box: Type a few letters of what you want to do.

B. Export...: Each command found shows its menu and, if it has one, its keys. Enter does the command.

Result: The commands found are listed, each with its menu and its keys (B). Typing export finds Export, Import an export and Check the book, because the words that say what Check the book is for mention export.

3. Press Down to move through the list, and Enter to do the command.

Result: The window closes and the command runs. Press Escape instead to close the window and do nothing.

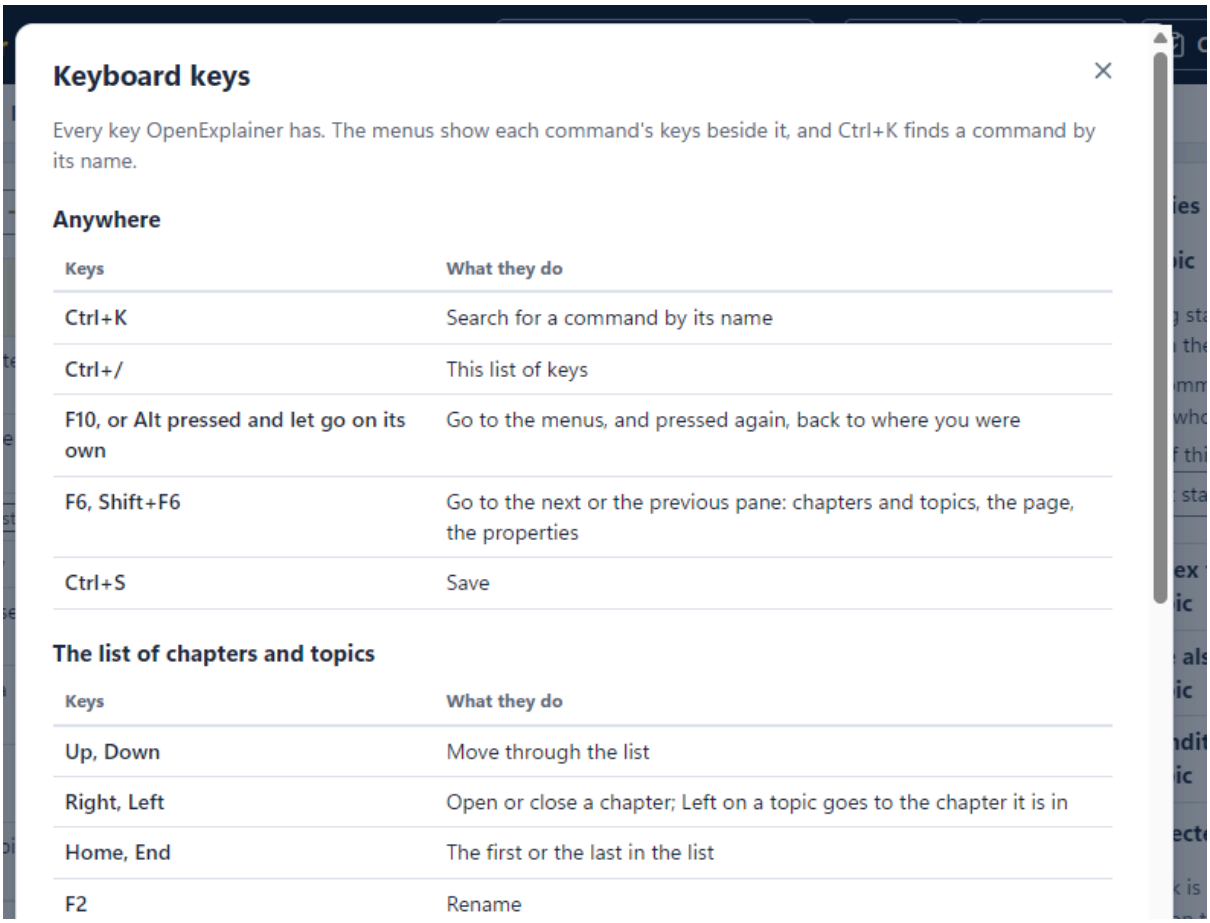
With nothing typed, the list holds every command, in the order of the menus. Page Up and Page Down move through it a page at a time.

The list of keys

OpenExplainer has a window that lists every key it knows, in tables. The menus also show each command's keys beside it.

To see every key:

1. Press Ctrl+/, or choose **Help**, and then **Keyboard keys**.



   Result: A window called *Keyboard keys* opens. Its tables are headed Anywhere, The list of chapters and topics, Menus and the Add and ... buttons, The page, Search for a command and Dialogs.

2. Press Escape, or click **Close**, to leave the window.

Topic 10.1, page 102 describes the keys in words.

2.5 Making your first project

A project is a book. It has chapters, and each chapter has topics. In this exercise you make a project with one chapter and one topic, and you add a paragraph to its page. You use the same few commands every time you make a book.

When you start a book, you choose what to start from in the list called *Start from*. **Software user guide** has a few chapters already made: overview, installation, using and troubleshooting. **OpenStage how-to book** has a few starter chapters. **Empty project** has nothing in it. This exercise uses an empty project.

The picture at step 5 marks the button you use to add things (A), and the two commands in its menu: Add chapter is B, and Add topic under selected is C. The picture at step 9 marks the button that adds a block to the page (A) and Paragraph (B).

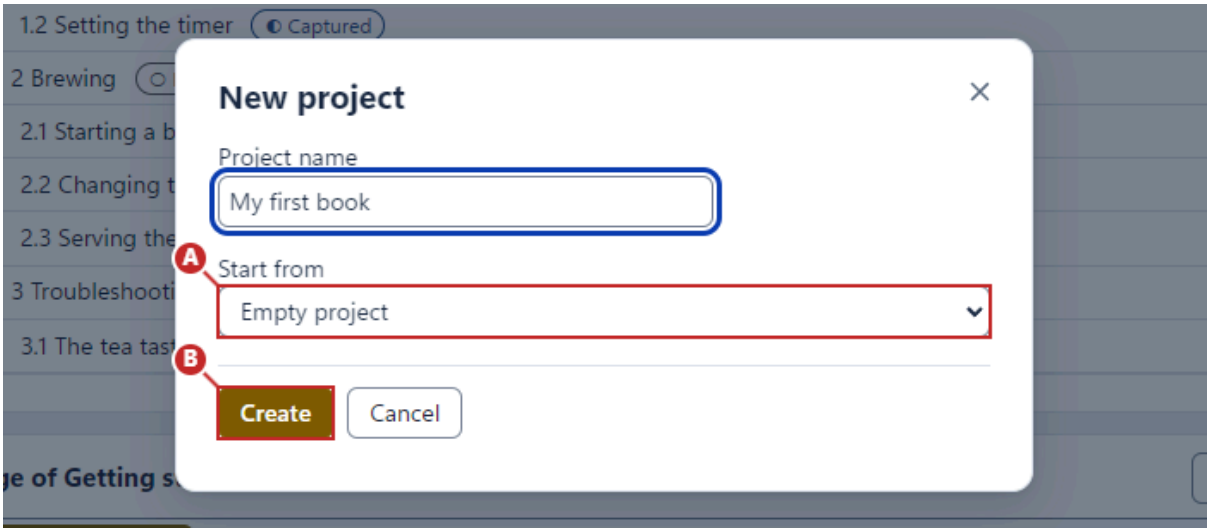
To start an empty project and add a chapter, a topic and a paragraph:

1. On the **File** menu, choose **New project...**.

   Result: A window called *New project* opens.

2. Type the project's name.

3. In the list called *Start from*, select **Empty project**.



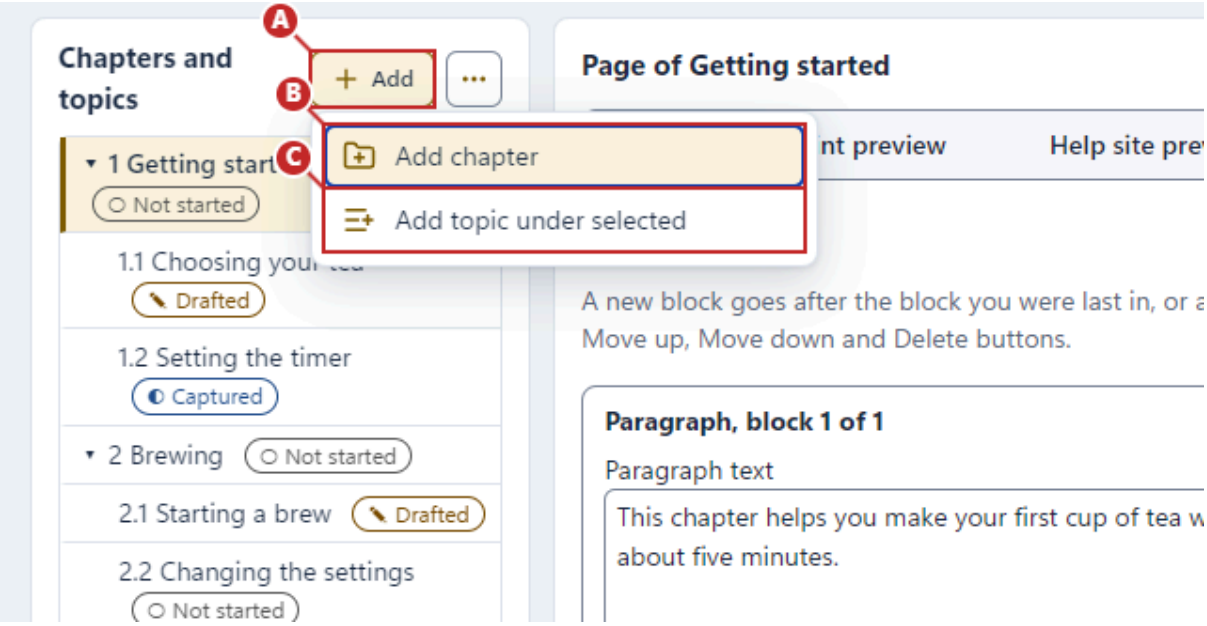
A. Start from: What the new book begins with: a software user guide, an OpenStage how-to book or an empty project.

B. Create: Makes the project.

4. Click **Create**.

   Result: The project opens. The navy bar says *Unsaved changes* beside its name, and the list of chapters and topics is empty.

5. Above the list of chapters and topics, click **Add** (A), and choose **Add chapter** (B).



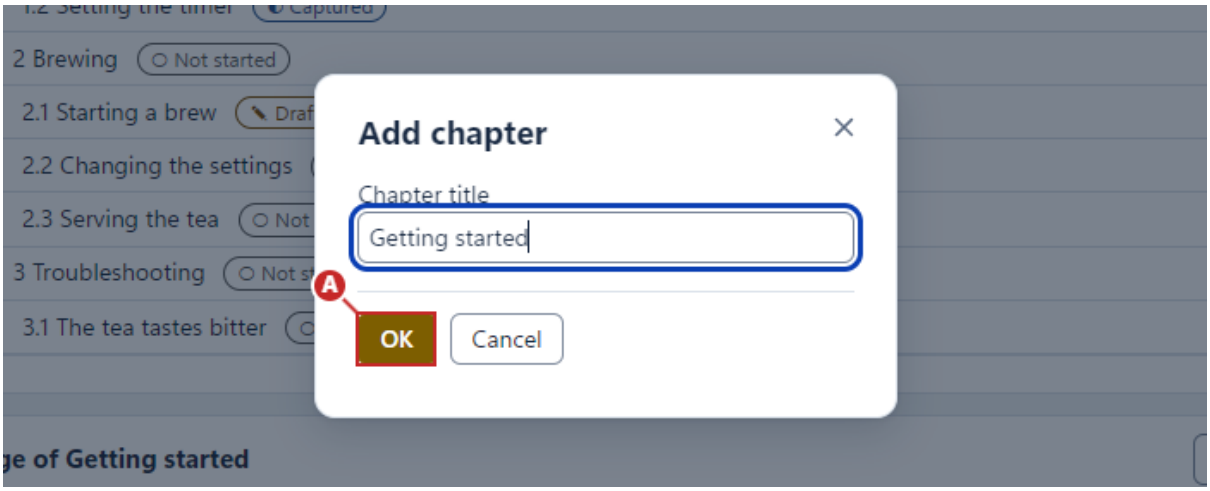
A. Add: Opens a menu that adds a chapter or a topic.

B. Add chapter: Adds a chapter at the end of the book.

C. Add topic under selected: Adds a topic under the chapter or topic that is chosen.

Result: A small window asks for the chapter's title.

6. Type a title, such as Getting started, and click **OK**.

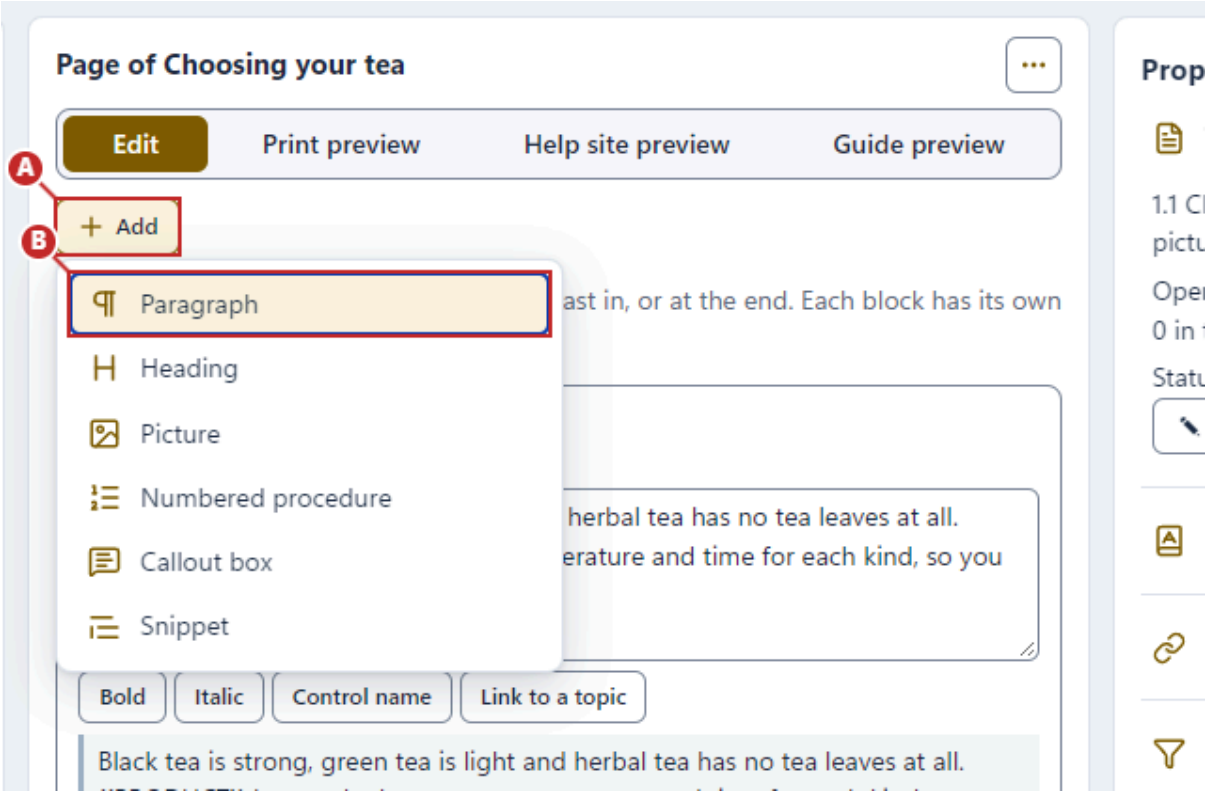


A. OK: Adds the chapter.

Result: The chapter is in the list, and its page is open.

7. Click **Add** again, and choose **Add topic under selected** (C in the picture at step 5).
8. Type a title, such as Saving your work, and click **OK**.

9. Under the tabs of the page, click **Add** (A), and choose **Paragraph** (B).



A. Add: Opens the menu of blocks you can put on the page.

B. Paragraph: Puts a block of text on the page.

Result: A box for a paragraph appears on the page, with the keyboard in it, ready for you to type in.

> **TIME SAVER**
> You can use the keyboard for every one of these steps. Press F10 to go to the menus, or Tab to move from control to control, and Enter to press the one you are on.

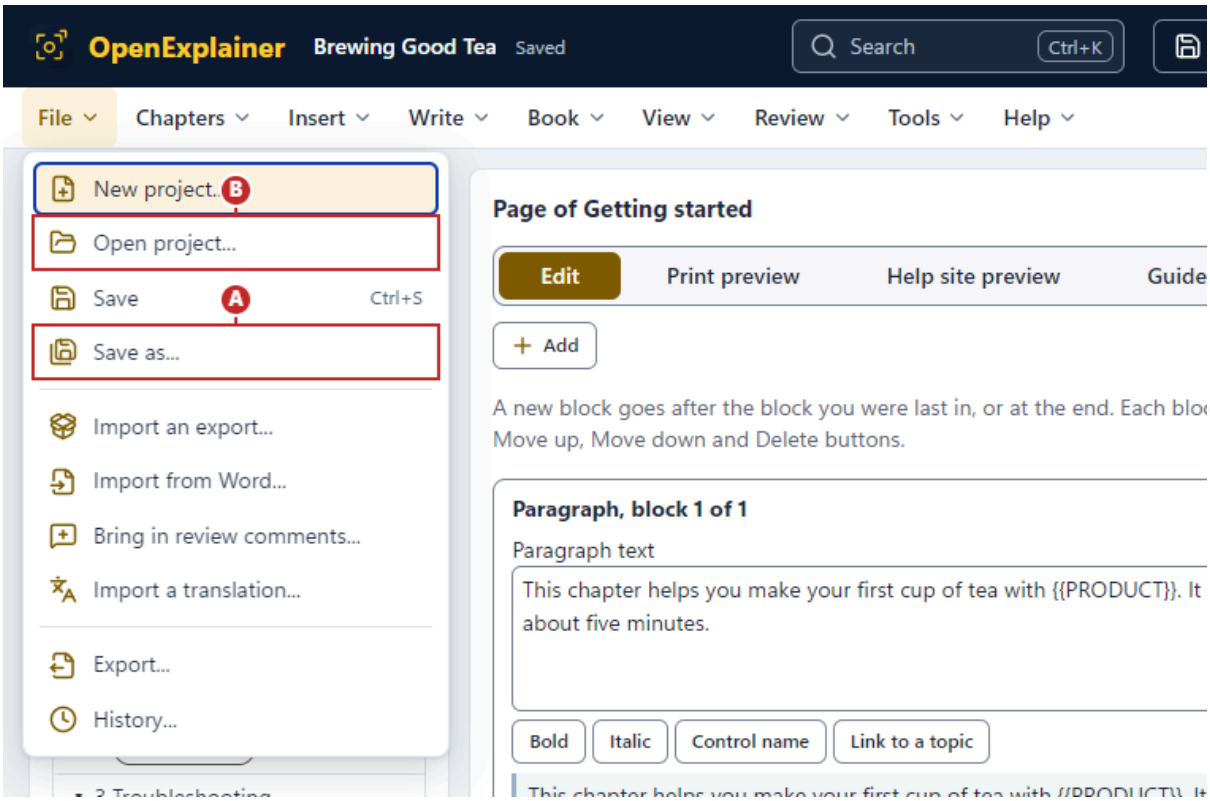
Now type a sentence in the box. Under the box, a preview shows how the sentence will look in the book. Words that you put between double square brackets, such as **Save**, are the **names of buttons and menus**. They are shown in bold in the book. Chapter 3, page 25 explains this in full.

2.6 Saving and opening a project

A project is saved as **one file** that holds all the text, the pictures and the settings. You can copy it, e-mail it or keep it on a USB drive. The file's name ends in **.openexplainer**.

To save a project for the first time:

1. On the **File** menu, choose **Save as...** (A in the picture).



A. Save as...: Saves the project to a new file.

B. Open project...: Opens a project that you saved.

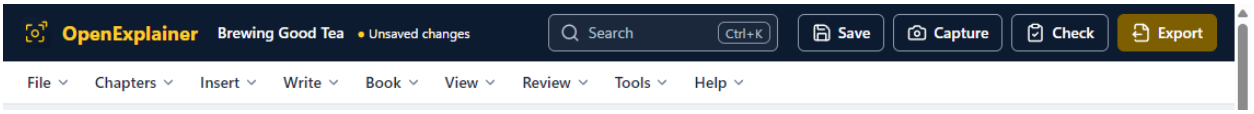
   Result: Windows shows its own Save window.

2. In Windows' Save window, choose a folder and type a name for the project.
3. Click **Save** in that window.

   Result: The project is saved. Beside the project's name, the navy bar now says *Saved*, and the line at the bottom of the window shows where the file is.

After that, **Save** in the navy bar keeps your work up to date without asking anything. You can also choose **Save** on the **File** menu, or press Ctrl+S.

The navy bar says in words whether the project is saved. In the picture, a change has been made, so it says *Unsaved changes*.



> **⚠ CAUTION**
>
> OpenExplainer does not save by itself. Save before you close the program or start a new project, or the changes since your last save are lost.

The title of the program's window names the project. You see it in the title bar, on the taskbar and in the Alt+Tab list. It says *(unsaved changes)* in words when something is not saved, as the navy bar does. So a screen reader can tell you which project is open.

Opening a project again

To open a project you saved:

1. On the **File** menu, choose **Open project...** (B in the first picture).

   Result: Windows shows its own Open window. If you have changes that are not saved, OpenExplainer first asks whether you want to continue without saving them.

2. Choose the file you saved, and click **Open** in Windows' Open window.

   Result: Your book appears, with every chapter, topic and picture as you left it.

Opening a project from Windows

You can also open a project without starting OpenExplainer first. In File Explorer, double-click the project's file, or right-click it and choose **Open with**. The installer sets this up for you alone, so it needs no administrator rights.

Windows opens the project in OpenExplainer only if it is not already set to open that kind of file with another program. If it is, right-click the file, choose **Open with**, and choose OpenExplainer.

If OpenExplainer is already open, the project opens in a second copy of the program. What you were working on in the first copy is not touched.

Before you do something big, such as moving chapters or bringing in a Word manual, take a **snapshot** (topic 2.7, page 21). If the change goes wrong, you can go back.

2.7 Snapshots: a way back

A **snapshot** is a dated copy of your whole book, kept inside the project file. If a change goes wrong, you can put the whole book, or only one topic, back the way it was. Take one before you do something big: moving chapters, bringing in a Word manual, or sending the book out for review.

Taking a snapshot

To take a snapshot:

1. On the **File** menu, choose **History...**.

   Result: A window called *History: snapshots of the book* opens, with a list of the snapshots, oldest first.

2. Click **Take a snapshot now**, type a name such as *Before moving chapters*, and click OK.

   Result: The snapshot is added to the list, and the window says so. If you leave the name empty, it is named from the date.

The book keeps up to **20** snapshots. At 20 the **Take a snapshot now** button is off, and you delete one first. Snapshots are saved inside the project file, so they travel with it. Each one adds the words of the book to the file; the pictures are not copied.

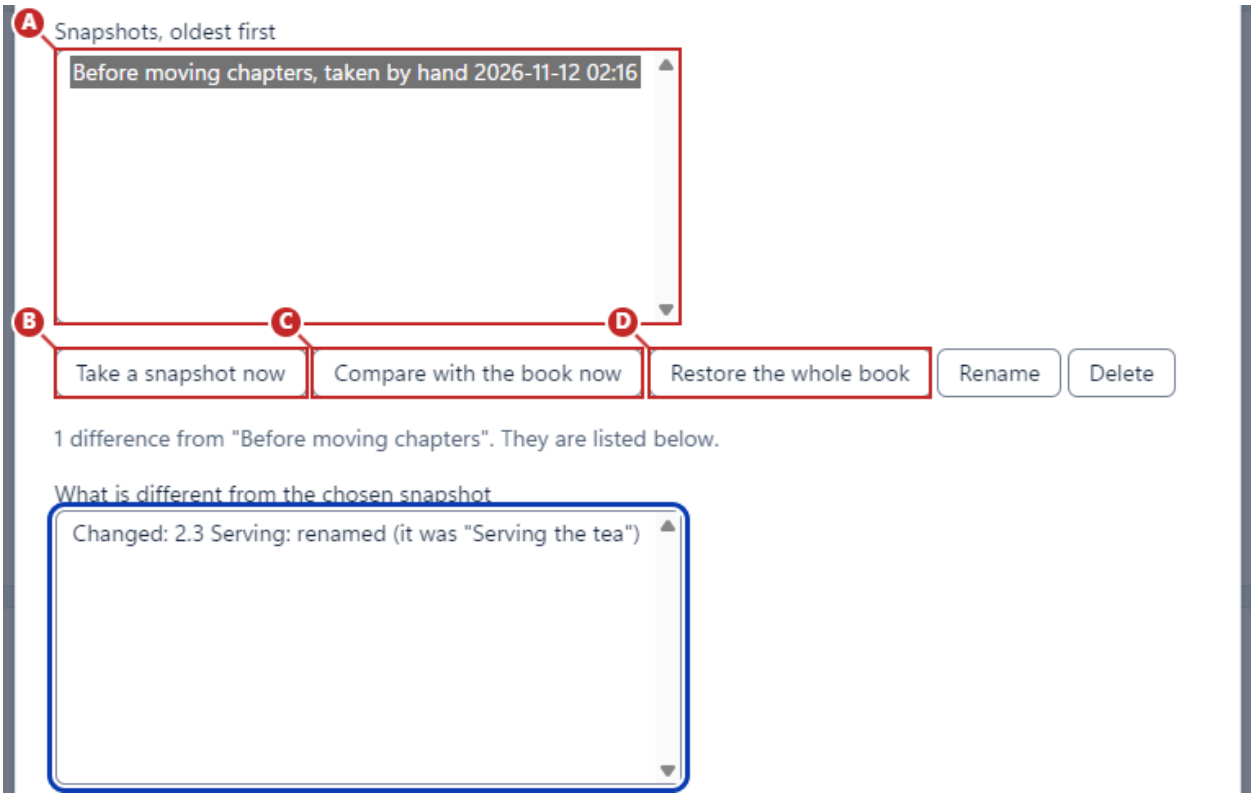
> ✎ **DESIGN TIP**
>
> A snapshot made by an older version is upgraded when it is read, like any project, so snapshots never go out of date. The JSON export holds the book as it is now and does not carry snapshots.

Seeing what has changed

To see what is different from a snapshot:

1. Choose a snapshot in the list, and click **Compare with the book now**.

   Result: A second list says in words what is different from that snapshot: which topics were added, removed or changed, and which settings changed.



A. Snapshots: The snapshots, oldest first.
B. Take a snapshot now: Keeps the book as it is now.
C. Compare with the book now: Says in words what is different from the chosen snapshot.
D. Restore the whole book: Puts the whole book back as the snapshot had it.

Going back

> ⚠ **CAUTION**
> Restoring replaces what you have done since. Before it does, OpenExplainer keeps the book as it is now as a new snapshot called *Before restoring* and the name. If you restore the wrong one, restore that snapshot to undo it.

To put the whole book back:

1. Choose the snapshot, and click **Restore the whole book**.
2. Read the question, which says what will be replaced, and click **Restore the book**.

To put back one topic:

1. Compare the snapshot with the book, as above.
2. Choose the topic in the second list, and click **Restore the chosen topic**.

   Result: That topic comes back as the snapshot had it. A topic you deleted since is put back where it was. Other topics are not touched.

Rename gives a snapshot another name, and **Delete** removes one after asking. Neither changes the book. Press Escape or click **Close** to leave the window: the keyboard goes back to where it was.

Everything in the window is done by keyboard. The two lists are list boxes with labels, and each result is said in the line under the buttons, so a screen reader reads it.

CHAPTER 3

Building your book

In This Chapter

- Chapters and topics
- Writing a topic's page
- Numbered steps
- Writing boxed notes
- Marking bold, italic and control names
- Reusing the product name everywhere
- Leaving notes and tracking progress
- Links between topics and See also
- Reusing text and building variants
- Bringing in a Word manual

This chapter is about the writing side of OpenExplainer: how a book is arranged, and the blocks you use to write a page. Pictures come in Chapter 4, page 42. If you made the practice project in Chapter 2, page 9, you already know most of the commands.

The parts of a book

Project. One file, with the ending .openexplainer, that holds the whole book: its text, its pictures and its settings.

Chapter. A top-level part of the book. A chapter can hold topics.

Topic. A part of a chapter, with a page of its own. A topic can hold more topics.

Page. What a topic says. A page is built from blocks: paragraphs, headings, pictures, numbered procedures and boxed notes. A picture can carry lettered callouts (Chapter 5, page 55).

3.1 Chapters and topics

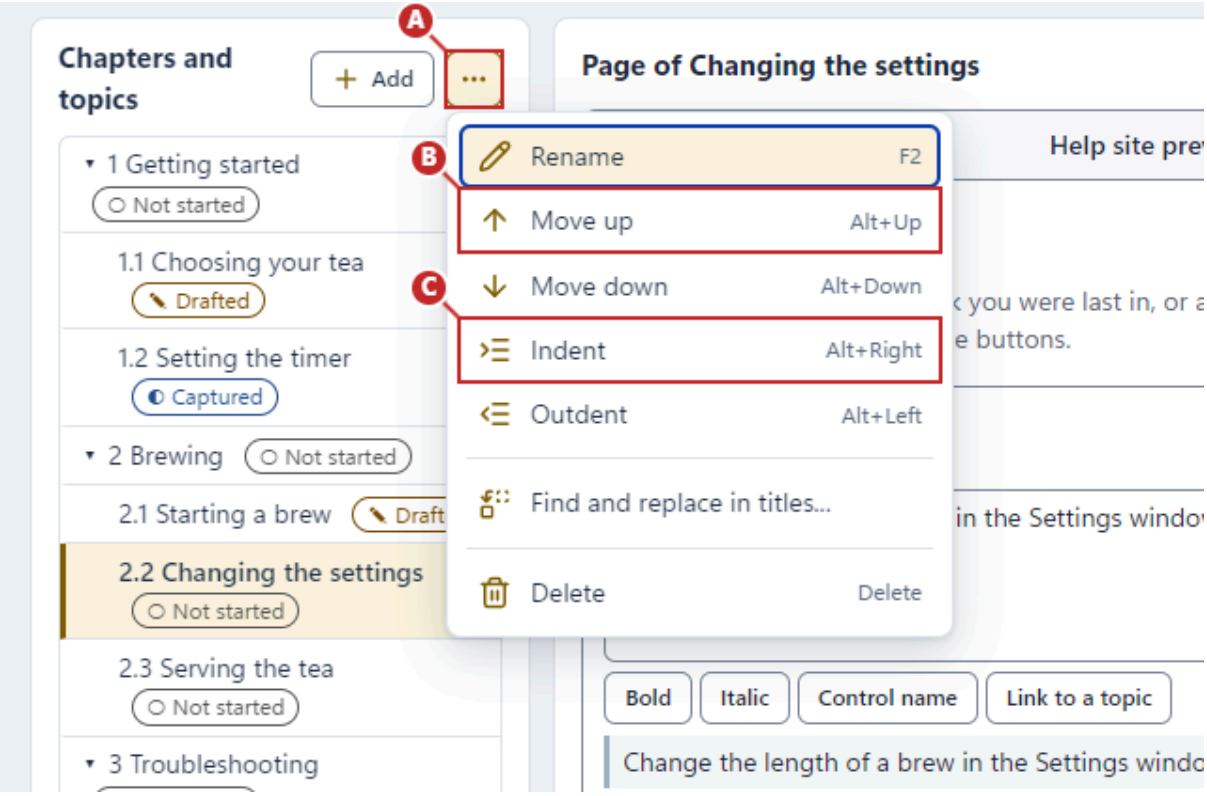
A book is a **tree**. The top level holds **chapters**. Inside a chapter are **topics**, and a topic can hold more topics. Every one of them has a page of its own, a status and a place in the list on the left.

You never type numbers. OpenExplainer numbers the list for you (1, 1.1, 1.1.1 and so on). It numbers the list again whenever you move something, so the numbers are always right.

To move a topic:

1. Click a topic in the list on the left.

2. Click the **...** button at the top of the list (A), and choose **Move up** (B in the picture).



A. The ... button: Holds the commands for the chosen chapter or topic.

B. Move up: Moves the topic above the one before it.

C. Indent: Puts the topic inside the item above it.

Result: The topic moves above the one before it.

3. Click the **...** button again, and choose **Indent** (C in the picture above) to put the topic inside the item above it.

Result: The topic becomes a sub-topic, and its number gets another part.

Move down moves the topic below the next one, and **Outdent** brings it back out. All of these are in the **Chapters** menu too.

The other commands

Add chapter and **Add topic under selected** make new items at the end. They are in the menu of the **Add** button above the list, and in the **Chapters** menu. **Rename** (or F2) changes a title. **Delete** removes the selected item and everything inside it, and asks you first. **Find and replace in titles...** opens a small form above the list, headed *Find and replace in chapter and topic titles*, for changing a word in every title at once. Escape puts the form away again.

> **⚠ CAUTION**
> Deleting a chapter or topic deletes every page and picture inside it. Look at what the confirmation says before you agree.

Working from the keyboard

In the list, Up and Down move between items, and Home and End go to the first and the last. Right opens a chapter, and Left closes it. Left on a topic goes to the chapter it is in. F2 renames. Delete removes. Alt with Up or Down reorders, and Alt with Right or Left indents or outdents. Topic 2.4, page 14 shows the list of every key.

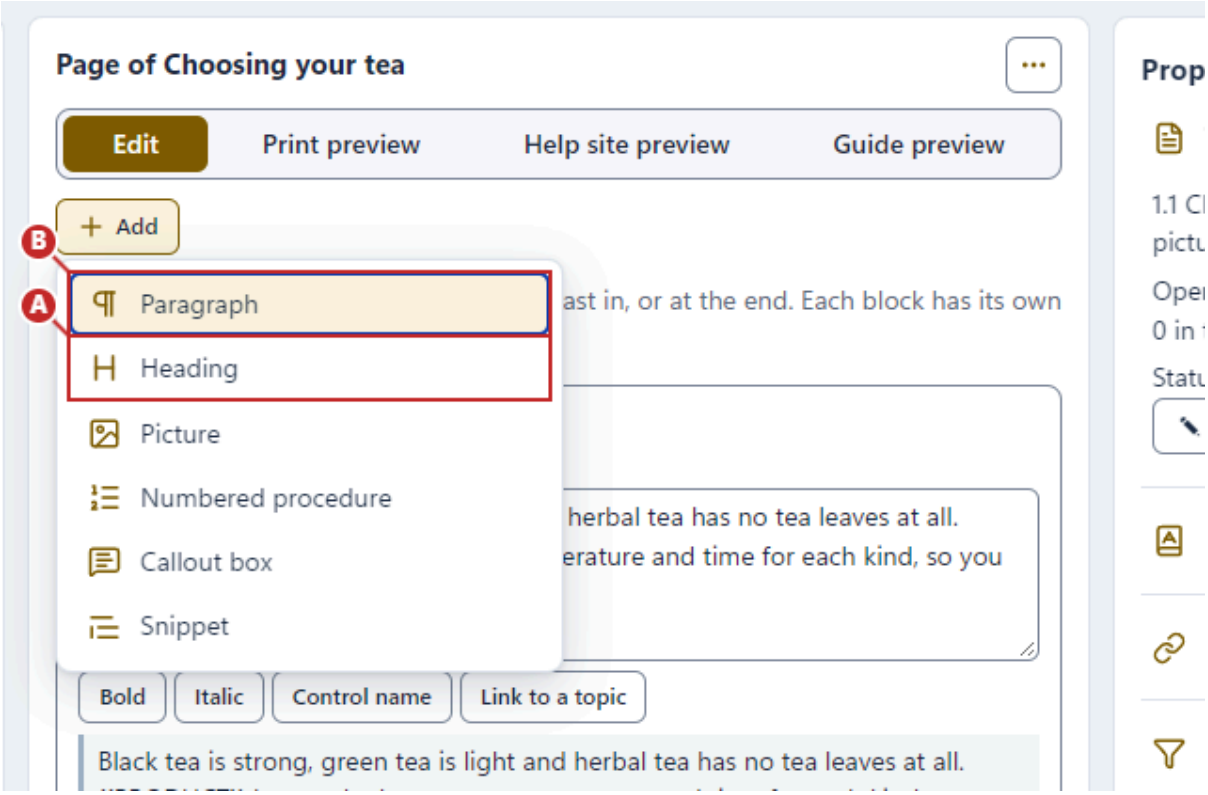
3.2 Writing a topic's page

A page is a stack of **blocks**. There are five kinds: a **paragraph**, a **heading**, a **picture**, a **numbered procedure** and a **callout box**. You add them one at a time with the **Add** button under the tabs of the page, or with the **Insert** menu.

OpenExplainer is not a free-flowing word processor. Each block is its own labelled box. That keeps the page quick to move around by keyboard, and it lets a screen reader say exactly which block you are in.

To add a heading and a paragraph:

1. Click **Add** under the tabs of the page, and choose **Heading** (A in the picture).



A. Heading: Puts a heading on the page.

B. Paragraph: Puts a block of text on the page.

Result: A heading box appears after the block you were last in, and the keyboard is already in it.

2. Type the heading.
3. Click **Add** again, and choose **Paragraph** (B).

   Result: A paragraph box appears. Type your text in it.

A heading is plain text: the marks for bold, italic and control names are not read in a heading, and the box says so. A new block always goes **after the block you were last in**, or at the end of the page if you were in none. Every block has its own **Move up**, **Move down**, **Comments** and **Delete** buttons. A button that cannot be used, such as Move up on the first block, is switched off.

Under each text box is a **preview** that shows how your words will look in the book. The editor keeps the keyboard in the box while you type, and never redraws the page under your fingers.

> **TIME SAVER**
> You do not need to save after every block. The navy bar says *Unsaved changes* beside the project's name until you do, and Ctrl+S saves.

To make words a link to another topic, see topic 3.8, page 35. To reuse a piece of a page in many topics, see topic 3.9, page 37.

3.3 Numbered steps

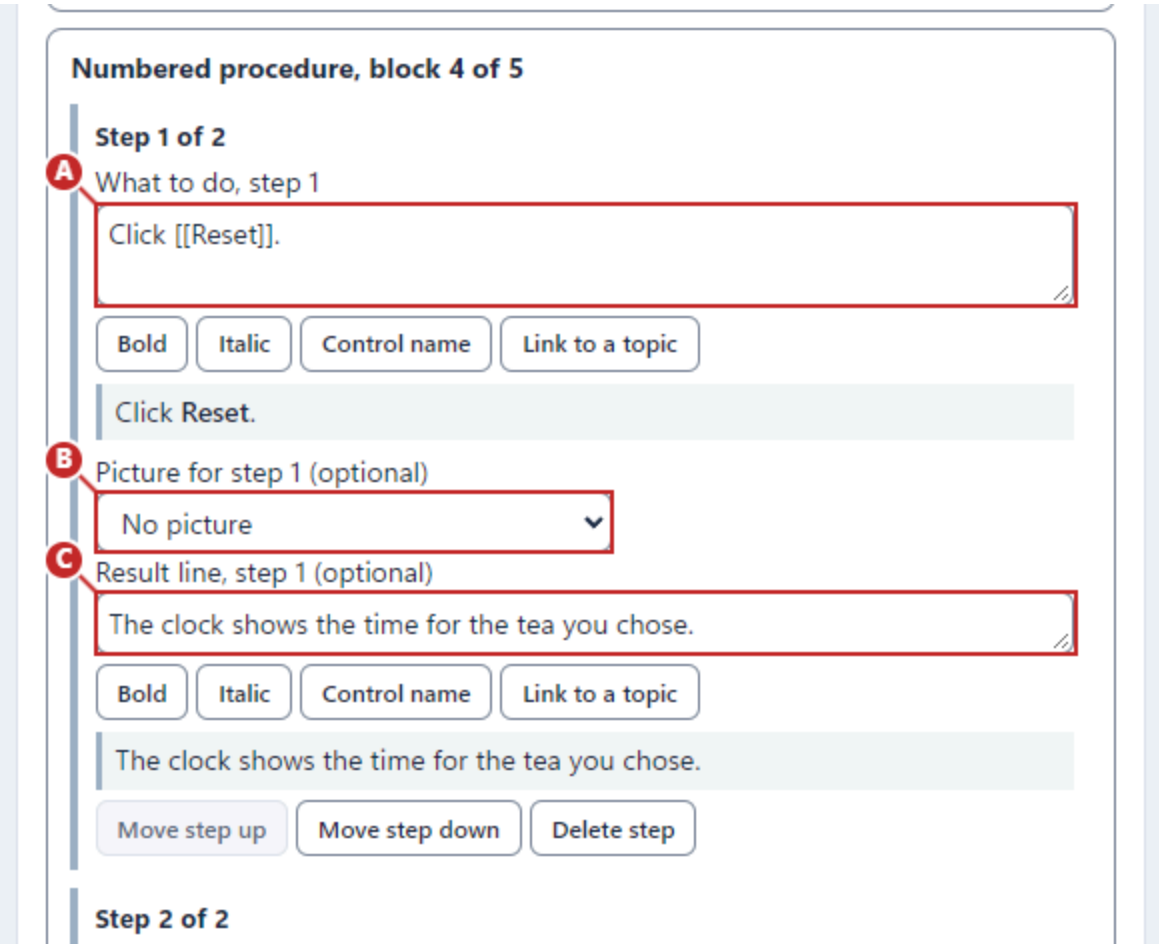
A **numbered procedure** is a list of steps. It is the part of a how-to book that readers use most, so it is worth writing well.

To write a numbered procedure:

1. Click **Add** under the tabs of the page, and choose **Numbered procedure**.

   Result: A procedure with one empty step appears, and the keyboard is in its text box.

2. Type what the reader does, as a short sentence that starts with a verb.



A. What to do: The sentence the reader follows.

B. Picture for step 1: Chosen from the pictures of the topic.

C. Result line: What the reader should see afterwards. It is optional.

3. Click **Add step** for the next step.

> **ϟ POWER TIP**
>
> If you take your pictures click by click (Chapter 4, page 42), OpenExplainer can write the first version of the steps from what you clicked, with no AI. Choose **Draft a procedure from the clicks** on the **Write** menu.

Each step has three parts. **What to do** is the sentence itself. A **picture for the step** is optional, and is chosen from the topic's pictures. A **result line** is optional too. It says what the reader should see afterwards, such as *The Save as window opens*. Give one only when the reader could not otherwise tell that the step worked, or when the next step needs what it makes. The numbers are never typed: they follow the order of the steps, and each procedure starts again at 1.

Writing good steps

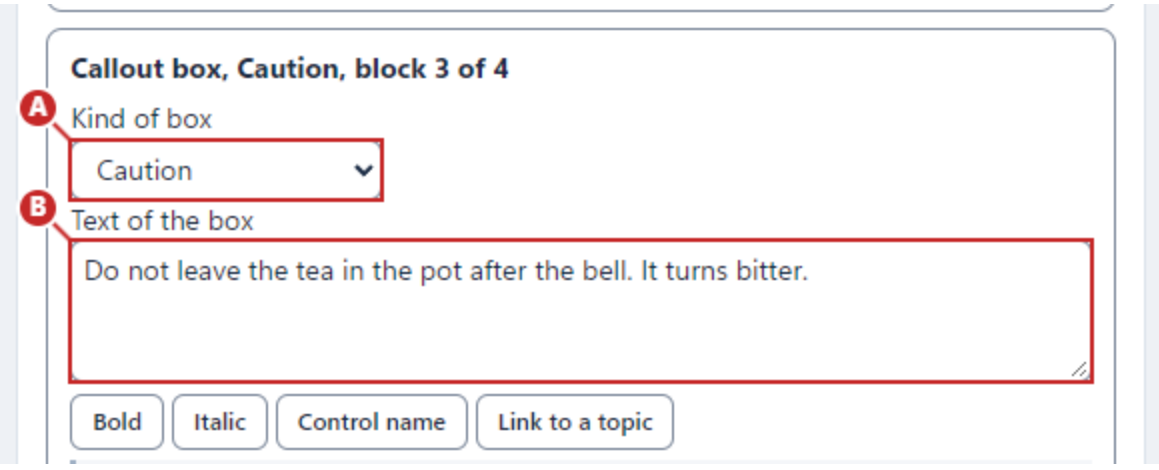
Start each step with a verb. Do one thing in each step. Name the control exactly as it appears on screen, and put its name between double square brackets so that it is shown in bold. Say **Click** for a button, **Choose** for a menu item, **Tick** for a tick box and **Select** for a choice in a list.

3.4 Writing boxed notes

A **callout box** stands out from the text and carries a short note. There are six kinds: **Design Tip**, **Caution**, **Bright Idea**, **Power Tip**, **Time Saver** and **New in this version**. Each kind says its name in words and has its own symbol, so you never have to rely on colour to tell them apart.

To add a boxed note:

1. Click **Add** under the tabs of the page, and choose **Callout box**.



A. Kind of box: Design Tip, Caution, Bright Idea, Power Tip, Time Saver or New in this version.

B. Text of the box: The words of the note.

Result: A box appears with a menu called Kind of box and a text box.

2. Choose the kind of box, and type its text.

> ✎ **DESIGN TIP**
> Keep a box to one or two short sentences. If a note needs more room, it is really a paragraph.

Using them well

Use **Caution** only where the reader can lose work or cannot undo something. Use a tip for everything else. If every note is a Caution, readers stop reading them.

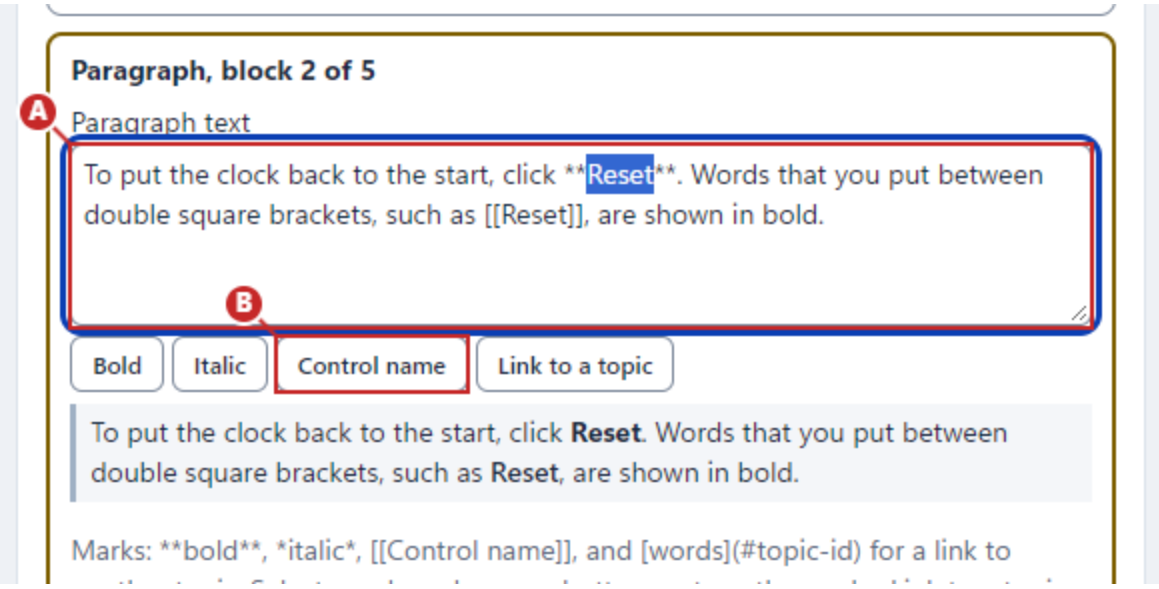
Put a box next to the step it bears on, not at the end of the topic. A reader who has reached the end has already done the step.

3.5 Marking bold, italic and control names

Three small marks make words bold, italic or a **control name**. A control name is the label of a button, menu or box exactly as it appears on screen. In the book it is shown in bold, and OpenExplainer also uses it to build the index and to check your book (Chapter 6, page 68).

To mark words as bold, italic or a control name:

1. Select the words in the text box.
2. Click **Control name**.



A. Text box: Select the words here first.

B. Control name: Marks the selected words as the name of a button or menu.

Result: The words are wrapped in marks, and the preview under the box shows them in bold.

Bold and **Italic** work the same way.

> ✳ **BRIGHT IDEA**
>
> If you press a mark button with no words selected, the editor tells you what to do instead of doing nothing.

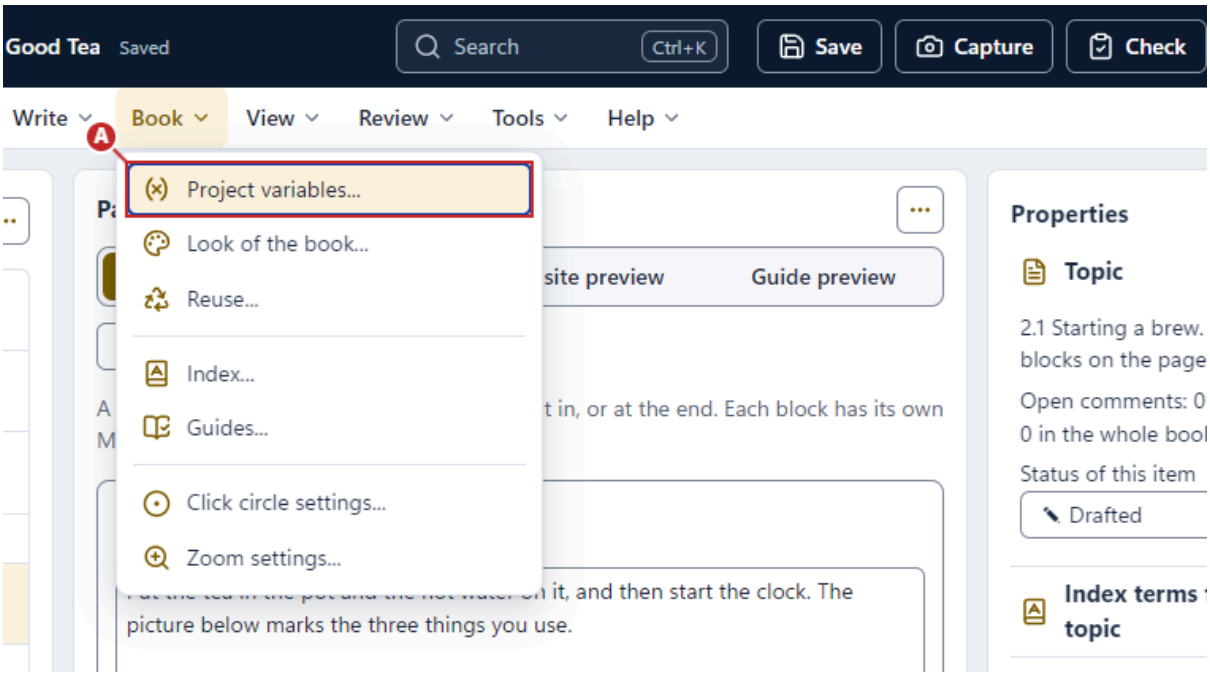
You can also type the marks yourself. Put two stars on each side of bold words and one star on each side of italic words. Put double square brackets around a control name. For example: **like this** for bold, *like this* for italic, and [[Save as]] for a control name. A backslash before a star or a bracket, as in those examples, keeps it as an ordinary letter.

3.6 Reusing the product name everywhere

A **variable** is a name that stands for a value you use in many places, such as the name of the program you are explaining. You write the variable's name between double curly brackets, with no spaces. OpenExplainer puts the value in when the book is made. When the product's name or version changes, you change it once.

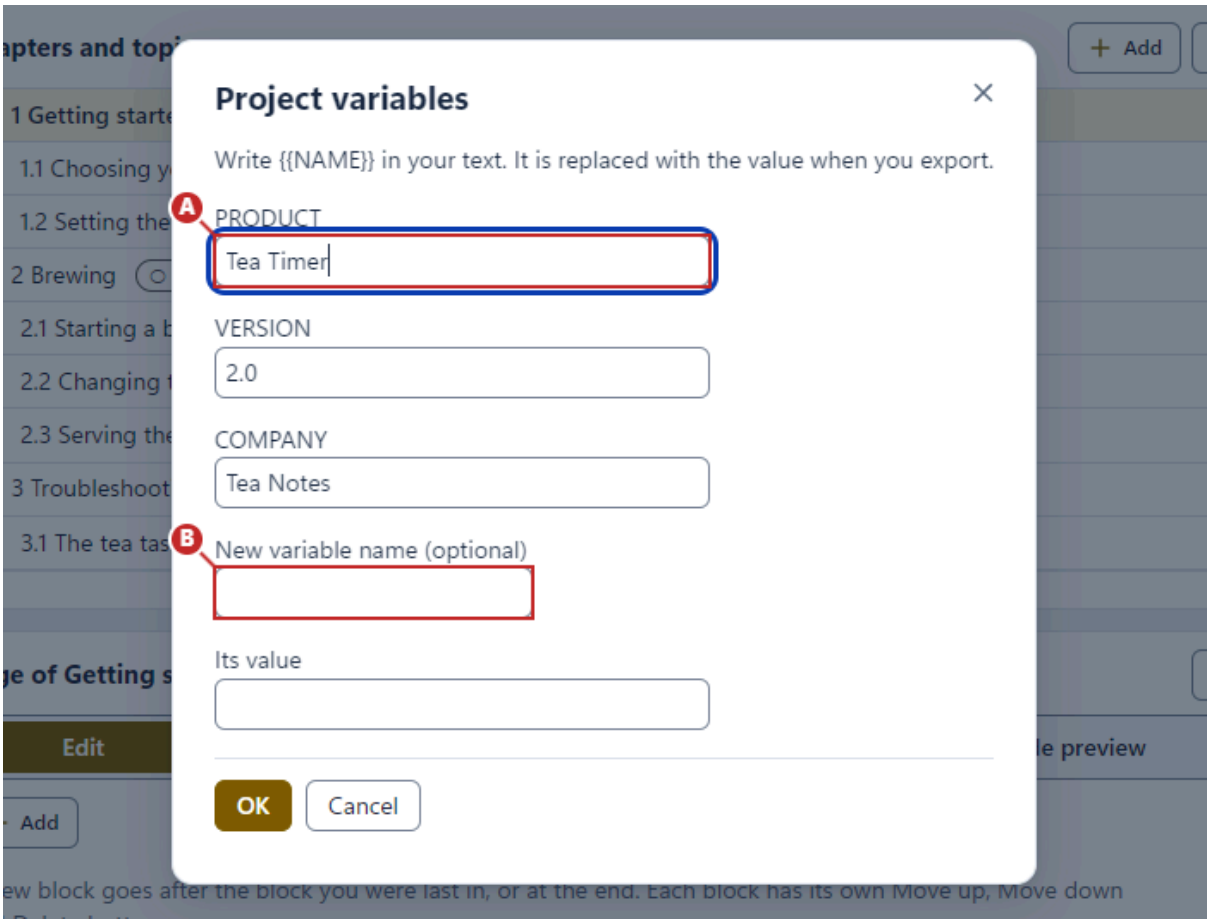
To set the values of your variables:

1. On the **Book** menu, choose **Project variables...**.



A. Project variables...: Opens the window of names and values.

2. Type a value in each box.



A. PRODUCT: One box for each variable the book has.

B. New variable name: Type a name here to add another variable.

3. To add another variable, type its name and its value at the bottom.
4. Click **OK**.

> ϟ **POWER TIP**
>
> A variable with no value is not hidden. Its bare name, still inside the curly brackets, would print in the book, and the book check tells you about it.

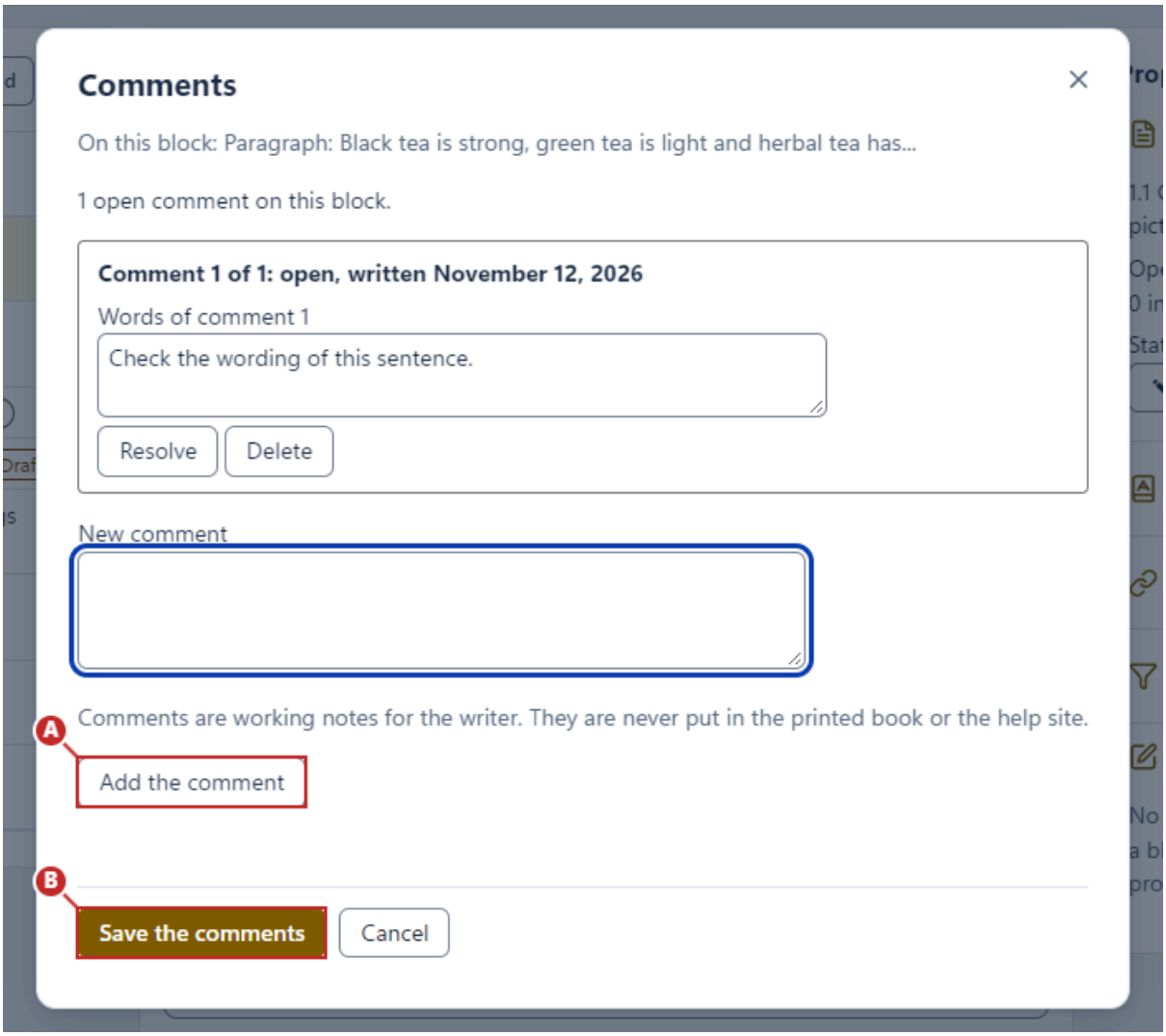
Three variables are special. **PRODUCT**, **VERSION** and **COMPANY** appear on the title page of the book. This guide uses them: its title page and many sentences come from the three values you can see in the window.

3.7 Leaving notes and tracking progress

A **comment** is a working note about one block, for you or for a colleague. A comment never goes into the book, the PDF or the help site. It goes into the JSON export only if you ask, and into the Word review copy (topic 6.3, page 73).

To write a comment on a block:

1. Click **Comments** on the block you want to write about.
2. Type your note in the New comment box.
3. Click **Add the comment**.



A. Add the comment: Adds what you typed in the box above it.

B. Save the comments: Keeps the comments.

Result: The button on the block now says how many comments are open.

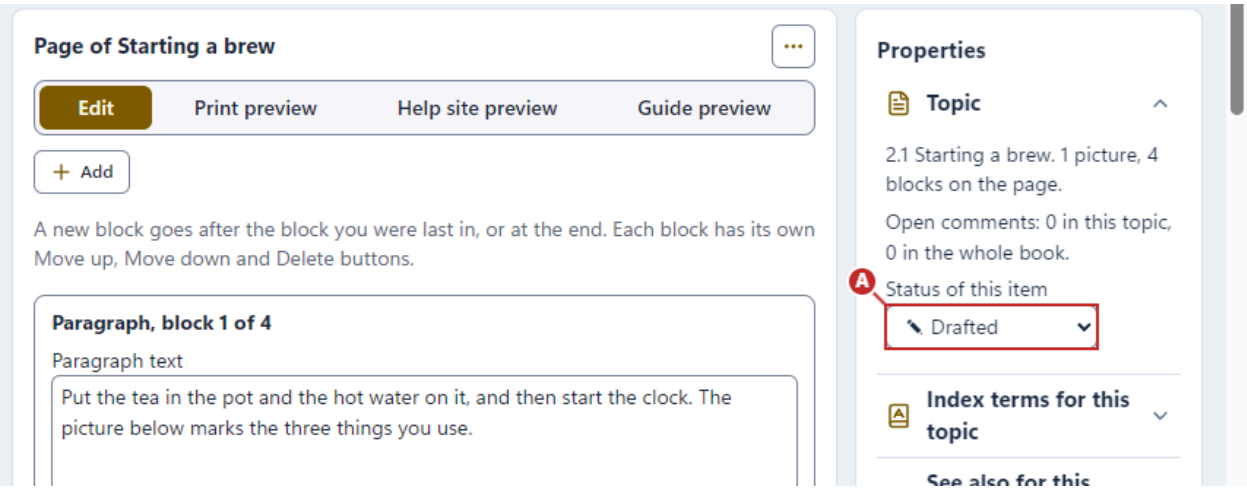
4. Click **Save the comments**.

When the point is dealt with, open the window again and press **Resolve**. A resolved comment is shown as resolved in words, not by style alone.

Choose **Unresolved comments…** on the **Review** menu to list every open comment in the book. **Go to the block** takes you to each one.

Status

Every topic also has a **status**. Choose it on the right, in the **Topic** section of Properties, under *Status of this item*: ○ Not started, ◐ Captured, ✎ Drafted, ✓ Reviewed or ⚠ Needs recheck. Each status has a word and a symbol, never a colour alone. Use them to see at a glance what is finished.



A. Status of this item: Not started, Captured, Drafted, Reviewed or Needs recheck, each with its own symbol.

To send the book to reviewers who do not have OpenExplainer, and bring their comments back, see topic 6.3, page 73.

3.8 Links between topics and See also

A **link** is a word or phrase in a paragraph that goes to another topic. It is better than typing a number such as *topic 6.1*, because a typed number goes wrong the moment you move a chapter. A link follows the topic wherever it goes and whatever it is called.

Making a link

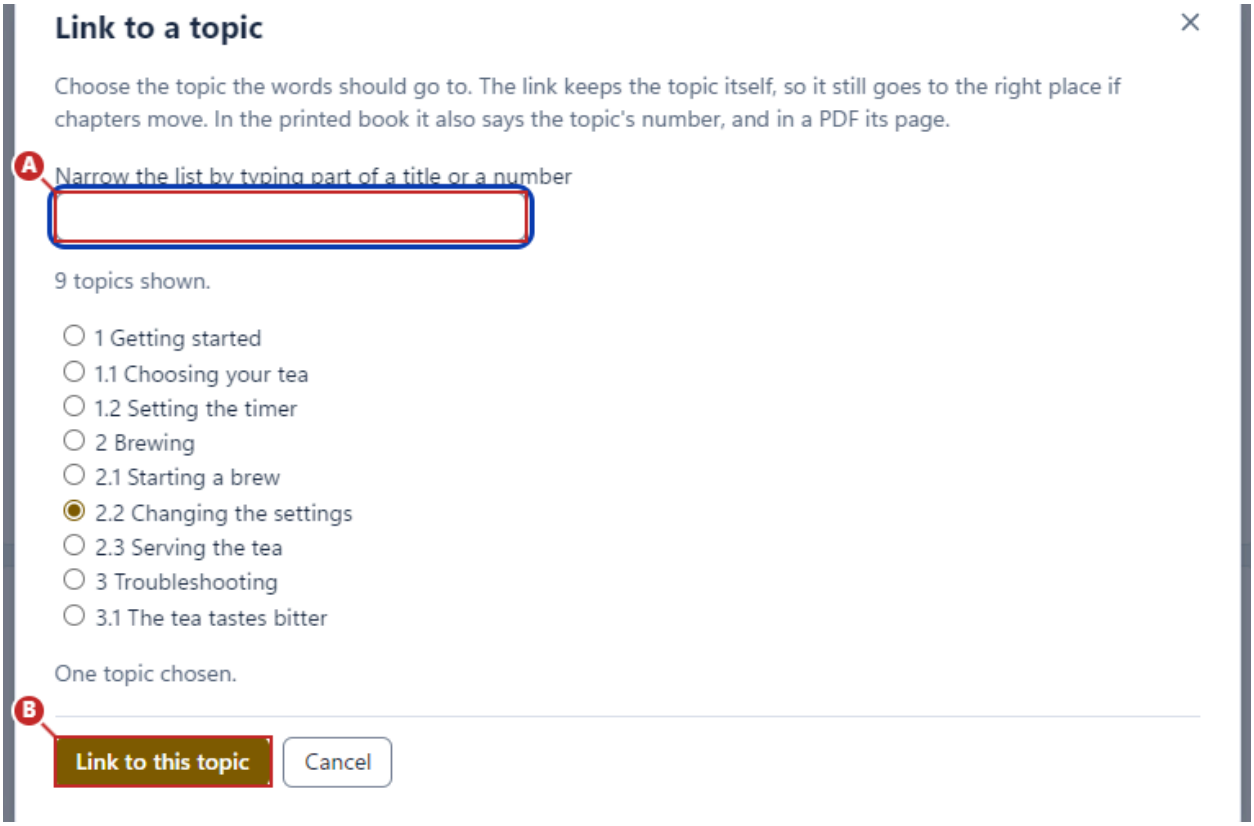
To make a link:

1. Select the words in a paragraph that should be the link.
2. Click **Link to a topic** on that block.

   Result: A window lists every chapter and topic with its number. A box at the top narrows the list as you type part of a title or a number.
3. Choose the topic in the list.

4. Click **Link to this topic**.

   Result: The words become a link. In the text box they are written as the words in square brackets, followed by the topic's code in round brackets. You do not have to type that.



A. Narrow the list: Type part of a title or a number.
B. Link to this topic: Makes the selected words a link to the chosen topic.

What the reader sees

In the **print preview, the PDF and the e-book**, the linked words are followed by the topic's number, such as *(4.2)*. In the PDF the page follows too, such as *(4.2), page 31*. In **Word** the words link to the topic's heading, and the number follows. In the **help site** the words are an ordinary link. If your linked words already say the number, such as *Chapter 4*, the number is not added again.

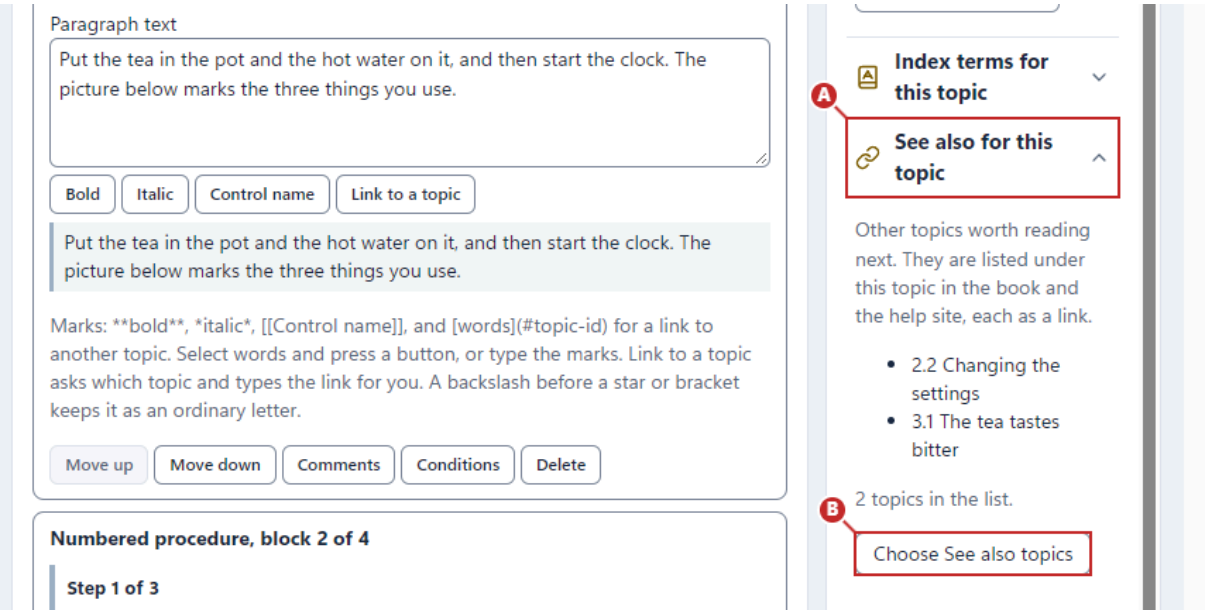
When a topic moves or is deleted

A link follows the topic when you rename it or move it. The number after the link changes by itself. If you delete a topic that other topics link to, OpenExplainer tells you how many links go to it. Their words stay as plain words, so no page is left with a broken link.

See also

To add a See also list to a topic:

1. Select the topic.
2. In the properties on the right, open the section **See also for this topic** (A in the picture), and click **Choose See also topics** (B).



A. See also for this topic: A section of Properties that opens and closes.

B. Choose See also topics: Opens a list of topics to tick.

3. Tick the topics worth reading next, up to 12.
4. Click **Keep these topics**.

   Result: A list called *See also* appears at the end of the topic in every output.

> ϟ **POWER TIP**
>
> Use **See also** for related reading, and a **link** inside a sentence for the thing the sentence is about. A reader should never have to guess where to look.

The book check lists a link or a See also entry that goes to a topic that is not in the book (topic 6.1, page 68). It also lists links whose words say nothing about where they go, such as *here* or *read more*. A screen reader user who goes through the links of a page hears only those words.

3.9 Reusing text and building variants

Some things belong in more than one place. One example is the same warning in six procedures. Another is a step that is true for one version of the program and not for another. OpenExplainer has three tools for this, all in one window called **Reuse**.

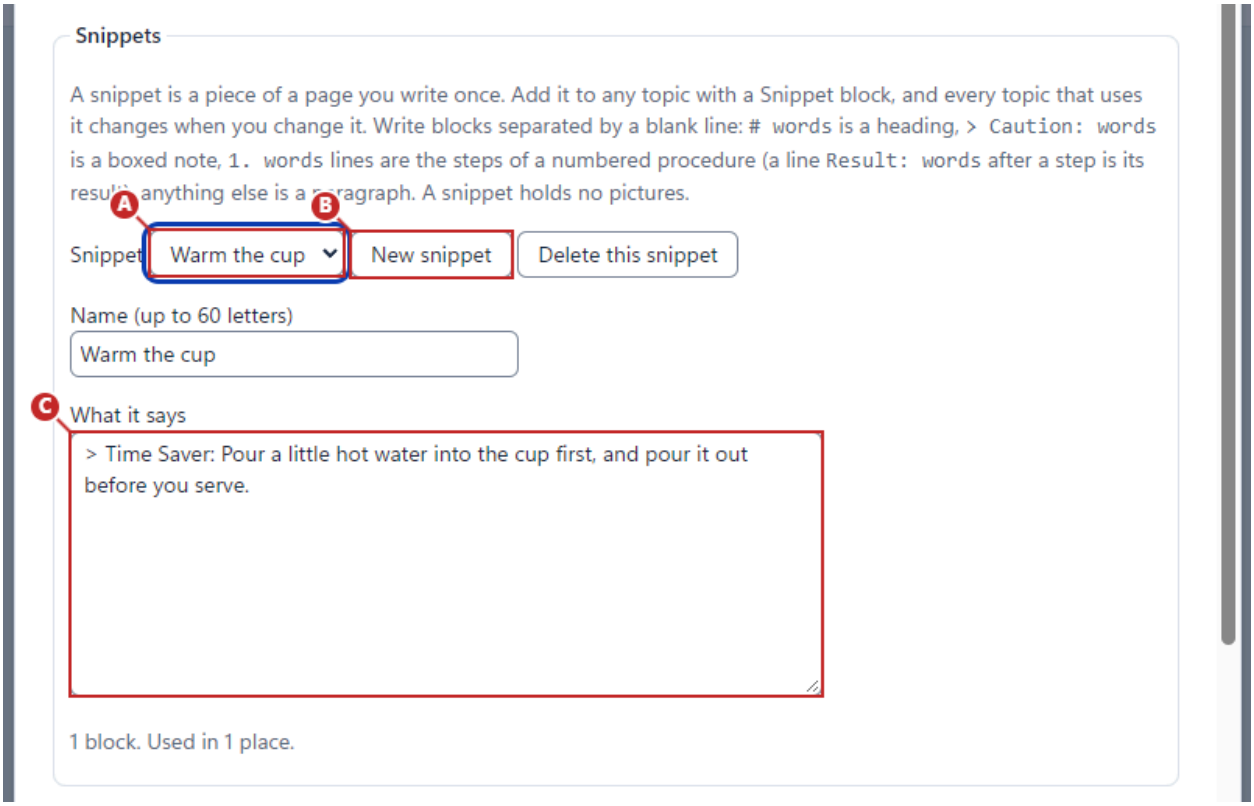
Snippets

A **snippet** is a piece of a page that you write once and use in many topics. It can be a heading, a paragraph, a boxed note or a numbered procedure. It cannot hold a picture or another snippet. Change the snippet, and every topic that uses it changes, in the preview and in every output.

To write a snippet:

1. On the **Book** menu, choose **Reuse...**.
2. Click **New snippet**.
3. Type a name of up to 60 letters.
4. Write what the snippet says.
5. Click **Save**.

The window explains the small marks. A blank line divides the blocks. A line that starts with # is a heading. A line that starts with > Caution: is a boxed note. Lines that start with 1. and 2. are the steps of a procedure.



A. Snippet: Chooses which snippet you are editing.
B. New snippet: Starts another snippet.
C. What it says: The words of the snippet, written with the small marks the window explains.

To use a snippet in a topic:

1. Select a topic.

2. Click **Add** under the tabs of the page, and choose **Snippet**.
3. Choose the snippet from the list.

If you delete a snippet, its words stay in every topic that used it, as ordinary blocks, so no page loses anything.

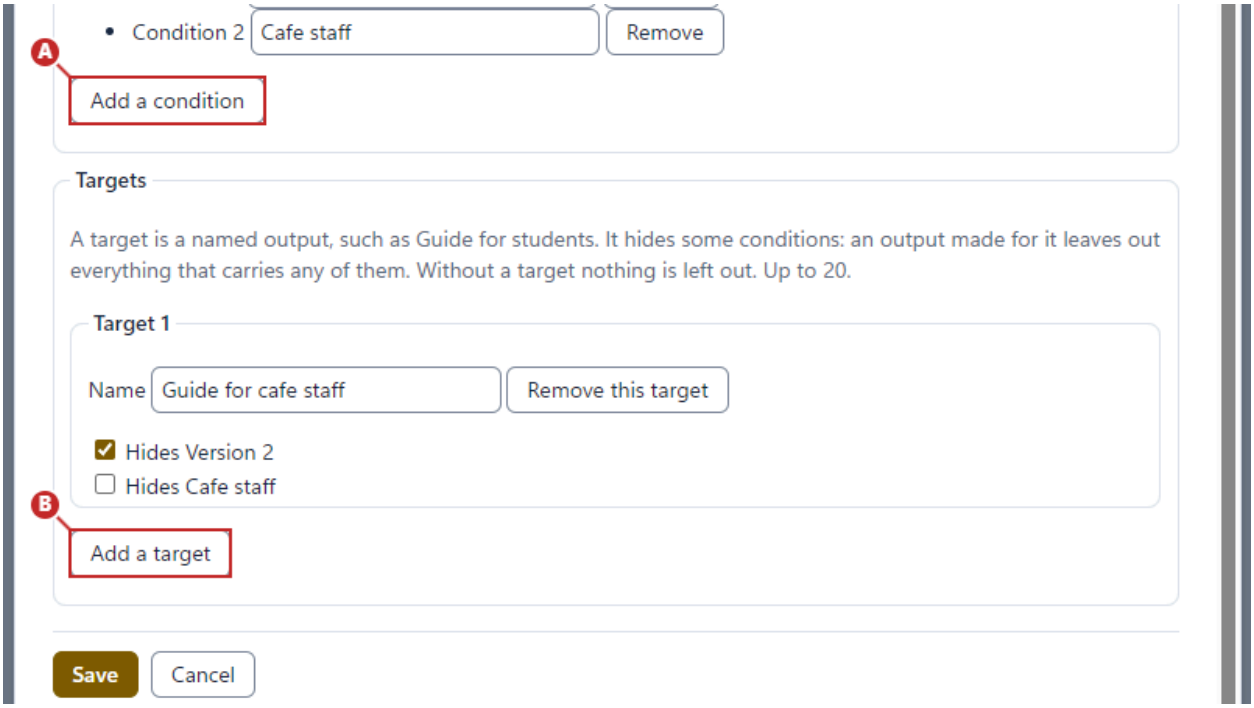
Conditions

A **condition** is a name for content that is only for some readers or some versions, such as *Administrators* or *Version 2*. In the Reuse window, **Add a condition** gives you a box for its name. Once the book has conditions, a block has a **Conditions** button that lists the conditions it carries. A topic has **Choose conditions for this topic** in the section **Conditions of this topic** of Properties.

Targets

A **target** is a named output, such as *Guide for students*, with a tick box for each condition it hides. When you export, **Make it for** leaves out everything that carries a condition the target hides. A hidden step goes from its procedure, and a hidden block goes from its page. A hidden topic takes the topics under it.

The preview has **Show the book for**, so you can see what a target gives before you export. With no target nothing is left out, and the JSON export always holds the whole book.



A. Add a condition: Adds a box for a condition's name.
B. Add a target: Adds a named output with a tick box for each condition it hides.

> **DESIGN TIP**
> To make one book for two versions of a program, put the steps that differ in blocks with a condition. Make a target for each version, and export once for each.

The book check notes snippets that no topic uses, snippets with nothing in them, and conditions that nothing carries (topic 6.1, page 68).

3.10 Bringing in a Word manual

If you already have a manual in Word, you do not have to type it again. **Import from Word** reads a .docx file and adds it to your book as chapters and topics.

> **TIME SAVER**
> Take a snapshot first (topic 2.7, page 21) if the book already matters to you. Then, if the import is not what you wanted, you can go back.

To bring in a Word manual:

1. On the **File** menu, choose **Import from Word...**.
2. Choose the .docx file.

   Result: A window tells you how many chapters, topics and pictures will come in, and what will be left out.
3. Read what the window says, and click **Add them to the book**.

   Result: The chapters are added at the end of the book, and every topic is marked Drafted. Cancel changes nothing.

> **POWER TIP**
> Every topic that comes in is marked **Drafted**. The words came from a file made for another tool, and you have not read them here yet. Read each topic, check its pictures' descriptions, then change its status.

What comes in

Headings. The top level of heading becomes chapters and the next level becomes topics. Deeper headings become sub-headings inside their topic. A file with only one level of heading makes each heading a topic. Numbers typed at the start of the headings, such as *1.2*, are removed, because OpenExplainer numbers the book itself.

Words. Bold, italic and a character style called *Control name* are kept. A link keeps its words.

Steps and lists. Numbered paragraphs become a numbered procedure, one for each list. A bulleted paragraph becomes a paragraph that starts with a bullet.

Boxes. Paragraphs in styles named Note, Tip, Caution, Warning or Important become boxed notes. So do the box styles that OpenExplainer writes in its own Word files.

Pictures. PNG and JPEG pictures come in with their descriptions. A table becomes one line for each row, with the cells divided by a bar.

What is left out

OpenExplainer leaves out the title page, contents list and index of a file it made. It also leaves out running headers and footers, comments, footnotes, endnotes, and pictures that are not PNG or JPEG. The window counts what was left out, so nothing goes missing without being said. Save a Word file in the older .doc form as .docx first.

CHAPTER 4

Taking pictures

In This Chapter

- Capturing a window
- Capturing a rectangle of the screen
- Capturing click by click
- Reviewing your clicks
- Showing where the reader clicks
- Importing pictures you already have

A how-to book is mostly pictures with words around them. OpenExplainer has three ways to take a picture of another program, and a way to bring in pictures you already have. This chapter explains each one and when to use it.

Everything here is for **still pictures**. OpenExplainer never records video, and it never records what you type.

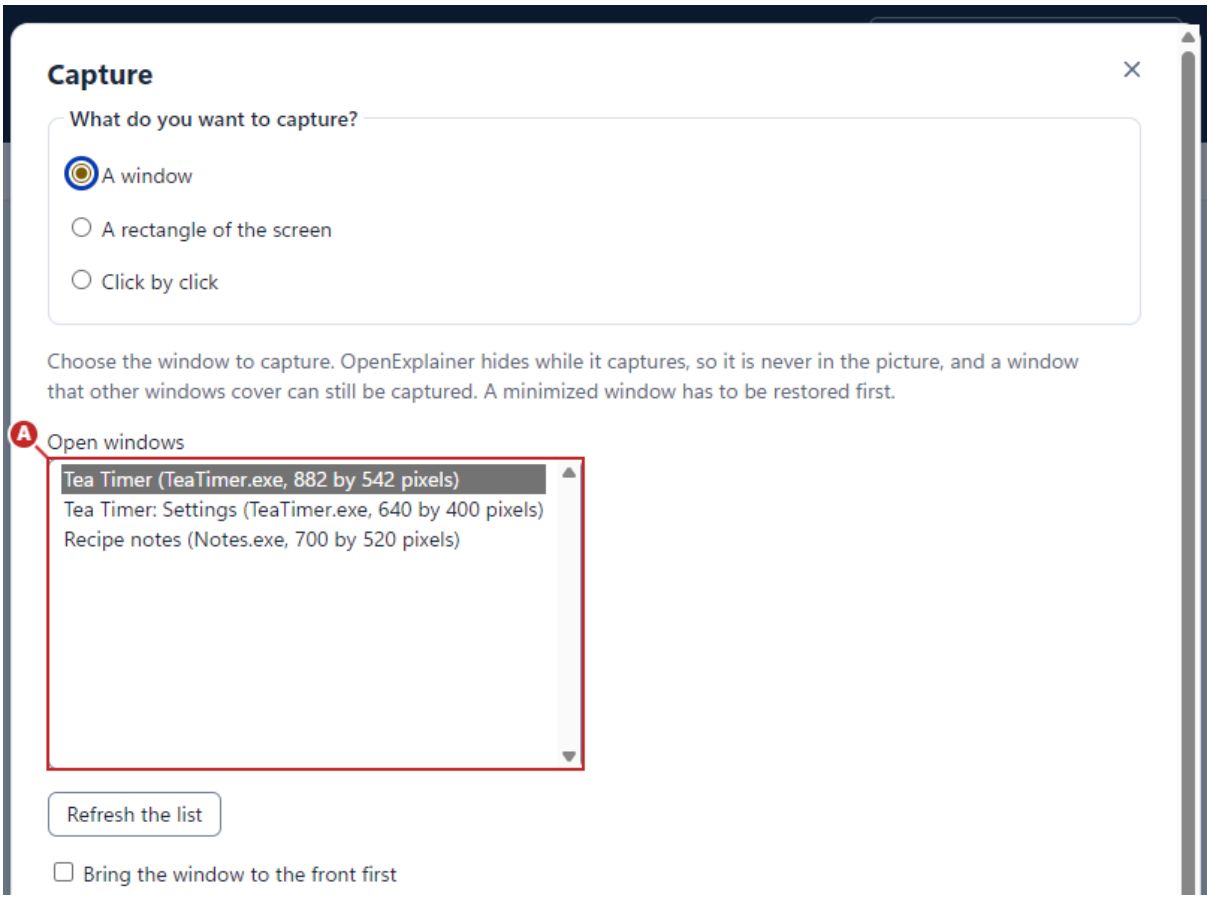
4.1 Capturing a window

Use this when you want a picture of **one program's window**. OpenExplainer hides itself while it captures, so it is never in the picture. A window that other windows cover can still be captured, because Windows is asked to draw that window itself.

To capture one program's window:

1. Select the topic that the picture belongs to, in the list on the left.

2. On the **Insert** menu, choose **Capture...**, or click **Capture** in the navy bar.



A. Open windows: Each entry gives the window's title, the program's name and its size.

Result: A window called *Capture* opens.

3. Choose **A window**.
4. Select the program's window in the list called *Open windows*.

   Result: Each entry gives the window's title, the program's name and its size. A window that belongs to another window, such as the Save or Open window of the program, is in the list too.

5. Click **Capture**.

   Result: OpenExplainer hides for a moment, takes the picture and comes back. The picture is in the topic's list of pictures.

> **POWER TIP**
>
> The picture shows exactly what is in the window at that moment. Look at each picture before you share the book, in case something private was open.

A window that is **minimized** has to be restored first, and the list says so. If the program's title bar comes out grey, because it was not the active window, tick **Bring the window to the front first**. That draws it as the active window, but it changes which window is on top for a moment.

If you need a moment to get the other program ready, type a number in **Wait before capturing, in seconds** (0 to 60). OpenExplainer hides, and a small Cancel window counts down. The picture is taken when the count reaches 0. A capture normally takes a fraction of a second. If one takes longer, press the Cancel button in the small Cancel window.

4.2 Capturing a rectangle of the screen

Use this when the thing you want is **not a whole window**: a part of a window, a menu, or a tool tip. A menu is a window of its own, so a capture of the window leaves it out, but a rectangle of the screen includes it.

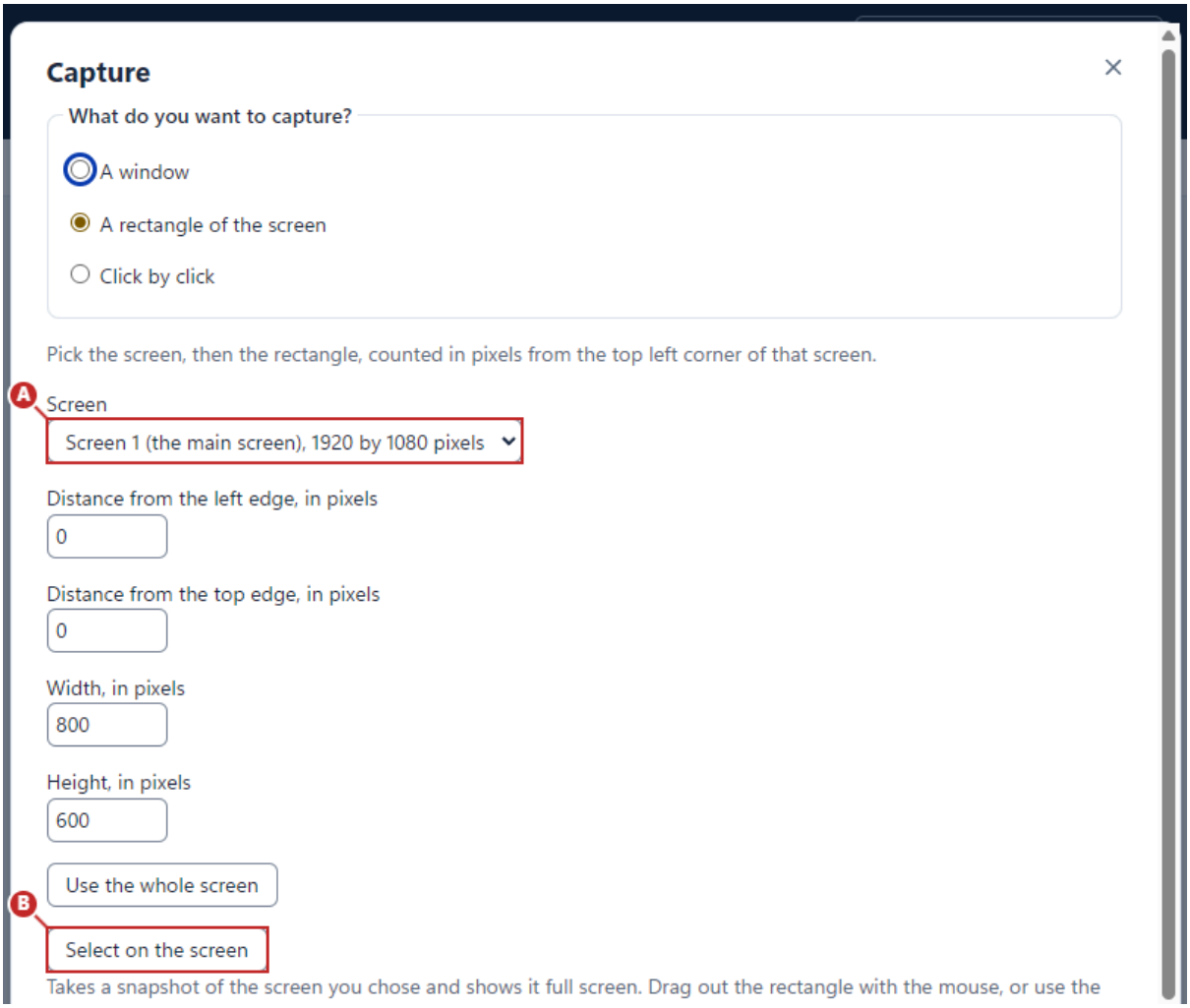
> **⚠ CAUTION**
> A rectangle of the screen shows whatever is in that rectangle, which can be more than one program. Check the picture before you share the book.

To capture a rectangle of the screen:

1. On the **Insert** menu, choose **Capture...**, or click **Capture** in the navy bar.

   Result: A window called *Capture* opens.

2. Choose **A rectangle of the screen**.

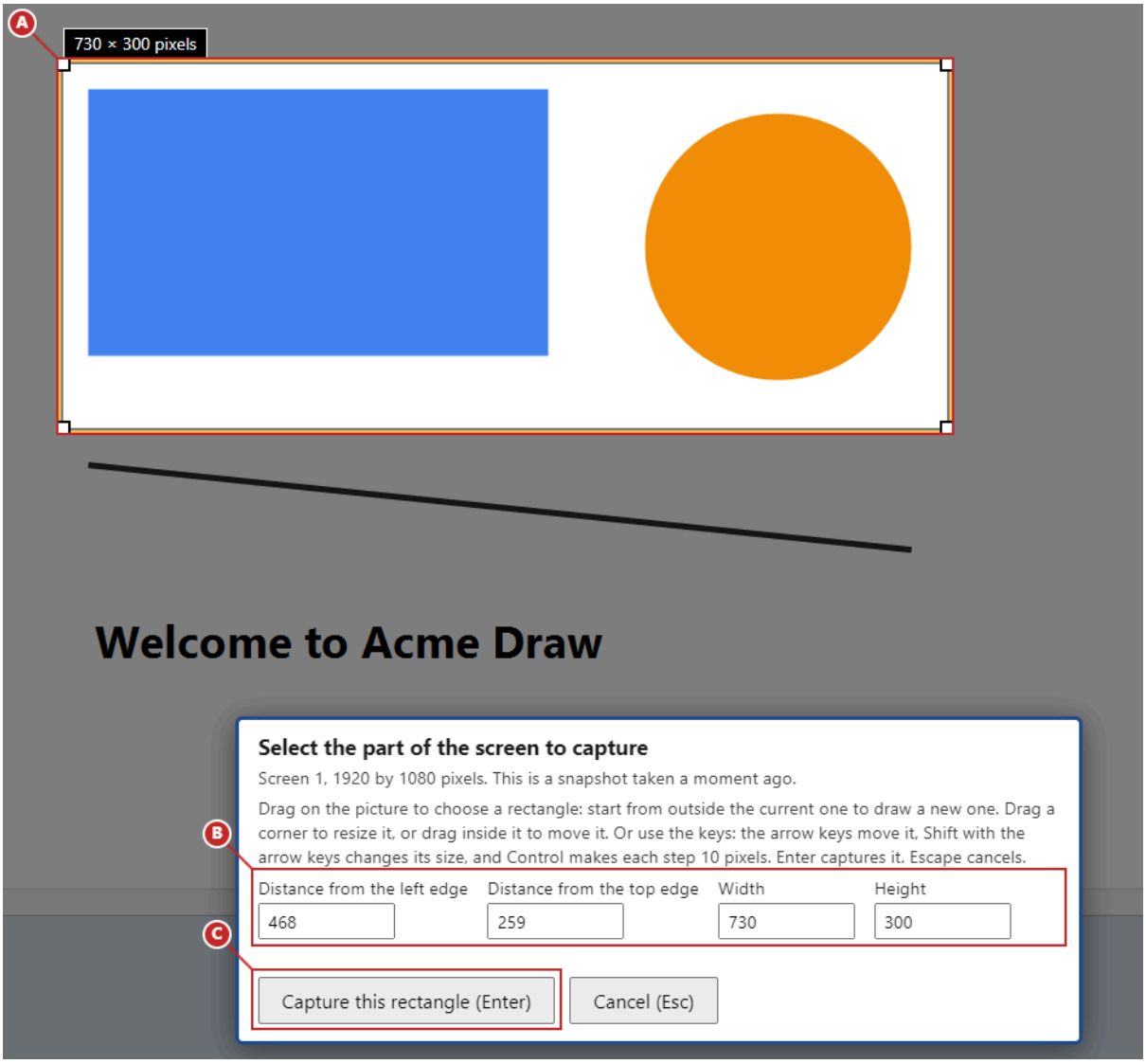


A. Screen: Which screen to take the rectangle from.

B. Select on the screen: Shows the screen full size so that you can drag the rectangle out.

3. Choose the screen, if you have more than one.

4. Choose the rectangle. Click **Use the whole screen** or **Select on the screen**, or type four numbers, counted in pixels from the top left corner of that screen.



A. The rectangle: Drag on the picture to draw it, drag inside it to move it, or drag a corner to change its size. Its size shows beside it.

B. The four number boxes: The same rectangle in numbers: the distance from the left and top edges, the width and the height. They change as the rectangle changes, and typing in them changes the rectangle.

C. Capture this rectangle: Takes the picture of the rectangle. Enter does the same.

5. Click **Capture**.

Select on the screen takes a snapshot of the screen you chose and shows it full screen, as in the picture. Drag out the rectangle (A) with the mouse, or use the arrow keys or the four number boxes (B). Click **Capture this rectangle (Enter)** (C) to take the picture.

To capture a menu, set **Wait before capturing** to a few seconds and click Capture. Then open the menu in the other program before the countdown ends.

4.3 Capturing click by click

This is the fastest way to document a task. You click through the steps in the other program, and OpenExplainer records **every click** as a picture, with a note of what you clicked. Afterwards you keep the pictures you want and describe them.

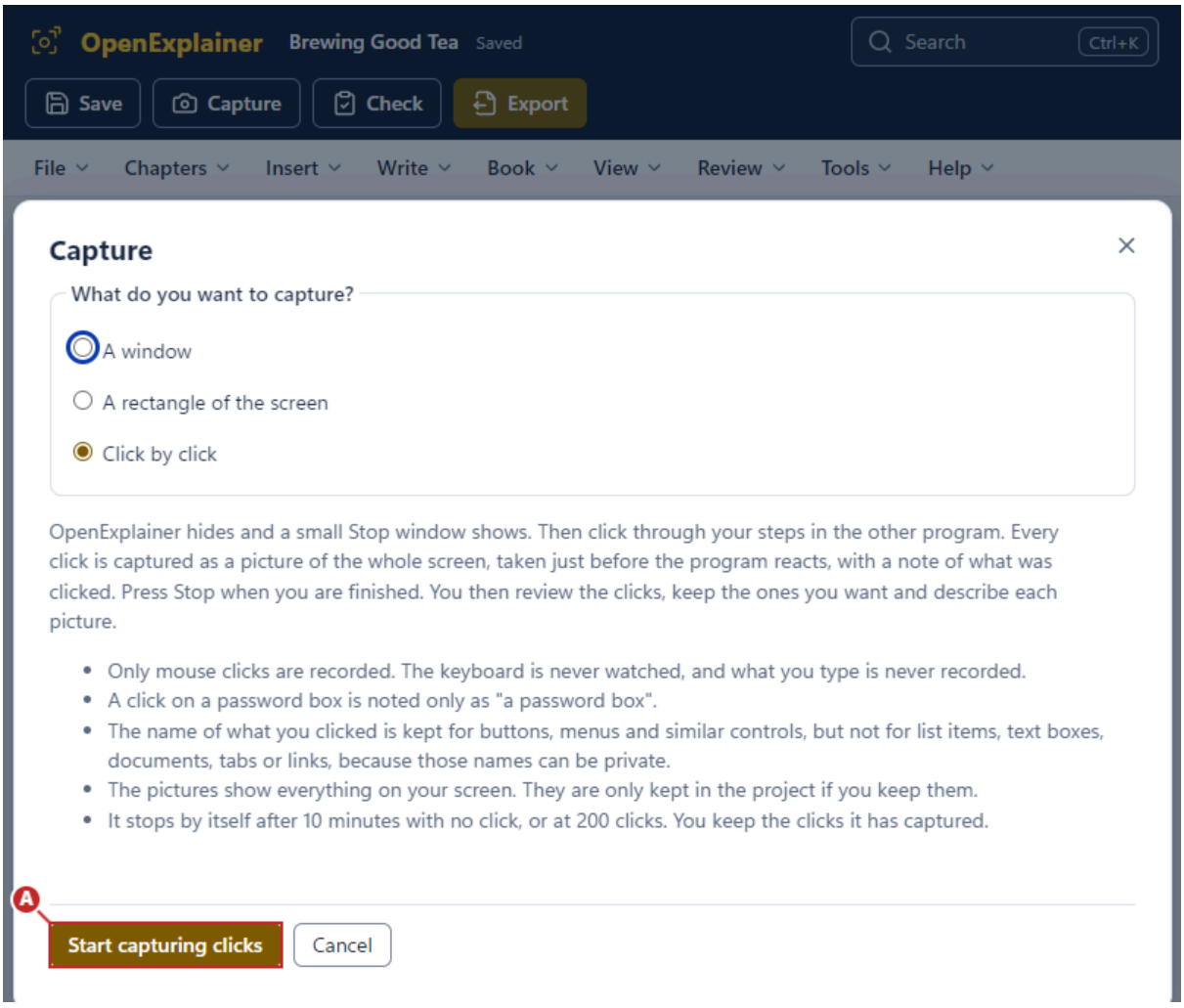
> **⚠ CAUTION**
> The pictures from the Capture window show **everything on your screen**, not only the program. Close private windows before you start. A picture is kept in the project only if you keep it, so throw away any picture that shows something it should not.

To record a task click by click:

1. On the **Insert** menu, choose **Capture...**, or click **Capture** in the navy bar.

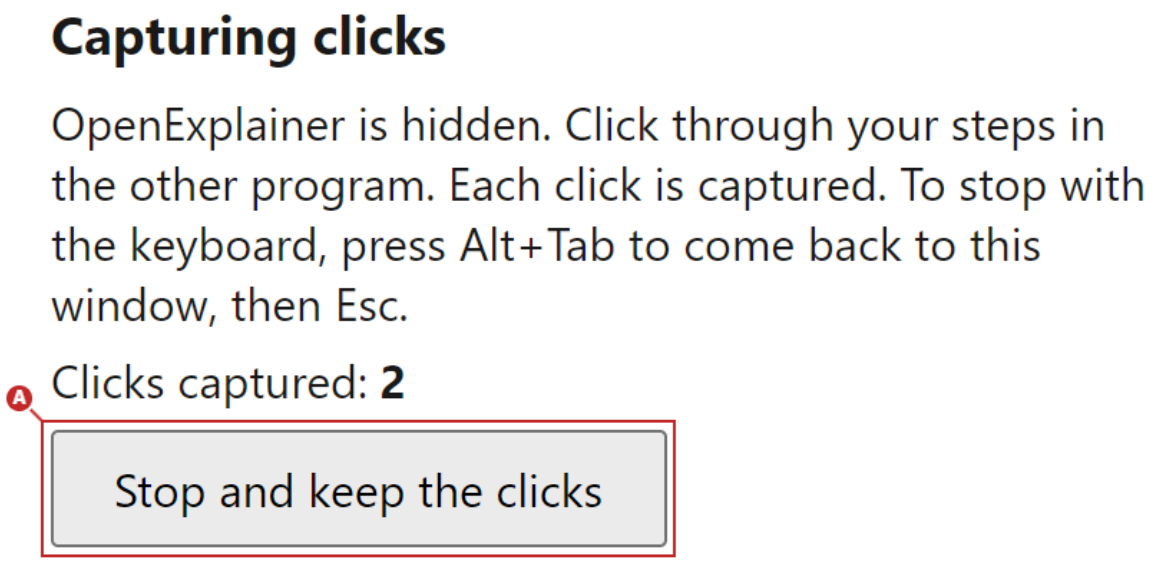
   Result: A window called *Capture* opens.

2. Choose **Click by click**.



A. Start capturing clicks: Hides the program and shows the small Stop window.

3. Click **Start capturing clicks**.



A. Stop and keep the clicks: Ends the capture and keeps every click so far. OpenExplainer comes back and shows the clicks for review.

Result: OpenExplainer hides, and a small window with a Stop button shows (A in the picture). It stays on the screen the whole time, and it counts the clicks.

4. Use the other program as the reader would: click through the steps, one at a time.
5. Click **Stop and keep the clicks** (A) in the small window.

   Result: A window called Review your clicks opens (see the next topic).

> **TIME SAVER**
> Click slowly and one thing at a time, and wait for the program to react before you click again. Do one task per capture, so each set of pictures becomes one numbered procedure.

What is and is not recorded

Only mouse clicks are recorded. The keyboard is never watched, and what you type is never recorded. A click that lands on a password box is noted only as *a password box*.

The name of what you clicked is kept for buttons, menus and similar controls. It is not kept for list items, text boxes, documents, tabs or links, because those names can be private.

Each picture is taken before the program reacts to the click, so it shows what the reader sees at the moment of the click. The capture stops by itself after 10 minutes with no click, or after 200 clicks. You keep the clicks it has captured.

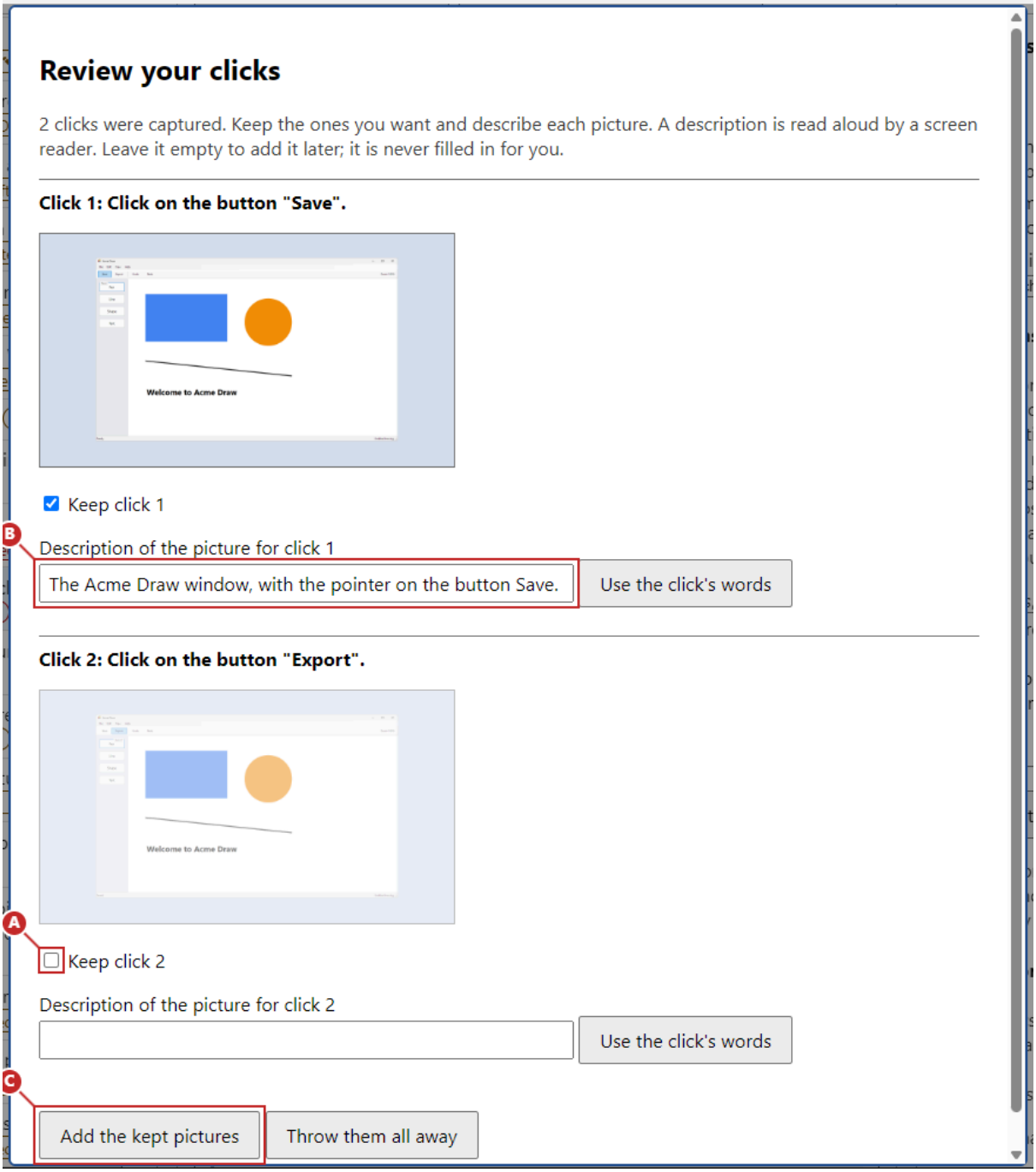
The picture of the small Stop window was taken on 1 November 2026. That window and the Cancel window are left out of every screen copy on purpose, so they cannot be photographed with the rest of the screen. Neither was seen again for this edition (topic 1.5, page 7).

4.4 Reviewing your clicks

When you press Stop, a window called **Review your clicks** lists every click that was captured. It says how many clicks there were. Each click is a small picture with a note of what was clicked, such as the button Save.

To review the clicks:

1. Look through the list, and untick the pictures you do not want (A in the picture).



A. Keep click 2: Untick a click to leave its picture out.

B. Description of the picture for click 1: What the picture shows, for a screen reader. It starts empty and is never filled in for you.

C. Add the kept pictures: Adds the pictures you kept to the topic, in the order you clicked.

2. Describe each picture you keep, in the box beside it (B).

   Result: A screen reader reads a description aloud. You can leave it empty and add it later. It is never filled in for you.

3. Click **Add the kept pictures** (C).

   Result: The pictures are added to the topic, in the order you clicked.

The picture of this window was taken on 1 November 2026, before the look of OpenExplainer changed, and it was not taken again for this edition, because the window appears only after a real click capture. The window is made the same way as every other window, so it now also has a **Close** button in its corner. Apart from that it may differ a little from the picture.

> ✳ **BRIGHT IDEA**
>
> The steps it writes say Click for a button, Choose for a menu item and Tick for a tick box. They never describe typing, because a capture does not record it. Add that yourself.

If you click **Throw them all away**, nothing is added and the clicks are forgotten.

Every kept picture that has an outline of the clicked item also gets a first **callout** by itself. It is named from the item's name when the name was kept.

When the click was on the window itself, Windows could not say what was clicked. The callout is then a small box round the click, with no name. Chapter 5, page 55 shows how to work with callouts.

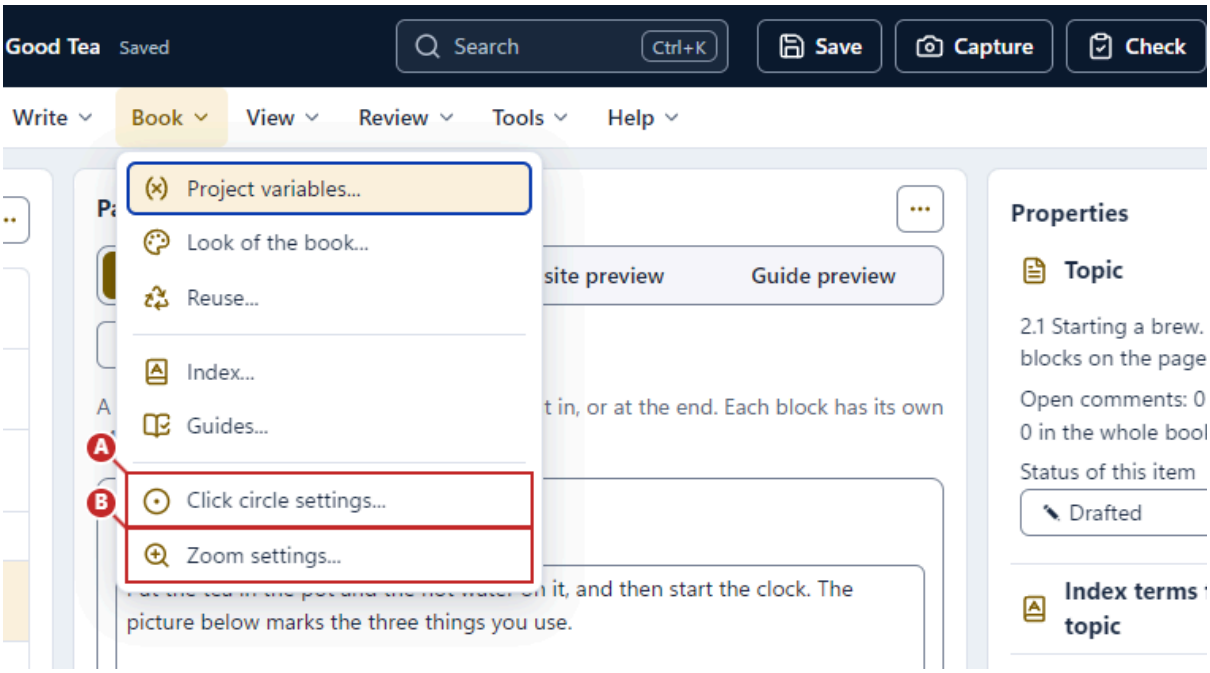
Then choose **Draft a procedure from the clicks** on the **Write** menu, or from the **…** button in the Pictures area. OpenExplainer writes a numbered step for each picture from your click notes, with no AI. It waits in Drafts until you accept it.

4.5 Showing where the reader clicks

A transparent **click circle** can be drawn on the spot of every click, so the reader sees exactly where to click. The circle is drawn from the click's position when the book is made. It is never drawn into the picture itself, so you can turn it on, off or move it whenever you like.

To set the click circle:

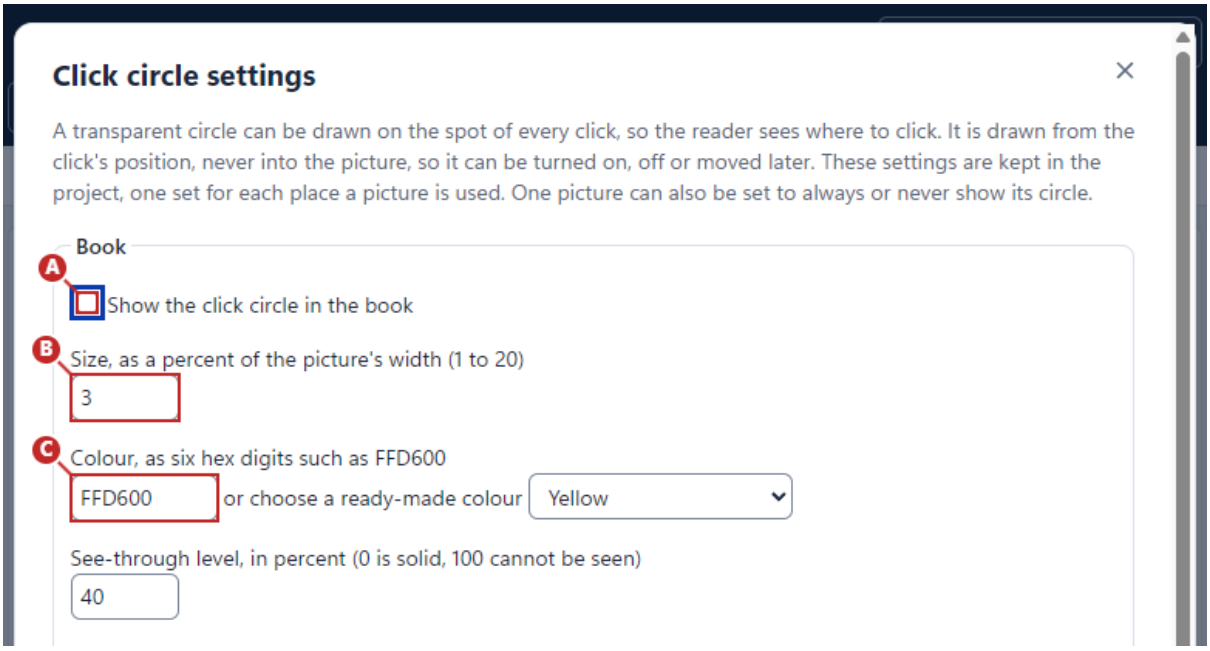
1. On the **Book** menu, choose **Click circle settings...** (A in the picture).



A. Click circle settings...: Opens the window for the click circle.

B. Zoom settings...: Sets how close a zoom goes.

2. Tick the box to show the circle in the book.



A. Show the click circle in the book: Turns the circle on or off.

B. Size: A percent of the picture's width.

C. Colour: Six hex digits, or a ready-made colour from the list.

3. Choose the circle's size, its colour and how see-through it is.

   Result: A small sample picture shows how the circle will look.

4. Click **Save the settings**.

Size is a percent of the picture's width (1 to 20). **Colour** is six hex digits such as FFD600, or a ready-made colour: Yellow, Orange, Green, Blue, Pink, Red or White. **See-through level** is a percent, where 0 is solid and 100 cannot be seen.

> ✎ **DESIGN TIP**
> A circle about 4 percent wide in the default yellow stands out without hiding what was clicked.

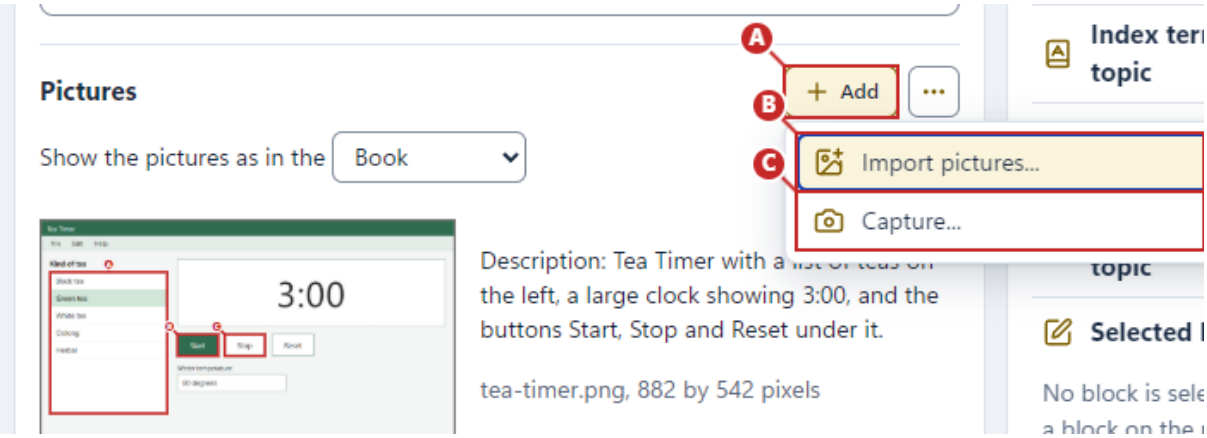
Each click picture in the list also has its own choice called **Click circle**: *Follow the project setting*, *Always show* or *Never show*. Beside it, **Move the circle** opens a window where you type a different place for the circle. *Put it back on the click* undoes that. Above the list of pictures, *Show the pictures as in the* Book or Quick guide shows them as they will look in each place.

4.6 Importing pictures you already have

You can also bring in pictures you already have, from another screenshot tool or from a designer. OpenExplainer accepts **PNG** and **JPEG** pictures.

To bring in pictures you already have:

1. Select the topic.
2. On the **Insert** menu, choose **Import pictures...**. You can also click **Add** in the Pictures area (A in the picture), and choose **Import pictures...** (B).



A. Add: Opens a menu for bringing pictures in.

B. Import pictures...: Adds PNG or JPEG pictures you already have.

C. Capture...: Takes a picture of another program.

3. In Windows' window for choosing files, choose the pictures (you can choose more than one), and click **Open**.

   Result: A window called Describe the picture asks what each one shows.

4. Describe each picture, or leave the box empty to do it later.

   Result: A picture with no description is marked *Needs a description* in the list until you write one.

> **POWER TIP**
>
> OpenExplainer copies the picture into your project. If you later change the original file, the book keeps the copy it made.

An imported picture has no click note, so it has no first callout and no click circle. You can still add callouts by hand (Chapter 5, page 55), and use the picture in a step or as a picture block.

CHAPTER 5

Explaining your pictures

In This Chapter

- Describing a picture for readers who cannot see it
- Pointing at part of a picture
- Showing only part of a picture
- Drawing marks on a picture

A picture shows what a program looks like. It does not say what to look at or what it is for. It says nothing at all to a reader who cannot see it. This chapter covers the four things that turn a picture into an explanation: a **description**, **callouts**, a **view** and **marks**.

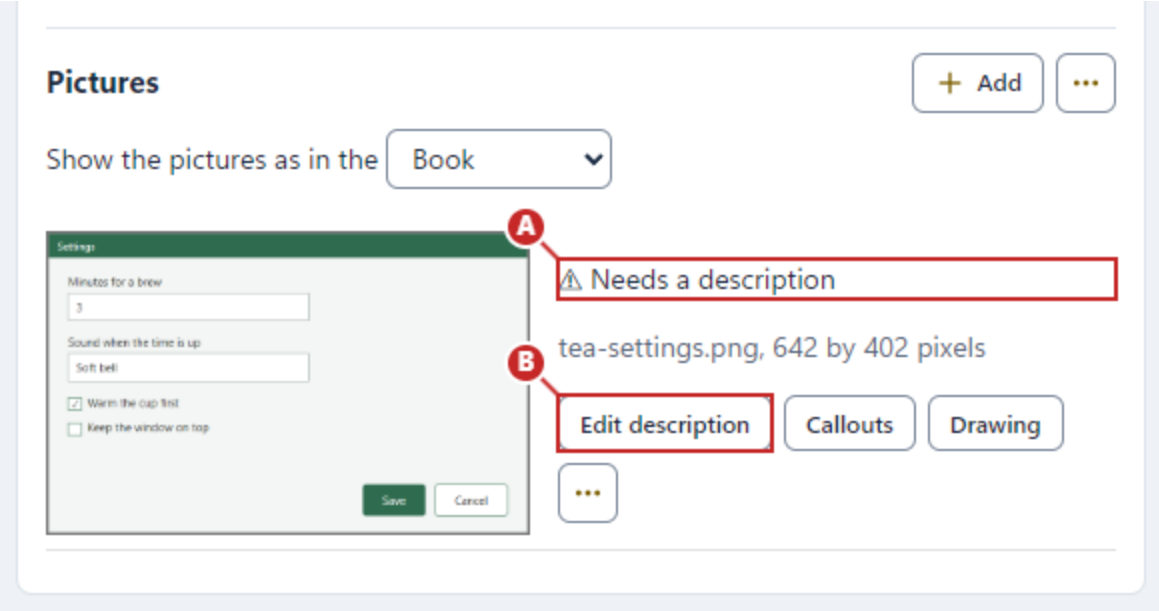
The pictures in this chapter show OpenExplainer working on a made-up book about brewing tea. The book is about a made-up program called **Tea Timer**. These tools work on a picture of any program, and a made-up one keeps anything private off the screen.

5.1 Describing a picture for readers who cannot see it

A **description** says what a picture shows, in words. A screen reader reads it aloud to a reader who cannot see the picture. OpenExplainer never fills one in for you: it asks, and you write it.

To describe a picture:

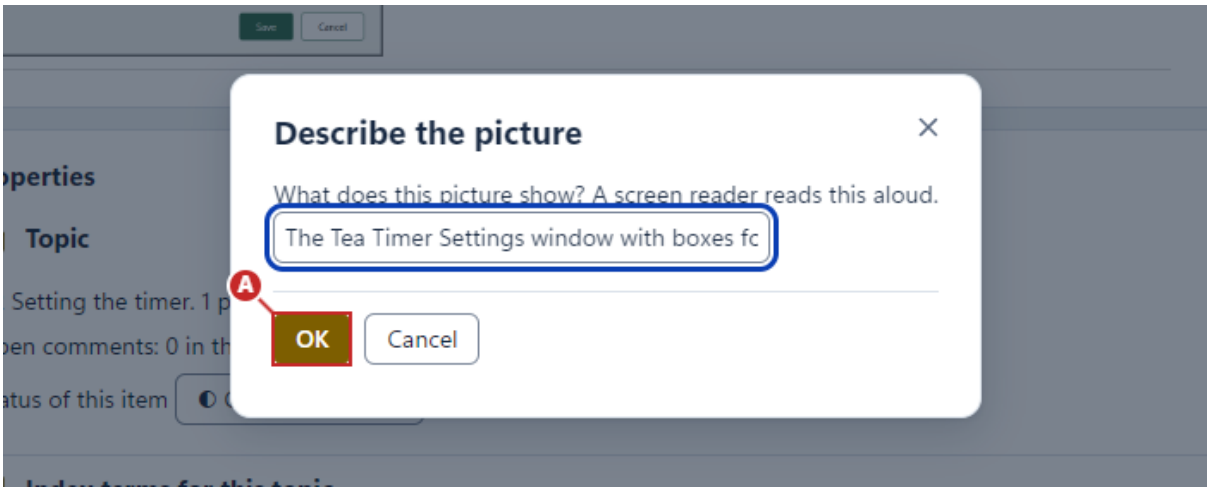
1. Find the picture in the list called *Pictures*, under the page. A picture with no description says *Needs a description*.



A. Needs a description: A picture with no description is marked in words.

B. Edit description: Opens the window where you write it.

2. Click **Edit description**.
3. Type what the picture shows.
4. Click **OK**.



A. OK: Keeps the description.

Result: The list now shows your words after the word *Description*, in place of the warning.

> ✎ **DESIGN TIP**
>
> A picture with no description is the most common accessibility problem in a book. The book check (Chapter 6, page 68) lists every one of them.

Writing a good description

Say what the picture shows and what matters about it for the task, in one or two sentences. Name the program and the parts that the reader will use. Do not start with *a picture of* or *a screenshot of*, because a screen reader already says that it is a picture.

The words of the step are the place for what to do, so the description should not repeat them. The callouts (next topic) name the individual parts, so the description does not need to list every button either.

For example, the picture of Tea Timer in this chapter is described like this: *Tea Timer with a list of teas on the left, a large clock showing 3:00, and the buttons Start, Stop and Reset under it.*

Where descriptions go

The description is the alternative text of the picture in every output. That means the print preview, the PDF, the Word file, the help site and Markdown. The PDF is tagged, so a screen reader finds each picture as a figure. The words on any marks you draw (topic 5.4, page 65) are added to it. In a short guide, a picture with no description of its own uses the words of its step.

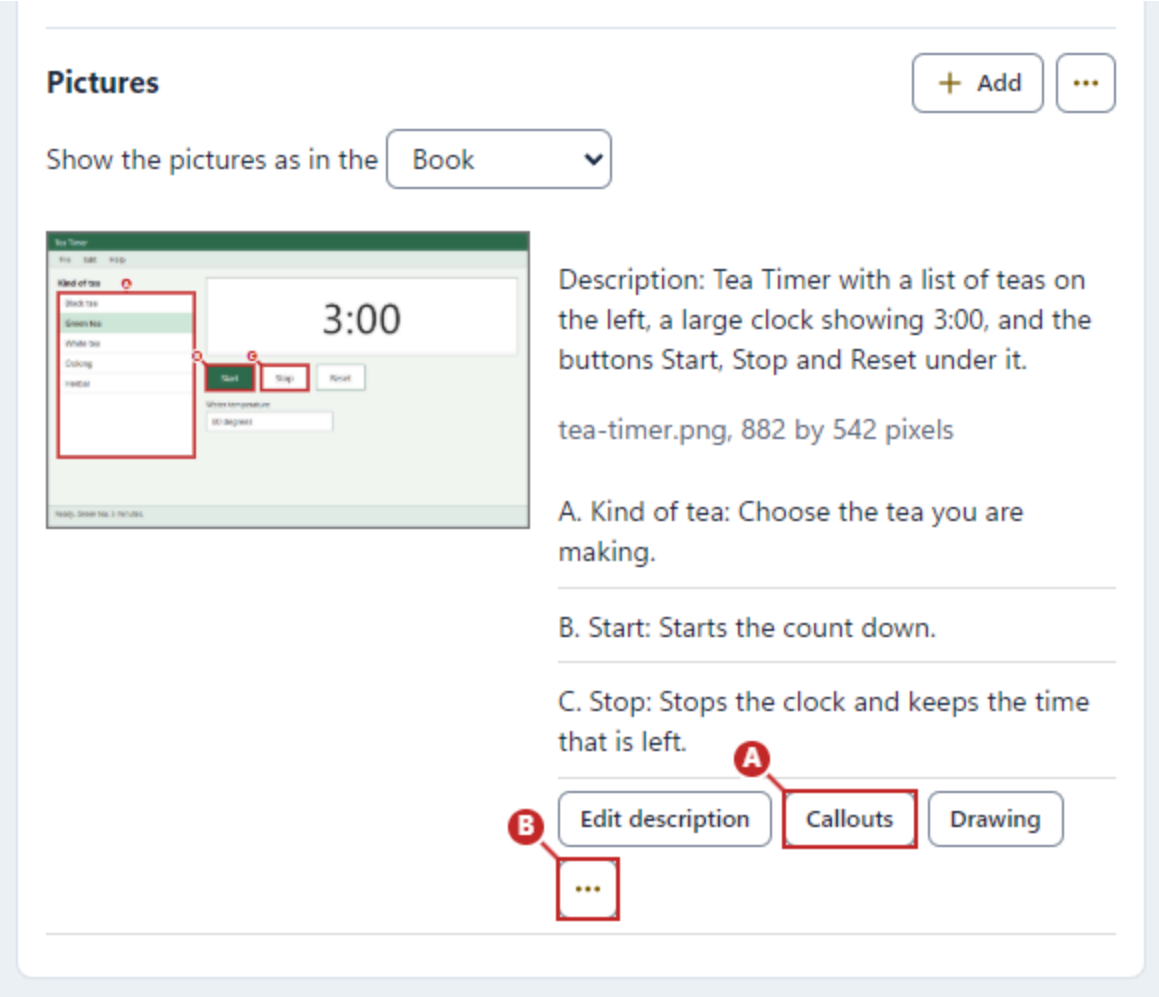
5.2 Pointing at part of a picture

A **callout** points at one part of a picture. It is drawn as a thin red outline around the part, a line, and a red marker with a white **letter**. Under the picture, a list says what each letter means: its letter, its name and a sentence, such as *A. Start: Starts the count down.*

Letters belong to **pictures**, and numbers belong to **steps**. Each picture has its own A, B and C, and a procedure's numbers are never letters. You never type a letter: they follow the order of the callouts, and they change by themselves when you move or delete one.

To add a callout to a picture:

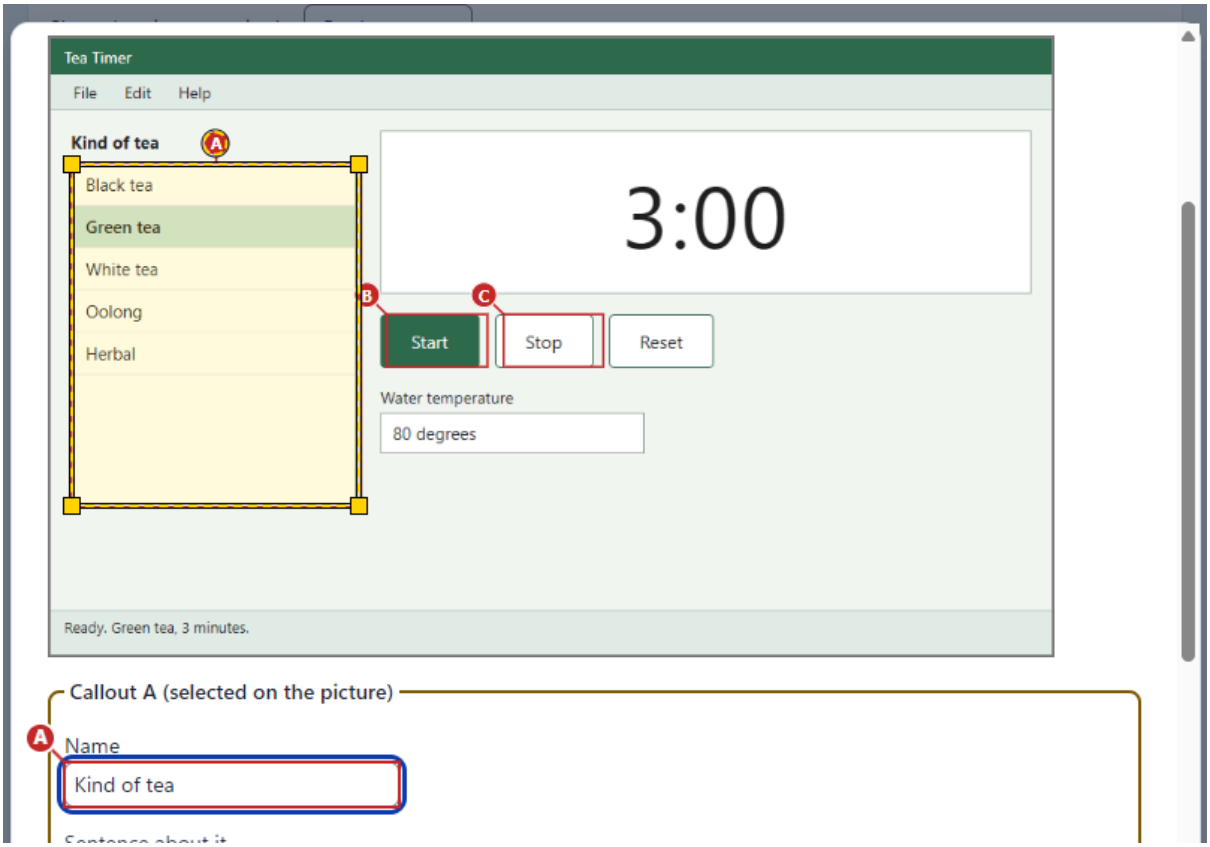
1. Click **Callouts** under the picture in the list (A in the picture).



A. Callouts: Opens the window where you add and place callouts.

B. The ... button: Holds Picture view, Draft callout text, Replace the picture file and Remove picture.

2. Click **Add a callout**.

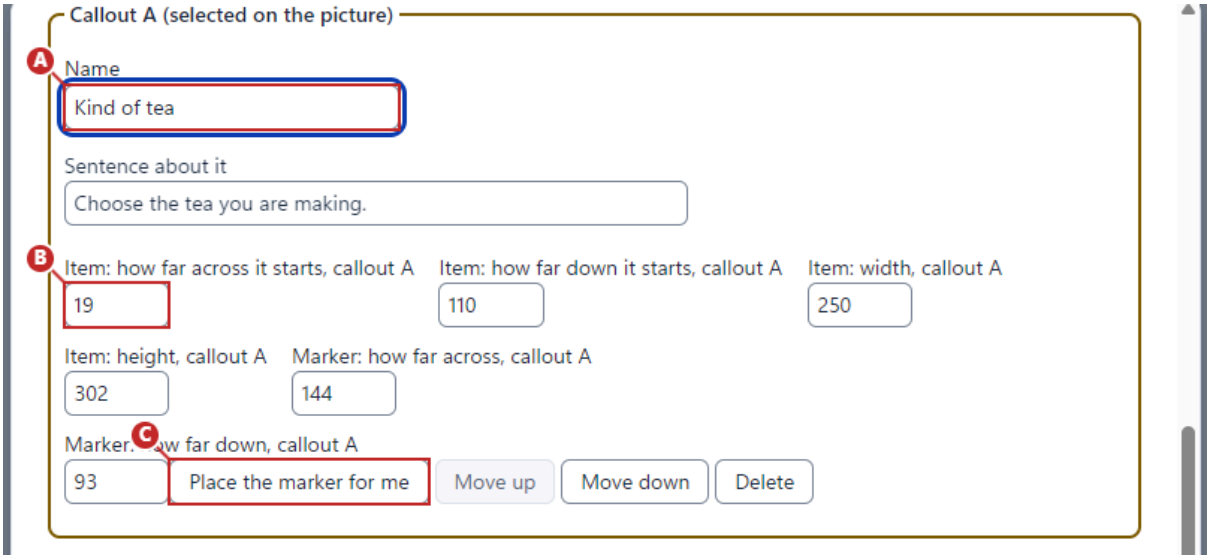


A. Callout A: The one being edited, with its name and its numbers under the sample.

Result: A group of boxes for callout A appears, and an outline appears on the sample.

3. Type the name of the part and a sentence about it.
4. Place the outline on the part (see below).

5. Click **Save the callouts**.



A. Name: What the part is called.

B. Item: how far across it starts: The outline's numbers are counted in pixels from the picture's top left corner.

C. Place the marker for me: Puts the marker next to the outline.

Result: The callouts appear on the picture, and the entries appear under it.

> ✳ **BRIGHT IDEA**
>
> A callout with no name or no sentence is shown in words as *needs a name* or *needs a description*, never hidden. The book check lists them too. If the built-in model is on, **Draft callout text** can suggest the sentences (Chapter 8, page 91), and you review each one.

Placing a callout

There are two ways, and you can mix them. The boxes hold the exact numbers. They say where the part starts: how far across and how far down. They also say how wide and high it is. All of it is counted in pixels from the top left corner of the picture. A second pair of boxes holds the place of the marker. **Place the marker for me** puts it next to the part.

Or point on the sample. Drag on an empty place to draw the outline. Drag the outline to move it, and drag a corner to resize it. Drag the marker to move only the marker. Press Escape during a drag to put everything back. Whatever you do on the sample fills in the boxes.

With the keyboard, Tab to the outline or the marker on the sample. An arrow key moves it by 1 pixel, and Shift with an arrow moves it by 10. Control with an arrow makes the outline wider, taller, narrower or shorter. The new place is said in words after a short pause, so a screen reader reports it.

Other buttons in the window

Each callout has **Move up**, **Move down** and **Delete**.

Make a callout from the click is on only for a picture taken click by click. It is on only when the clicked item has no callout yet. Such a picture already gets callout A by itself (Chapter 4, page 42).

If the click was on the window itself, callout A is a small box round the click with no name. That way it does not cover the whole picture. Give it a name and move it onto the right part.

A mistake, such as a word where a number belongs, is explained next to the box. The window stays open until you put it right.

Nothing is kept until you click **Save the callouts**. **Cancel** or Escape leaves everything as it was.

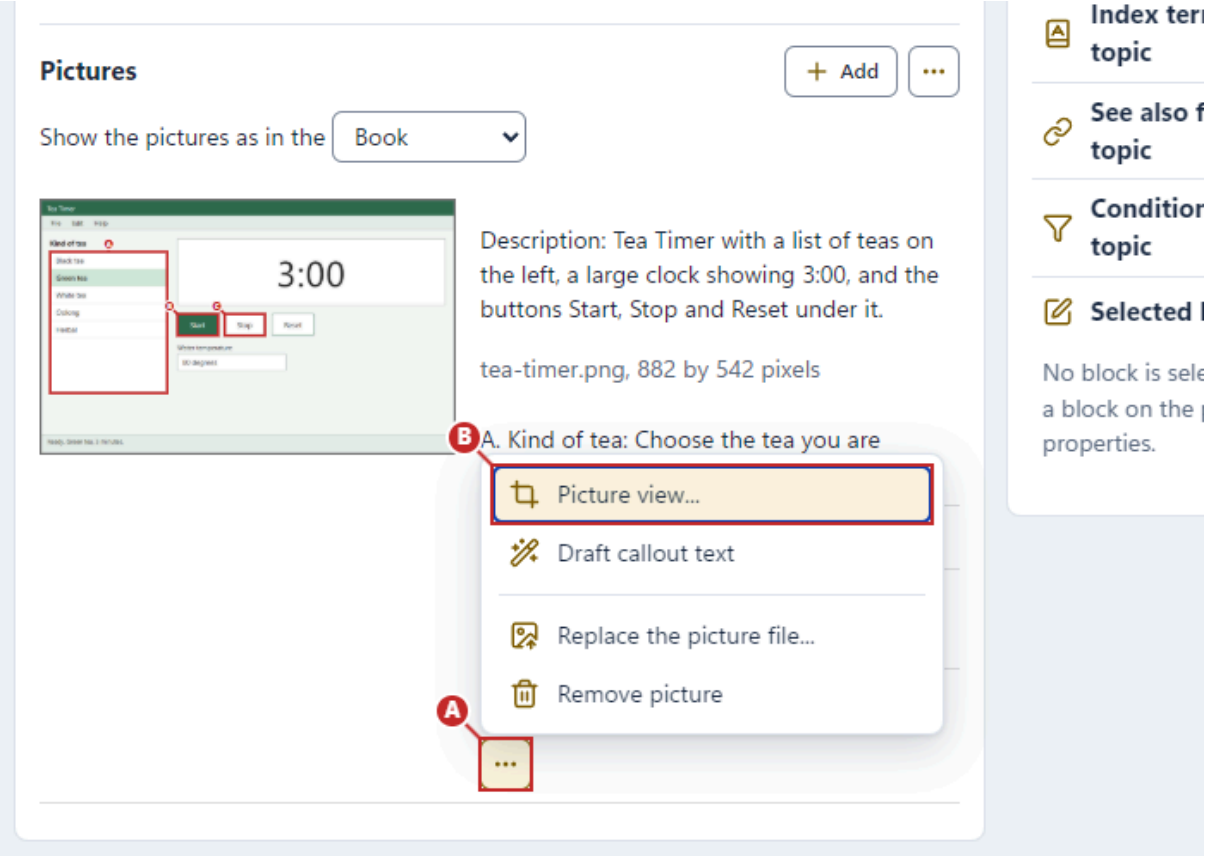
5.3 Showing only part of a picture

A whole window is often too wide to read once it is printed, because the writing in it shrinks. A **view** chooses which part of the picture is shown in the book. The picture file itself is **never cut**. You can change the view or take it off at any time. The callouts and the click circle always line up with what is shown.

> ϟ **POWER TIP**
>
> Open the Print preview tab and look at the notes for the writer above it. It lists every picture that will print too small, so you know where a view would help.

To show only part of a picture:

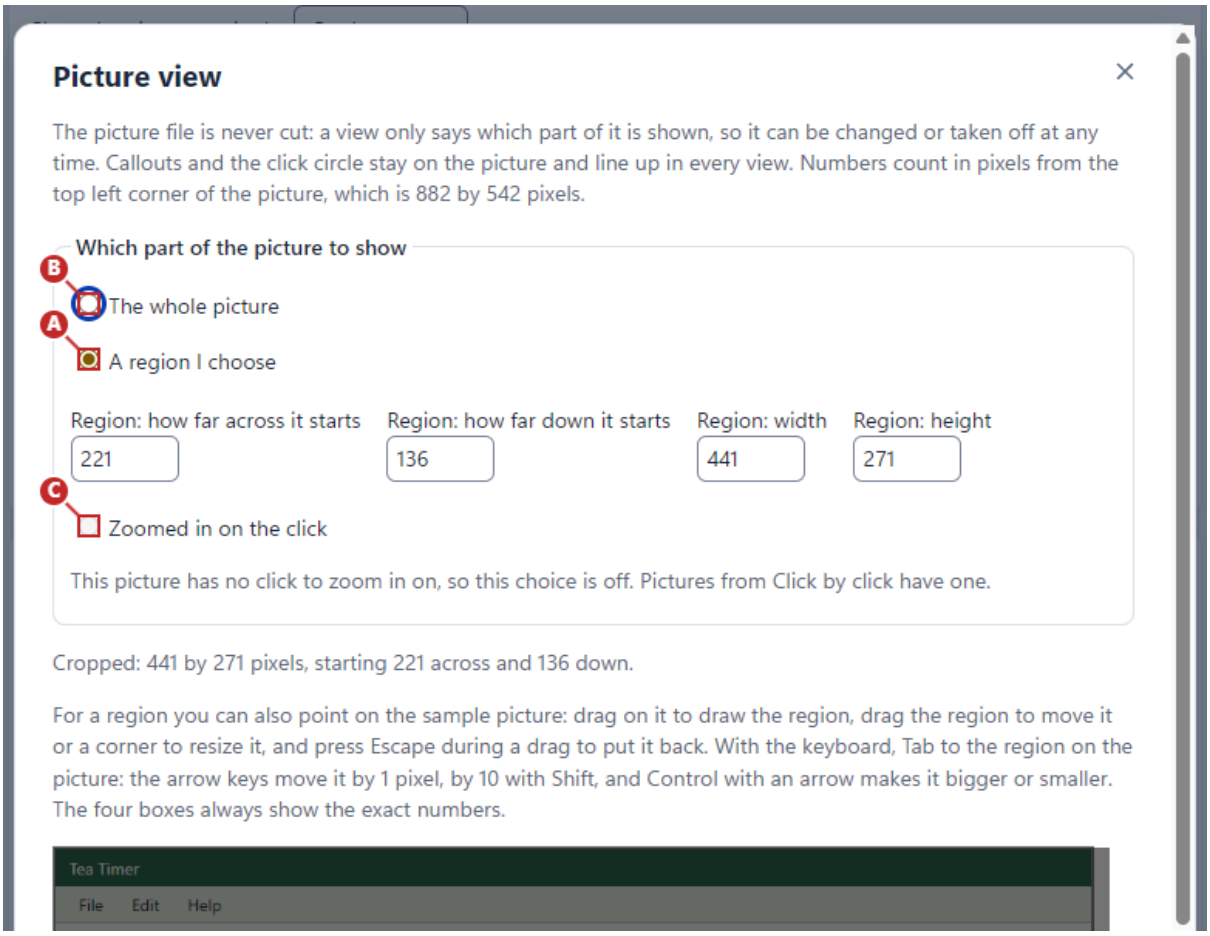
1. Click the **...** button under the picture in the list (A), and choose **Picture view...** (B).



A. The ... button: Holds the commands for one picture.

B. Picture view...: Opens the window that chooses which part of the picture is shown.

2. Choose **A region I choose**.



A. A region I choose: Shows only the part you give.

B. The whole picture: How every picture starts.

C. Zoomed in on the click: Off here, because this picture has no click.

Result: Four boxes appear for the region, and a bright outline shows it on the sample. The rest of the sample is dimmed.

3. Type where the region starts and how wide and high it is, or drag it on the sample.
4. Click **Save the view**.

   Result: The picture in the list and in the book now shows only that part.

The three choices

The whole picture is how every picture starts.

A region I choose shows the part you give. It is counted in pixels from the top left corner, and it starts at the middle half of the picture.

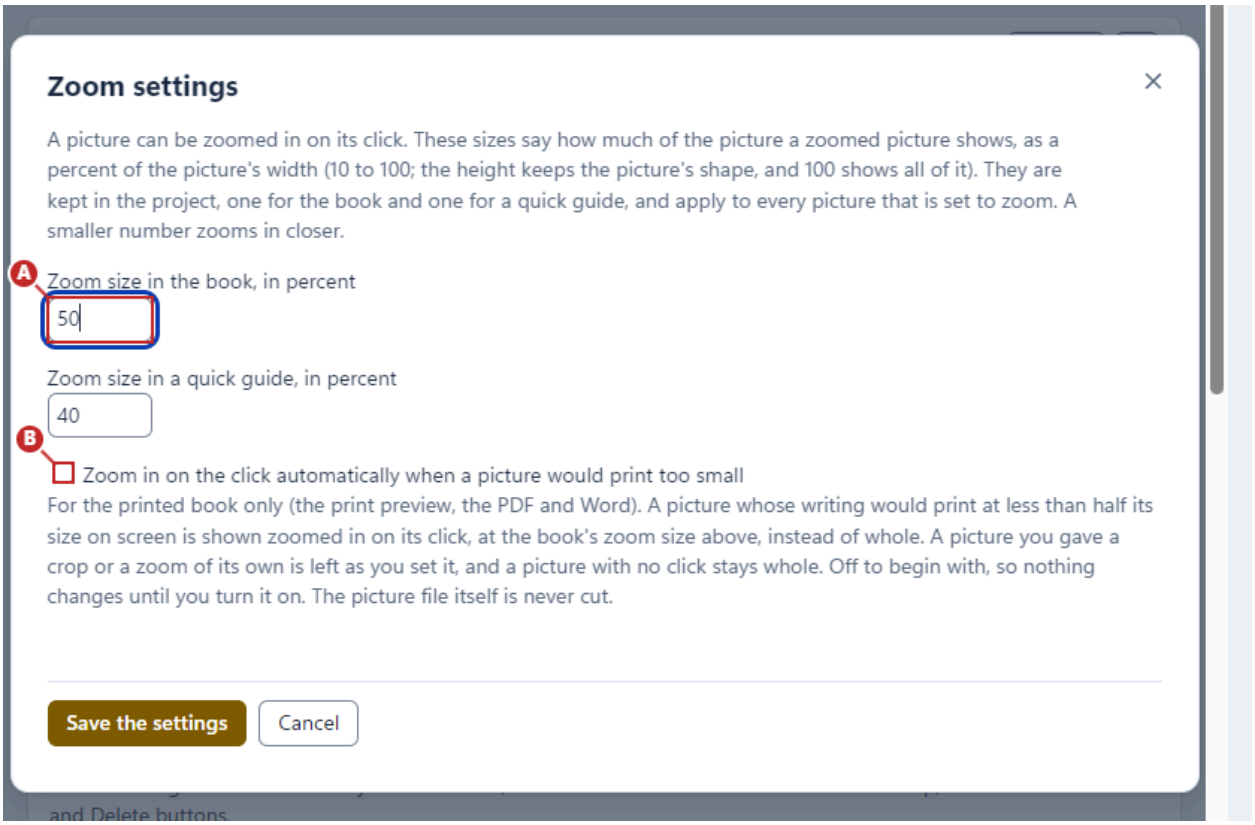
Zoomed in on the click is for pictures taken click by click. The part is centred on the click, and its size is a project setting (see below). It is switched off, with the reason, for a picture that has no click.

A line under the picture in the list says the view in words, for example *Picture view: cropped to 520 by 230 pixels, starting 0 across and 30 down*. A callout that is wholly outside the part shown is not drawn, and its entry says so in words.

Zoom settings and automatic zoom

On the **Book** menu, **Zoom settings…** holds the width of a zoomed part as a percent of the picture's width. There is one for the book and one for a quick guide.

It also has the box **Zoom in on the click automatically when a picture would print too small**. When it is ticked, the PDF and the Word file zoom in on the click by themselves. This applies to any click picture that would print below half its size on screen. Your own crop or zoom always wins over it.



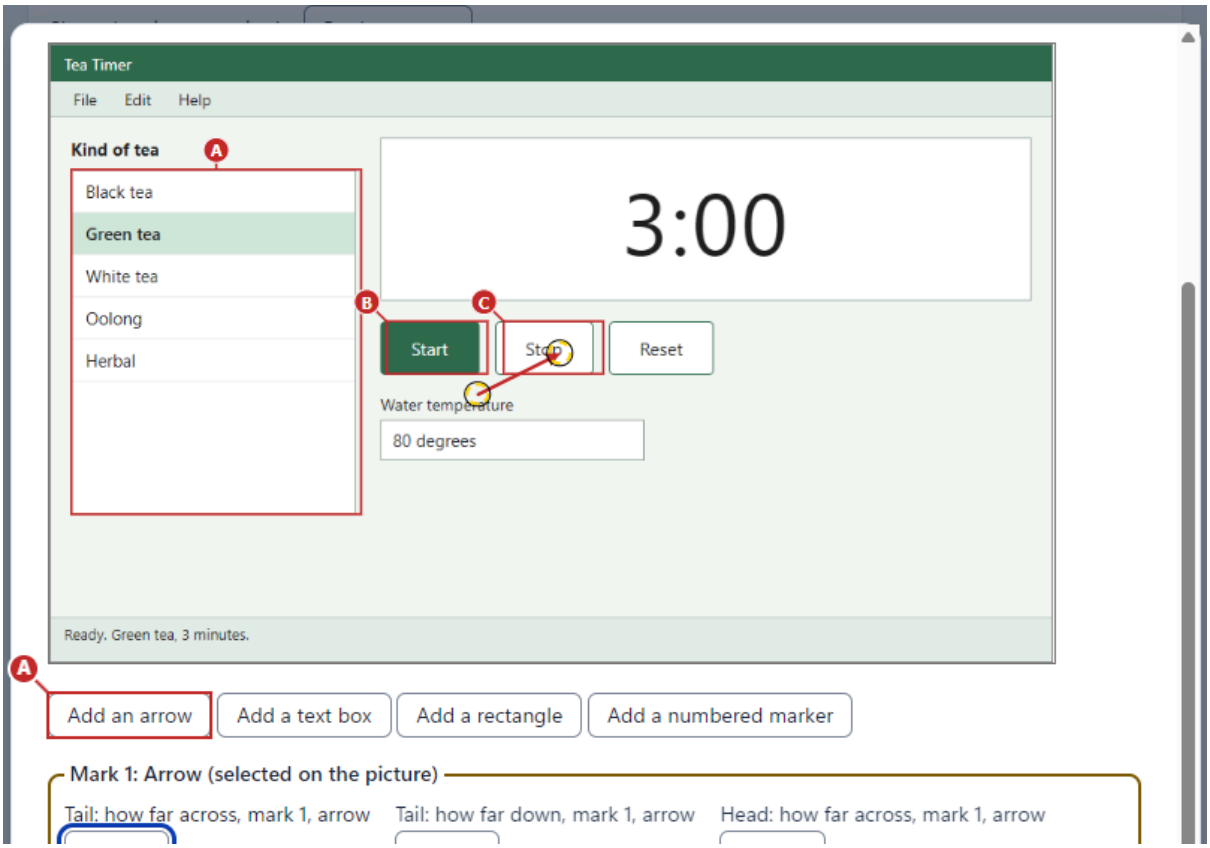
A. Zoom size in the book: How much of the picture a zoomed picture shows, as a percent of its width.
B. Zoom in on the click automatically: For the printed book only.

5.4 Drawing marks on a picture

A **mark** is something you draw on a picture to help the reader: an **arrow**, a **text box**, a **rectangle** or a **numbered marker**. Marks are kept beside the picture and drawn over it when the book is made, under the lettered callouts. The picture itself is never changed, so a mark can be moved, reworded or removed later.

To draw an arrow on a picture:

1. Click **Drawing** under the picture in the list.
2. Click **Add an arrow**.

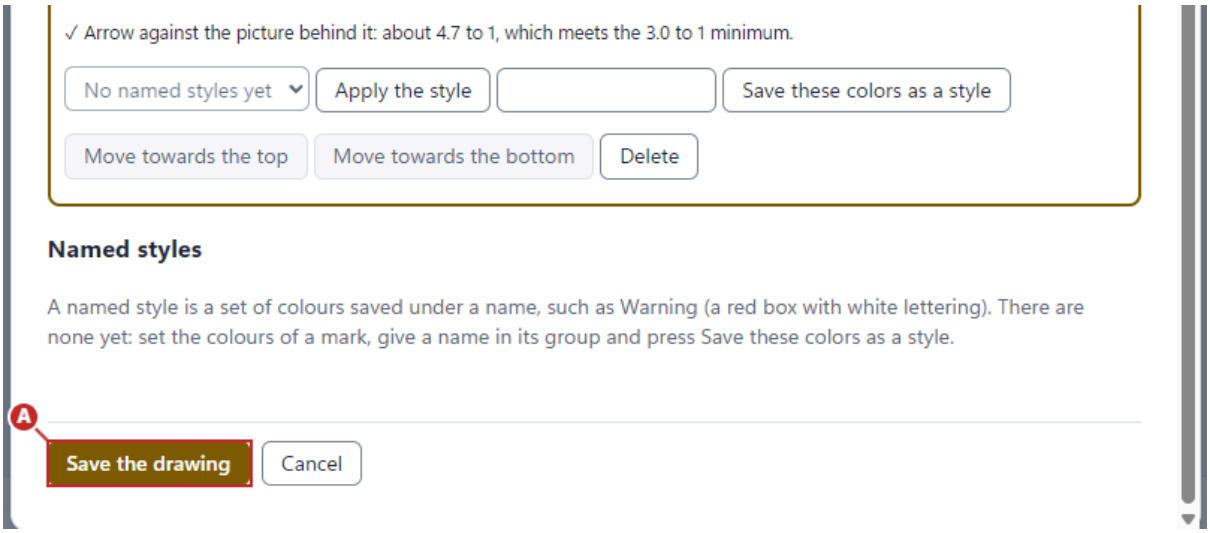


A. Add an arrow: Puts an arrow on the picture. The text box, rectangle and numbered marker are beside it.

Result: The arrow appears on the sample, and its boxes appear below.

3. Fill in the arrow's boxes, or drag it on the sample.
4. Say in the alternative text box what the arrow means for a reader who cannot see it.

5. Click **Save the drawing**.



A. Save the drawing: Keeps the marks. Nothing is kept until you press it.

Result: The arrow is drawn on the picture in the list, and the list says how many marks the picture has.

Which mark to use

An **arrow** points at something from a distance, and has a tail, a head and a thickness. A **text box** puts a few words right on the picture. A **rectangle** outlines or shades an area, and can have a see-through fill. A **numbered marker** is a small circle with a label of one to three letters or digits, for a sequence of things in one picture.

Use a **callout** (topic 5.2, page 57) when a part needs a name and a sentence in the list under the picture. Use a **mark** when you only need to point, or when the words belong on the picture itself.

Colours, contrast and alternative text

Choose each colour from ten ready-made swatches: Red, Orange, Yellow, Green, Blue, Purple, Black, White, Dark gray and Light gray. Or type six hex digits such as C62828.

As soon as you choose a colour, the window says in words how well it can be seen, with a tick or a warning sign. Text against its fill needs 4.5 to 1. An arrow, outline or marker against the picture behind it needs 3 to 1. The check only **warns**. You can keep any colour you like, and the check never stops you.

Every mark has a box for alternative text, and a mark with none says so under its boxes. The words of a text box and the labels of markers are added to the picture's description automatically.

If you use the same colours often, choose a mark's colours, type a name and click **Save these colors as a style**. Any other mark can then use **Apply the style**. Marks have an order, and **Move towards the top** and **Move towards the bottom** change which one is drawn over which. Nothing is kept until you click **Save the drawing**.

CHAPTER 6

Checking your work

In This Chapter

- Finding what is still missing
- Words to avoid and consistency
- Sending the book out for review

A book is finished when a reader can use it, not when the last page is written. OpenExplainer has two helpers for the last stage. **Check the book** looks for things worth fixing, and **Words to avoid** keeps your own word choices consistent. A third topic covers sending the book out for review.

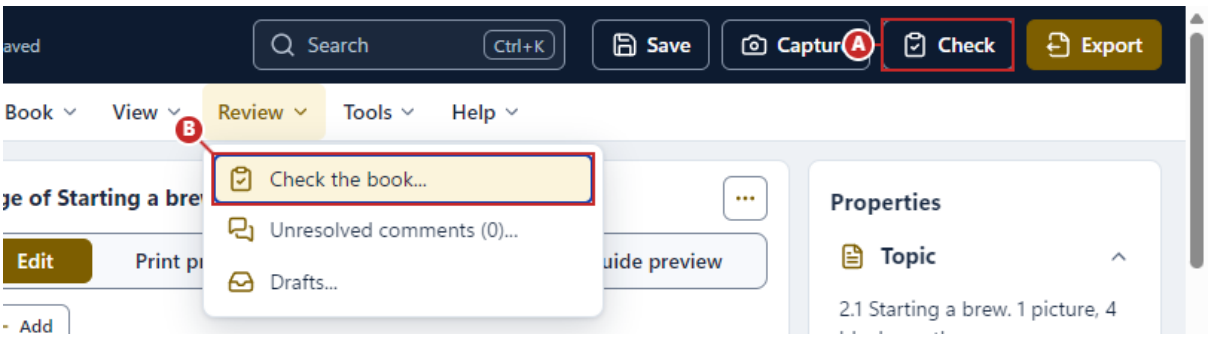
Both helpers are **optional**. They advise and never stop you. Nothing in them blocks saving, exporting or printing. A choice that looks like a mistake, such as a low-contrast colour, may be exactly what you meant.

6.1 Finding what is still missing

Check the book reads the whole project and lists what is worth fixing, in reading order. It works from the project alone, so it needs no AI and no internet. It is on the **Review** menu, and the **Check** button in the navy bar does the same.

To check the book:

1. Click **Check** in the navy bar, or choose **Check the book…** on the **Review** menu (A and B in the picture).



A. Check: The button in the navy bar that does the same.

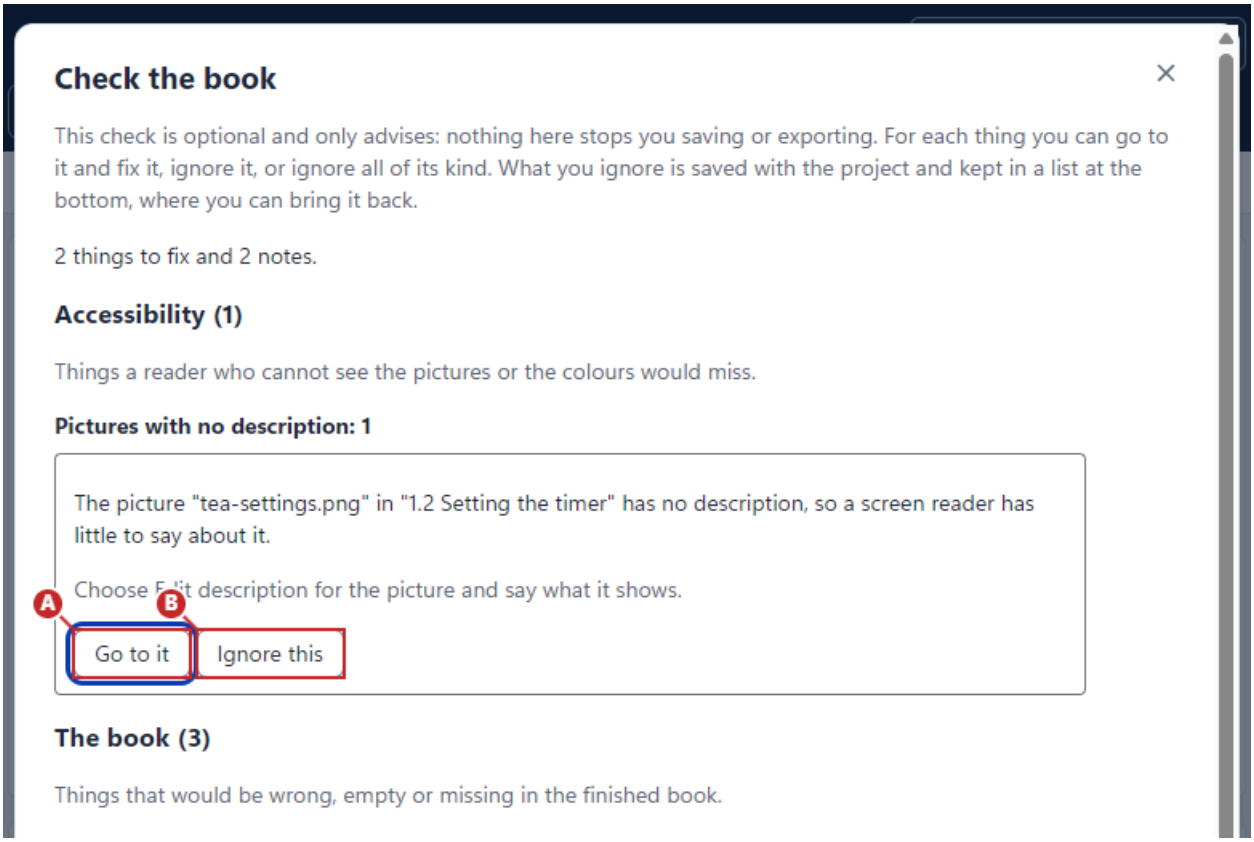
B. Check the book...: Opens the list of things worth fixing.

   Result: A window called Check the book opens. At the top it says in words how many things there are to fix and how many are only notes.

2. Read the list. Each thing is in plain words, with what to do about it, under one of two headings: **Accessibility** or **The book**.

3. For each thing, click **Go to it** to fix it, or **Ignore this** if you mean it to be that way.

   Result: Go to it closes the window, selects the topic and puts the keyboard on the control that fixes it.



A. Go to it: Closes the window and takes you to the thing to fix.
B. Ignore this: Moves the item to the Ignored list.

> **TIME SAVER**
> Run the check once when the pages are written and the pictures are in, and again before you export. An unfinished book will show many items. That is normal: use the list as a to-do list.

What it finds

Under **Accessibility**, the check lists a picture on a page with no description and an arrow or rectangle with no alternative text. It lists marks whose colours are too close together and a callout with no name or no sentence. As a note, it lists a picture that will print too small to read.

Under **The book**, the check lists a topic with nothing on its page, an empty block and a step with no words. It lists a picture whose file is missing from the project. It lists a variable that is used in the book but has no value, so that its name in curly brackets would print. As notes, it lists a picture that is in a topic but not on its page, and a variable with a value that nothing uses.

Some items are **notes**, not things to fix. They are labelled in words, so nothing depends on colour.

More for a bigger book

Under **The book**, the check also lists a topic with no title. It lists a link or See also entry that goes to a topic that is not in the book. It also lists a snippet block whose snippet has gone.

As notes, it lists topics in one chapter with the same title, and topics whose pages say exactly the same words (the check suggests a snippet). It also lists snippets that no topic uses or that have nothing in them, and conditions that nothing carries.

Under **Accessibility**, it lists a link whose words say nothing about where it goes, such as *here*. It also lists an accent colour too light for text on white, and a logo with no description.

Ignoring things

Ignore this moves one item to an **Ignored** list at the bottom of the window. When a kind has several items, **Ignore all *N* of these** moves them all. **Stop ignoring** brings an item back. What you ignore is saved with the project, so it stays ignored next time.

You can also run the check from inside the Export window, with **Check the book first (optional)**. It is never required.

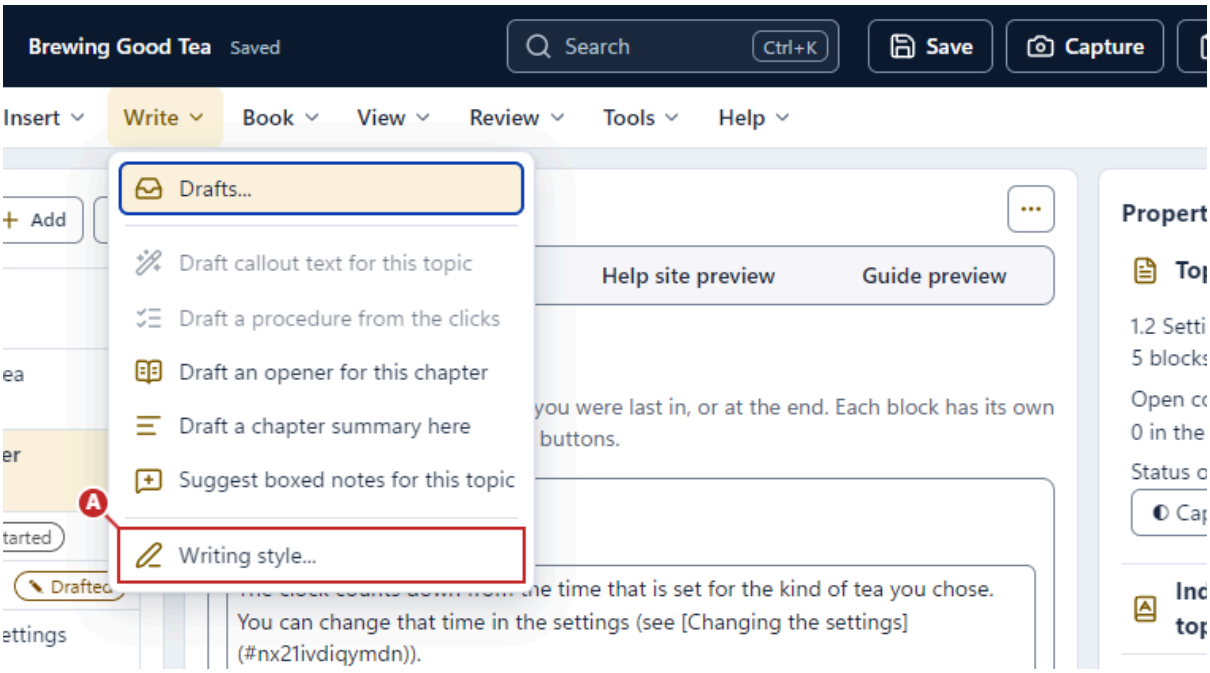
When most of the things to fix are of one kind, the line at the top says so, for example *35 of the things to fix are one kind: Topics with nothing on their page.* An unfinished book has many empty topics, and the list below groups them by kind.

6.2 Words to avoid and consistency

OpenExplainer can keep your wording consistent without any AI. The **Writing style…** command on the **Write** menu opens a window with two boxes.

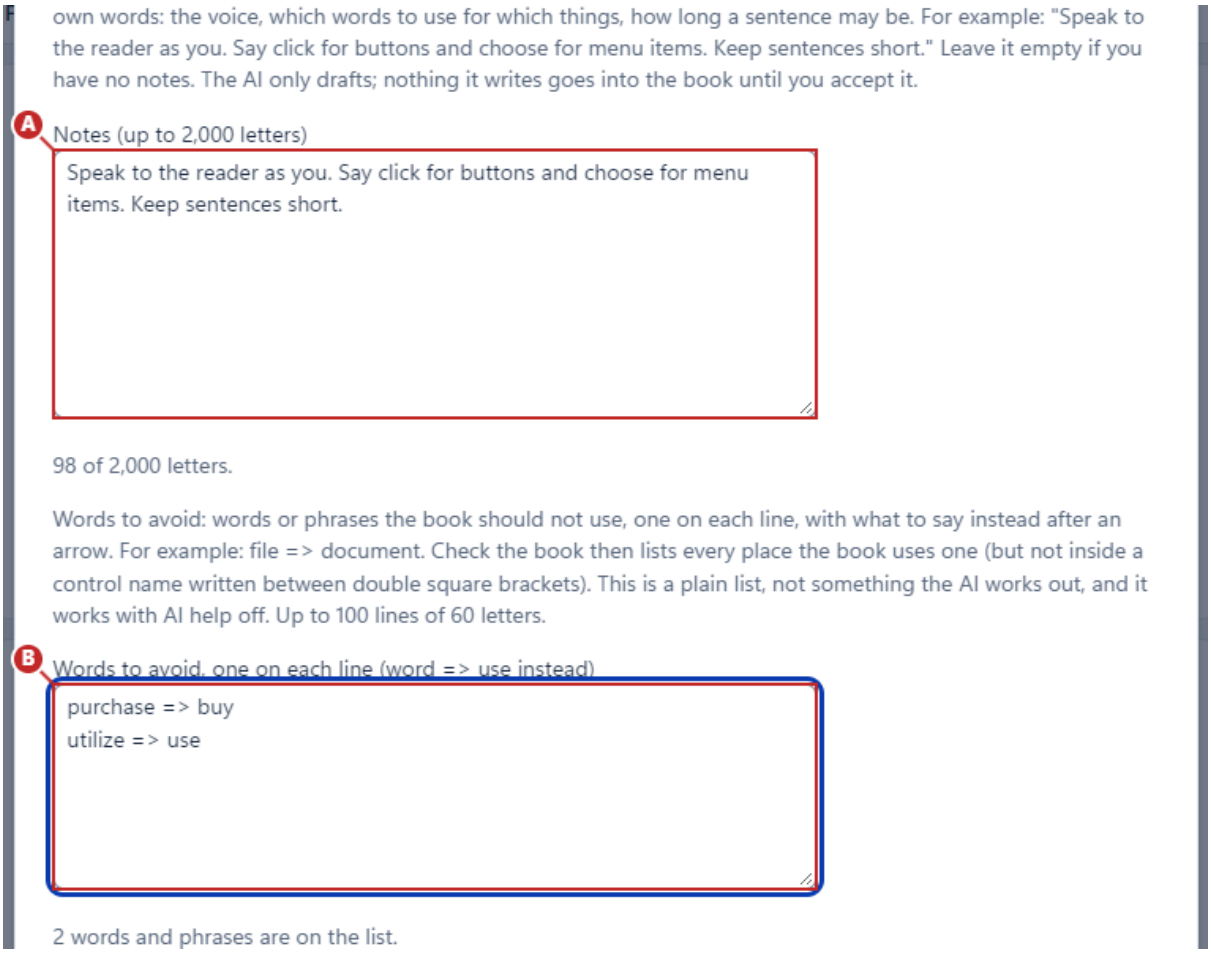
To make a list of words to avoid:

1. On the **Write** menu, choose **Writing style...**.



A. Writing style...: Opens the window for notes to the AI and the words to avoid.

2. Type one word or phrase on each line, in the box called **Words to avoid**.



A. Notes: For the built-in AI.

B. Words to avoid: One word or phrase on each line, with what to use instead after an arrow.

3. Click **Save**.

   Result: The list is saved with the project, and the book check starts using it.

After an arrow (=>) you can say what to use instead, for example: purchase => buy.

> **POWER TIP**
>
> Put your house rules in Words to avoid as you meet them. Examples are a word a reviewer keeps changing and a product name that has to be spelled one way. The list is saved with the project.

The book check lists every place where the book uses one of those words. Each is a note, with the exact words and where they are. The check does **not** look inside a control name written between double square brackets. That name must match what the program shows on the screen.

The Notes box

The first box, **Notes**, is for the built-in AI. Say in your own words how the book should read: the voice, which words to use for which things, how long a sentence may be. The AI reads the notes before it drafts anything (Chapter 8, page 91). With AI help off, the Notes box does nothing, and everything else here still works.

Other consistency notes

When your book has pictures taken click by click, the check can also compare your writing with what the program actually showed. It adds notes for three things.

A control name in double square brackets differs from the name Windows reported when it was clicked. An example is Save as in the text when the program says *Save as…*.

A step starts with a verb that does not suit the kind of control, such as Press for a menu item.

A word or phrase from your Words to avoid list is used.

6.3 Sending the book out for review

A colleague who does not have OpenExplainer can still review your book, in Word. You send a **review copy**; they add comments; you bring their comments back to the right places in your book.

> ✳ **BRIGHT IDEA**
>
> An assistant can do the whole round with its tools: make the review copy, read the returned file and list the comments (Chapter 9, page 97).

Making the review copy

To make a review copy:

1. Click **Export** in the navy bar, or choose **Export…** on the **File** menu.
2. Choose **Word review copy**.
3. Click **Export** in the window.

   Result: A Word file is saved. It is the whole book, as in the Word export, with your open comments in it as real Word comments. A line at the top asks reviewers to comment rather than edit.

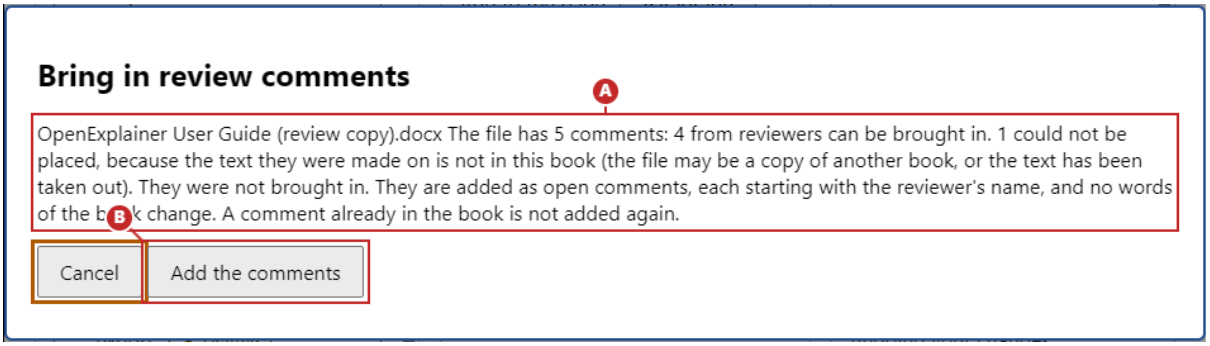
Your working comments (topic 3.7, page 34) go into the review copy as Word comments from *The author*, next to the block they are on. Resolved comments are not sent. Each block and each heading is also tied to the book by a hidden bookmark. That is how the comments find their way back.

> **⚠ CAUTION**
> Only comments come back. If a reviewer changes the words in the Word file, those changes are not brought into the book. Ask reviewers to comment rather than edit.

Bringing the comments back

To bring the reviewers' comments back:

1. On the **File** menu, choose **Bring in review comments...**.
2. Choose the file the reviewers sent back.



A. What the file holds: How many comments the file has, how many can be brought in, and how many could not be placed.

B. Add the comments: Adds each comment to the block it was made on, as an open comment that starts with the reviewer's name.

Result: A window says how many comments the file has, how many can be brought in, and how many could not be placed (A in the picture).

3. Click **Add the comments** (B).

Result: Each comment is added to the block it was made on. It is an open comment that starts with the reviewer's name, such as *Pat: Which Save?*. No words of the book change.

The picture of the Bring in review comments window was taken on 1 November 2026, before the new look. It was not taken again for this edition, because the window comes after Windows' own Open window, which was not tried (topic 1.5, page 7).

A comment made on a heading goes to the topic's first block. Your own comments coming back are left as they are. A comment that is already in the book is not added twice, so you can bring the same file in again safely. A comment that cannot be placed, because the text it was made on has gone from the book, is counted and not added.

Then work through them with **Unresolved comments** (topic 3.7, page 34): click **Go to the block**, make the change, and press **Resolve**.

CHAPTER 7

Making your outputs

In This Chapter

- Previewing before you export
- Making a PDF for print
- Making a Word file
- Making a help site
- Exporting Markdown and JSON
- Short step-by-step guides
- Building the index
- Setting the look of your book
- Making an e-book
- Translating your book

The book you write in OpenExplainer is kept in one project file. Every output is made from that file and never changes it. You can make a PDF today, edit a topic tomorrow and make it again.

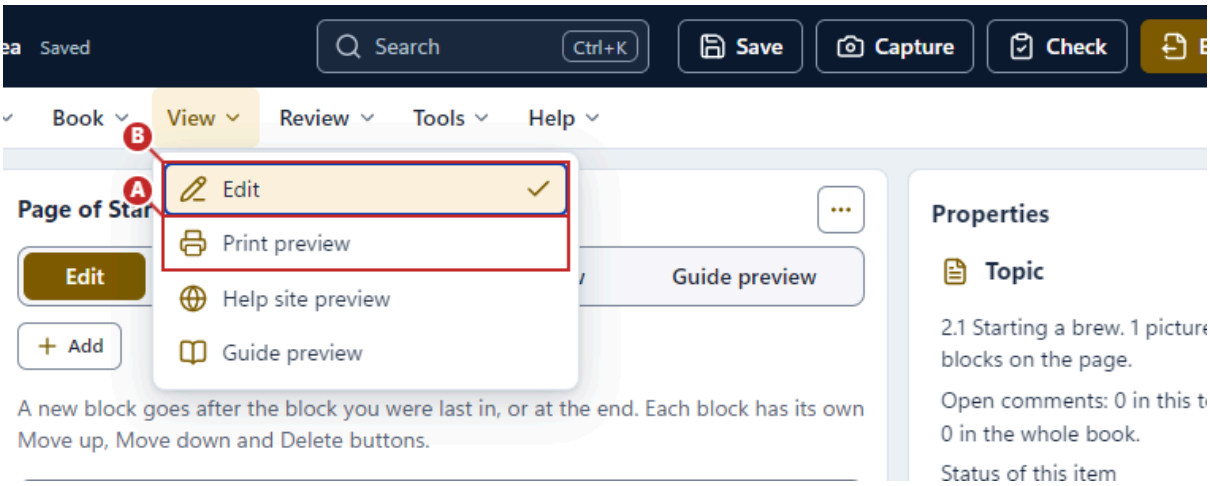
This chapter covers the previews and then each output. The **PDF** and the **Word file** are for paper, and the **help site** is for the web. **Markdown** and **JSON** are for other programs. The chapter also covers the short **guides**, the **index**, the look of the book, the **e-book** and translation.

7.1 Previewing before you export

The page pane has four tabs: **Edit**, **Print preview**, **Help site preview** and **Guide preview**. The preview tabs show your book as it will look, so you can check it without making a file.

To look at the print preview:

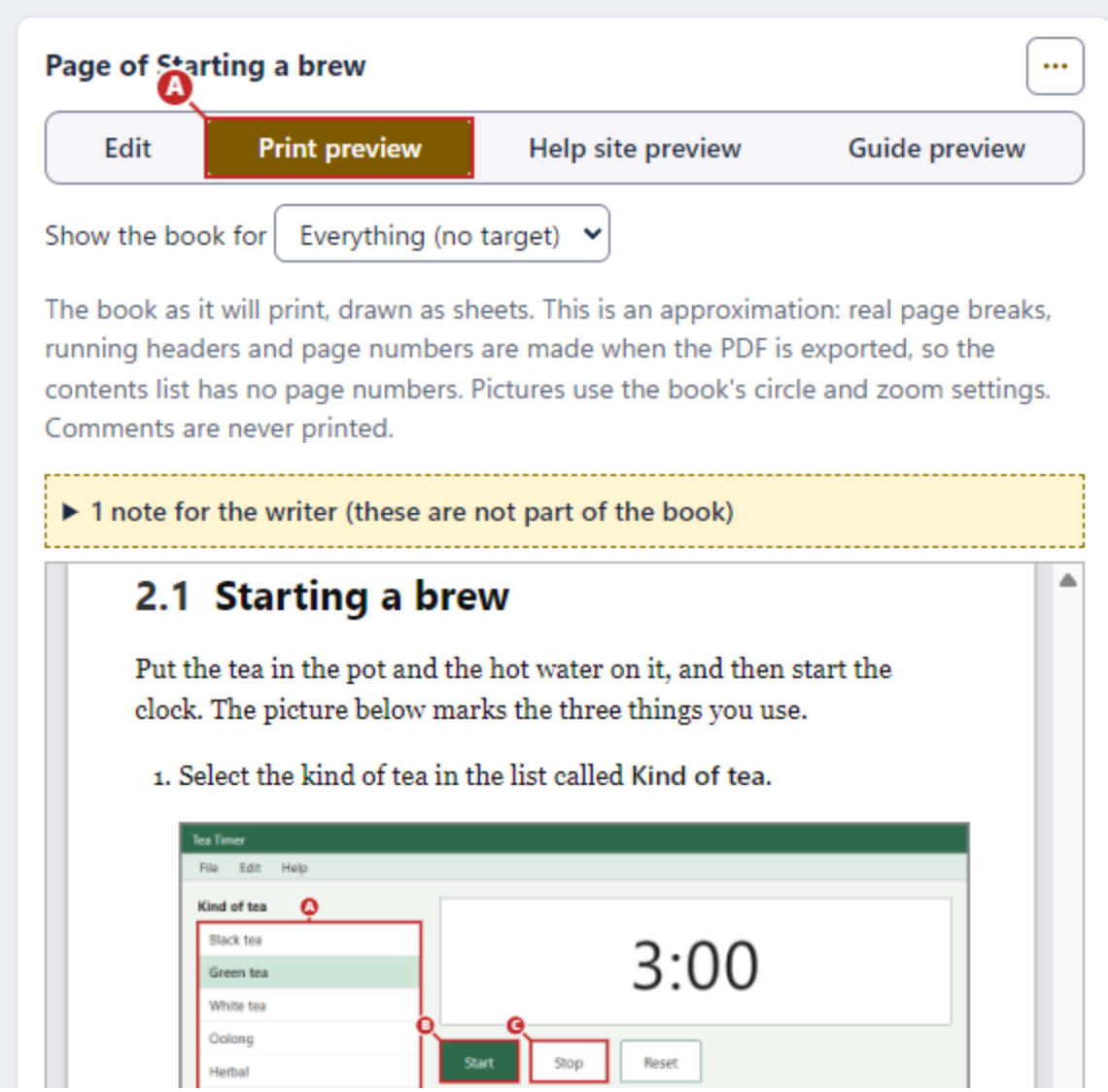
1. Click the **Print preview** tab, or choose **Print preview** on the **View** menu (A in the picture).



A. Print preview: Shows the whole book as it will print.

B. Edit: The tick shows which view is on.

2. Read down the sheets, or follow a link in the contents to jump to a topic.
3. Click the **Edit** tab to go back to writing.



A. Print preview: The tab that is chosen.

The tabs work like any tab list. Left and Right move between them, and Home and End go to the first and last. The preview is real page content, not a picture. A screen reader reads it, and you can reach and scroll its region with the keyboard.

Notes for the writer

Above a preview you may see a line such as *2 notes for the writer (these are not part of the book)*. Open it to see things worth a look. The notes cover a picture whose writing will print too small, a picture with no description, and a callout outside the part that is shown. They also cover a missing picture file. Each note sits next to the picture it is about. An export never contains them.

What the previews are

The **print preview** is an approximation of the printed book. Real page breaks, the running header and page numbers belong to the PDF. The **help site preview** shows one topic as a web page, with the contents tree, the breadcrumbs and the Previous and Next links. The **guide preview** shows one short guide (topic 7.6, page 81).

7.2 Making a PDF for print

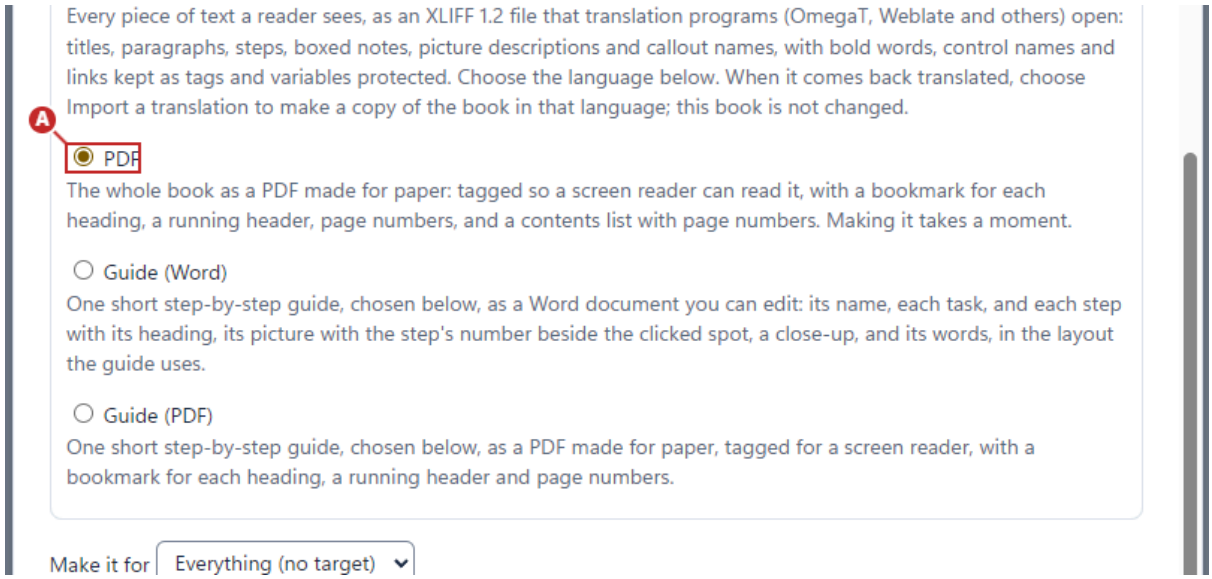
The **PDF** is the main output: a book made for paper. It is letter size unless you choose A4 (topic 7.8, page 85). It has a running header, page numbers and a contents list with page numbers.

To make a PDF:

1. Click **Export** in the navy bar, or choose **Export...** on the **File** menu.

   Result: A window called *Export* opens, with a list of the kinds of export.

2. Choose **PDF** (A in the picture).



A. PDF: The whole book as a PDF made for paper.

3. Click **Export** at the bottom of the window.
4. Choose where to save the file in Windows' Save window.

   Result: The program makes the PDF, which takes a moment, and then says where the file was saved.

> **POWER TIP**
>
> Unless you choose others in Look of the book, the fonts are Georgia for the text and Segoe UI for the headings. They are the fonts installed on your PC, and the PDF embeds the ones it used. A PC without them falls back to Times New Roman and Arial.

What is in it

The PDF has real text, so it can be searched and selected. It is **tagged** (headings, lists, links and figures). It has the book's language, the book's name as its title and a **bookmark** for each heading. Each picture's description is its alternative text.

Each chapter starts on a new page. The title page has no header or page number. Pictures are never split across pages, and a step is kept with its picture where there is room.

The contents list has dotted leaders and page numbers. The program finds the numbers by printing the book more than once. If it cannot read them out of the finished file, the contents is left without numbers, and the export says so. Wrong numbers would be worse.

Comments and notes for the writer are never in it. Nothing is sent anywhere: the PDF is made on your PC by the same browser engine that shows the program.

7.3 Making a Word file

The **Word** export makes one .docx file that you can edit and restyle. Everything in it is a real Word style, not hand-made formatting, so changing a style changes the whole book.

> ✳ **BRIGHT IDEA**
>
> If your publisher or your organisation has its own Word template, give it to your editor with this file. Because everything uses real styles, restyling is a matter of changing the styles.

To make a Word file:

1. Click **Export** in the navy bar, or choose **Export...** on the **File** menu.
2. Choose **Word**.
3. Click **Export** in the window.
4. Choose where to save the .docx file.

   Result: The program says where the file was saved.

The title page holds the book's name and the PRODUCT, VERSION and COMPANY variables. The contents is Word's own table of contents, already filled in with links. It has no page numbers yet, because only Word knows where its pages break: in Word, right-click the contents and choose **Update Field**.

A chapter is Heading 1 and starts a new page, and each topic is one level deeper, so Word's navigation pane works. Chapters that have topics open with an **In This Chapter** list. Each procedure is a numbered list that starts again at 1, with its picture indented under the step and the result line after it.

Pictures are placed in the text, cut to the part shown, with their marks drawn in and their description as alternative text. The callout entries follow each picture as captions, such as *A. Start: Starts the count down.* Each kind of callout box has its own style. It says its kind in words

and shows it by a coloured edge.

The fonts, the accent colour, the paper size, the header, the footer and the language come from **Look of the book** (topic 7.8, page 85). A link to another topic is a link to its heading, with the topic's number after it (topic 3.8, page 35).

There is a running header, which is the book's name unless Look of the book says otherwise. A page number sits in the footer, beside any words you chose for it. Comments are never included, except in the Word review copy (topic 6.3, page 73).

7.4 Making a help site

The **help site** is a folder of web pages, one for each topic. You can put it on a web server or open it from your own PC.

To make a help site:

1. Click **Export** in the navy bar, or choose **Export...** on the **File** menu.
2. Choose **Help site**.
3. Click **Export** in the window.
4. Choose where to save the zip file.
5. Unzip it, and open the file called index.html.

   Result: The first topic opens, with the contents tree on the left.

Each page has a skip link, a header with the book's name and a search box, and the contents tree with the current topic marked. Above the title is a trail of links. Below it are the topic itself, a list of the topics inside it, and Previous and Next links. A **Print this page** button prints only the topic, without the navigation.

The links between pages are ordinary links, so the site works with no script. The only script is for the search box and the Print this page button. Search looks at every word of every topic and ranks title matches first. It says in words how many topics match, and it puts the keyboard on the results. The search also works when the pages are opened straight from a folder.

The pages are made by the same code as the **Help site preview**, so the preview shows what you will get. If the book has an index, every page links to an **Index** page.

For AI assistants

Next to each page there is a **Markdown version** of it, with the same name ending in .md, and each page points to it. The site also has a file called **llms.txt**. It lists the pages of every chapter with a line about each. **llms-full.txt** is the whole book as one Markdown file. An AI assistant that reads your published site can use them. OpenExplainer sends nothing anywhere: these are files in the zip.

7.5 Exporting Markdown and JSON

Markdown and **JSON** are for other programs: a documentation tool that reads Markdown, or an AI assistant that reads the JSON.

Markdown

Choose **Markdown** in the Export window. You get a zip file with one text file, book.md, and the pictures as PNG files. The title is a heading, then a contents list, then each chapter and topic as a heading one level deeper than the one before.

A numbered procedure is a numbered list. A callout box is a quotation that starts with its kind in bold. Each picture is a Markdown picture link whose text is its description, followed by its callout entries as a list.

JSON

Choose **JSON with images**. You get a zip file with book.json and the project's own picture files. The JSON holds the structure of the book, its text, the callouts, the marks, the variables and the settings. It is meant to be read by another program. You can bring it back with **Import an export...** on the **File** menu.

Working comments are left out unless you tick **Include the working comments in the JSON export**. An export that is damaged, or made by a newer version of the program, is refused in plain words and nothing is changed. An imported book counts as unsaved until you save it.

A Markdown export is one file, book.md. A help site has a Markdown file for every page instead (topic 7.4, page 80). The JSON export holds the book as it is now. It does not carry snapshots (topic 2.7, page 21), and a book made for a target is the whole book in JSON (topic 3.9, page 37).

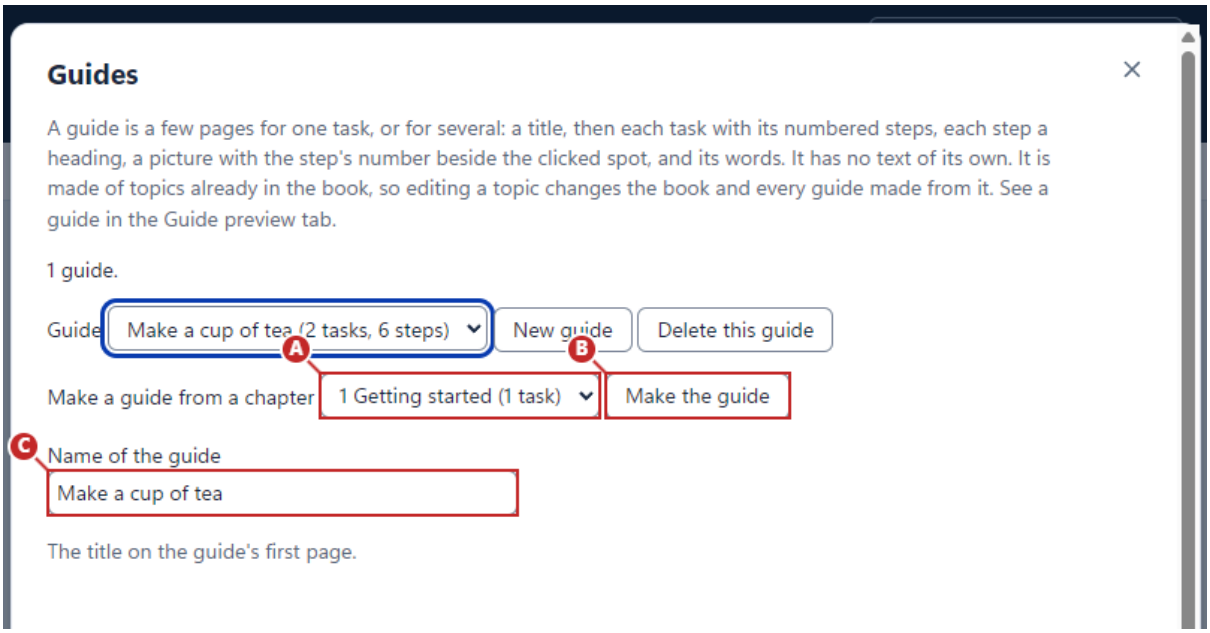
7.6 Short step-by-step guides

A **guide** is a few pages for one task, or for several: the quick handout rather than the book. A guide holds **no text of its own**. It is a list of topics that are already in your book, with a layout and a few choices. So you write each step once, and editing the topic changes the book and every guide made from it.

To make a guide from a chapter:

1. On the **Book** menu, choose **Guides...**.

2. In the list called *Make a guide from a chapter*, select a chapter.



A. Make a guide from a chapter: Chooses the chapter whose topics with steps become the tasks.

B. Make the guide: Makes a new guide from that chapter.

C. Name of the guide: The title on the guide's first page.

3. Click **Make the guide**.

   Result: The chapter's topics that have numbered steps become the guide's tasks, and the chapter's name becomes the guide's name.

4. Change the name, the layout and the tasks if you want to.

5. Click **Save**.

What a guide takes from the book

From each task a guide takes the numbered steps, with their pictures and result lines, and the boxed notes, in page order. The paragraphs, headings and loose pictures stay in the book. Each step is a heading (*Step 3*), its picture, its words and its result line.

> **ϟ POWER TIP**
>
> In a guide, a picture with no description of its own uses the words of its step. So every picture in a guide has a description.

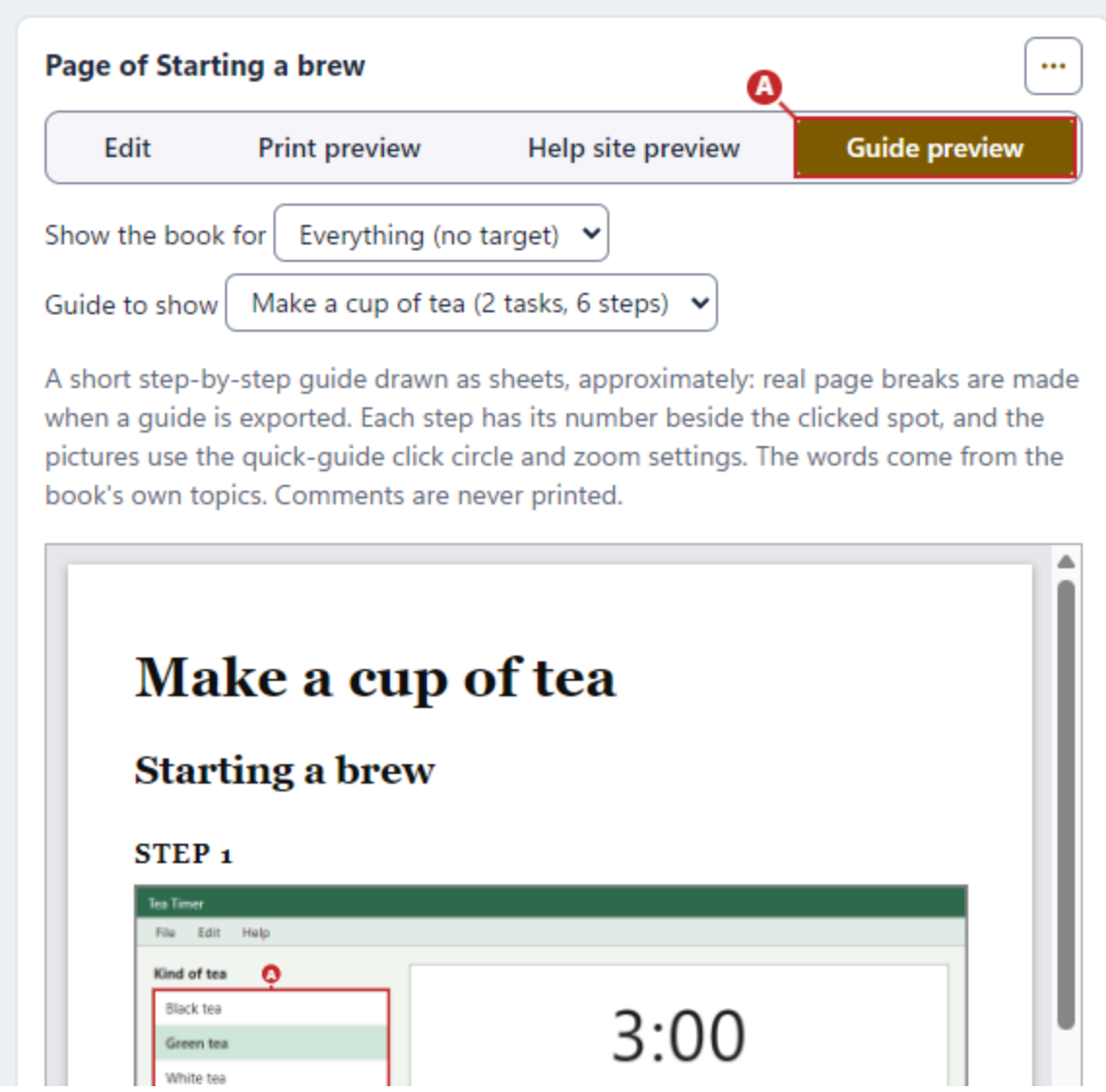
Each step picture has the **step's number** drawn beside the clicked spot. The number sits above the spot and to its left, so that the spot and the button's label stay readable. Beside the picture is a **close-up** of the click. The click circle and the zoom size follow the *Quick guide* settings (Chapter 4, page 42). A guide can turn the close-ups off.

The two layouts

Picture on top puts the picture across the page, the close-up under it, and the words below. **Pictures on the left** stacks the picture and its close-up on the left, with the words beside them. You choose one for the whole guide and can change it for any step.

Seeing and exporting a guide

The **Guide preview** tab shows one guide, chosen from a list. To make a file, choose **Guide (Word)** or **Guide (PDF)** in the Export window and pick the guide. The guide has no cover, no contents and no chapters. Its name is at the top, and each task after the first starts a new page.



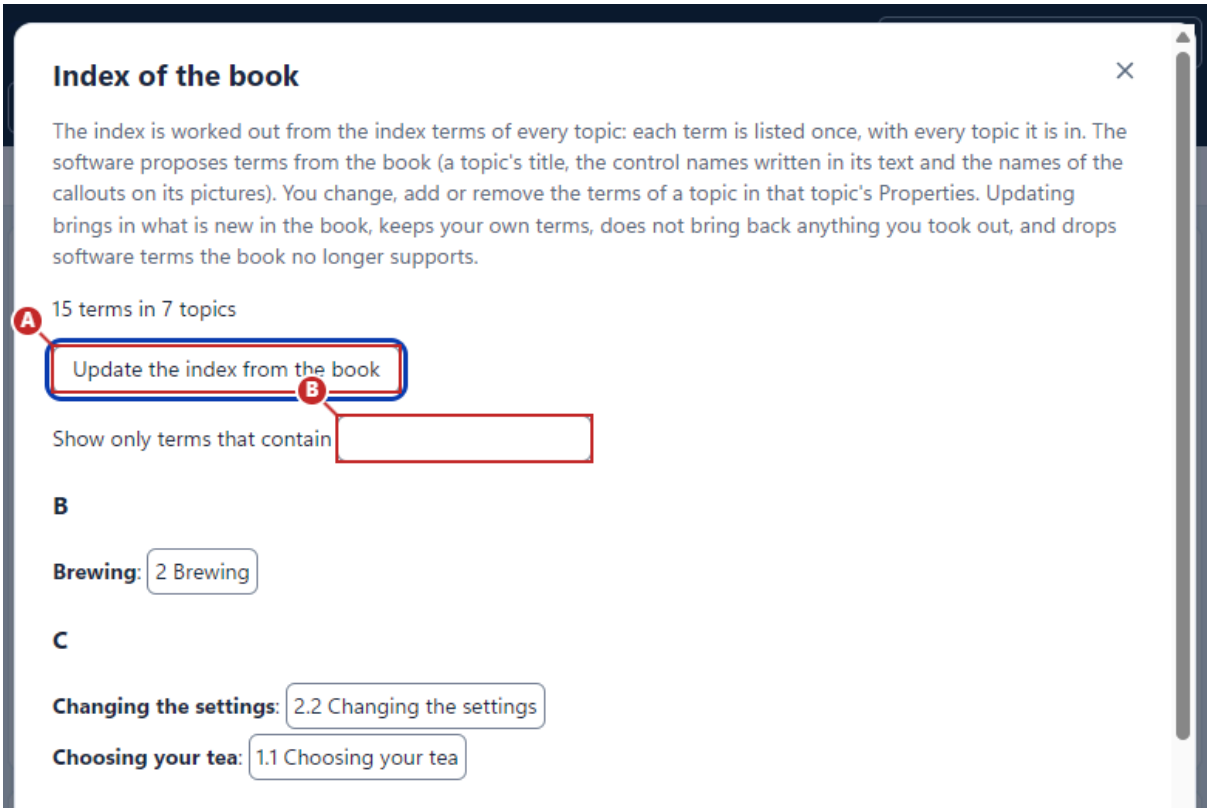
A. Guide preview: The tab that is chosen.

7.7 Building the index

OpenExplainer writes the **first pass** of the index for you, from the book itself, and you edit it. It needs no AI.

To make the first pass of the index:

1. On the **Book** menu, choose **Index...**.
2. Click **Update the index from the book**.



A. Update the index from the book: Makes the first pass, or brings the terms up to date.

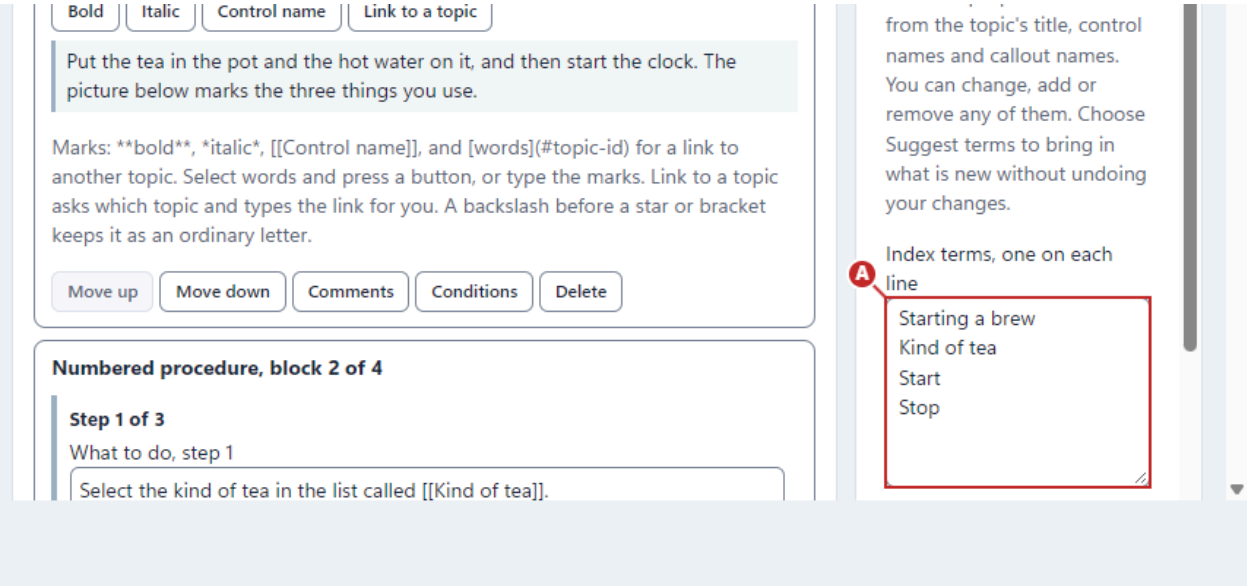
B. Show only terms that contain: Narrows the list as you type.

Result: The window says how many terms it added and took out, and lists the terms under a heading for each letter.

The first pass takes each topic's title and every control name written in double square brackets. It also takes the names of the callouts on the pictures on the page. Headings are not terms, because a sub-heading is rarely something a reader looks up. Each term is listed once, and under it is a button for every topic it is in, which goes to that topic. A box above the list filters the terms.

Editing the terms

The terms belong to topics. To change them, select a topic and open the section **Index terms for this topic** of Properties, on the right. The box **Index terms, one on each line** holds that topic's terms. Change, add or remove any line. **Suggest terms** brings in new ones from the book without bringing back one you took out.



A. Index terms, one on each line: Change, add or remove any line.

Update the index from the book keeps your own terms. It does not bring back a term you removed. It keeps the software's terms that the book still supports and drops the ones it does not.

Where the index appears

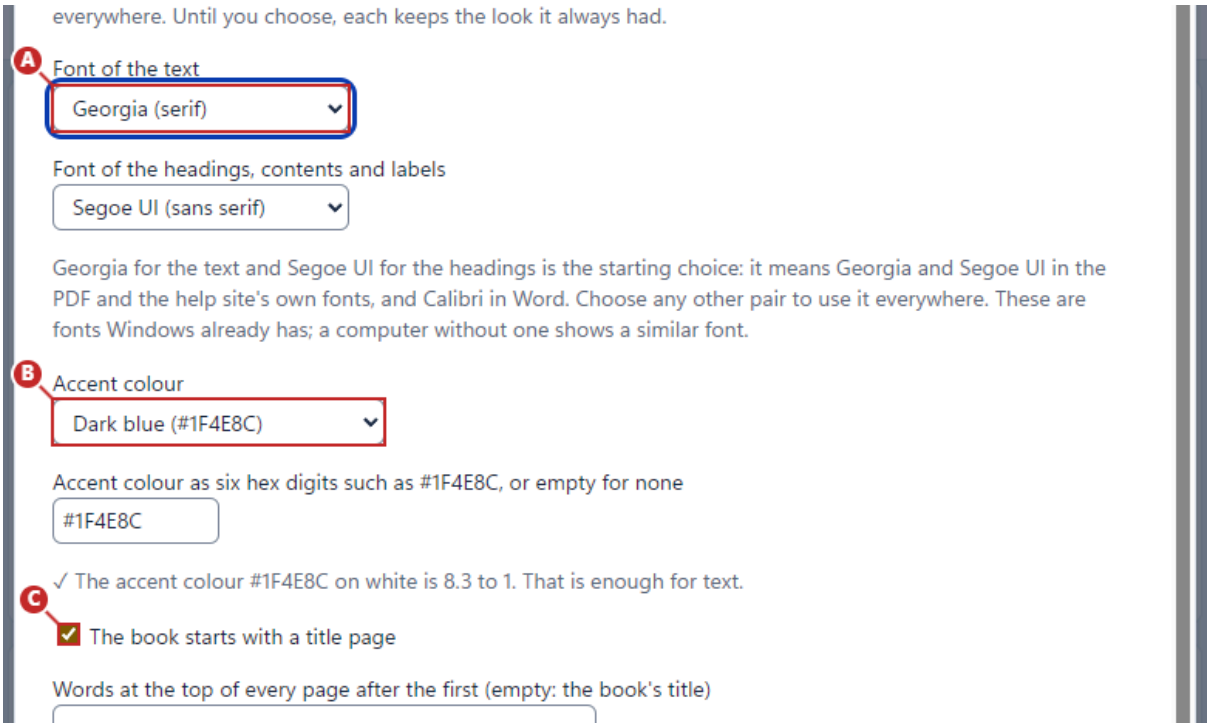
The index is in every output made for reading. In the PDF it is at the end of the book and is the last entry in the contents. Each term has the **page numbers** of its topics. In Word it has an Index heading with a heading for each letter. The e-book and the help site each have an index page, and Markdown has an Index list. A book with no terms has no index.

7.8 Setting the look of your book

Look of the book sets how the PDF, the Word file, the help site and the e-book look, so the book looks the same everywhere. Until you choose, each keeps the look it always had.

To set the look of the book:

1. On the **Book** menu, choose **Look of the book…**.



A. Font of the text: Seven fonts that Windows already has.

B. Accent colour: The colour of headings and links.

C. The book starts with a title page: Ticked to begin with.

Result: A window opens. It holds the fonts, an accent colour, the title page, the words at the top and bottom of each page, the paper size, the language and the logo.

2. Make your choices.
3. Click **Save**.

Result: The look is kept with the project and announced in words.

What you can choose

Fonts. Choose one font for the text and one for the headings, contents and labels. The seven fonts are Georgia, Times New Roman, Cambria, Segoe UI, Calibri, Arial and Verdana. The starting choice is Georgia for the text and Segoe UI for the headings, and Calibri in Word.

Accent colour. The colour of headings and links. Choose one of six dark colours, or type six hex digits such as #1F4E8C. A colour that is too light for text on white gets a warning in words and a symbol. It is still allowed, because the check only advises.

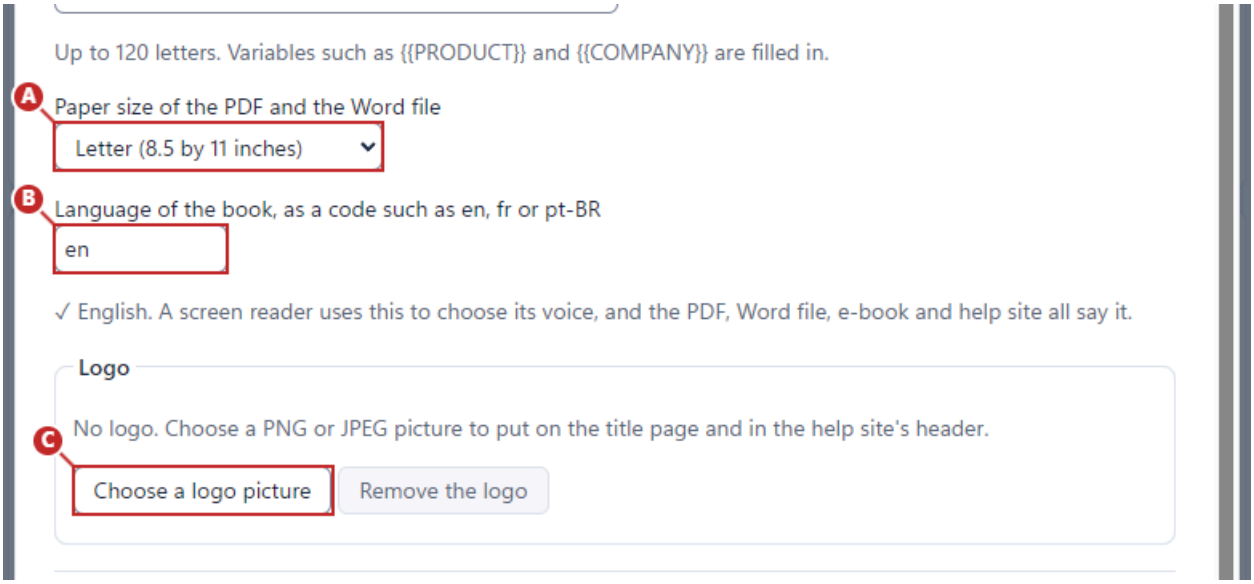
> **✎ DESIGN TIP**
> Choose an accent colour that passes the check. The contrast checker only warns, so a light colour is your choice. A reader with weak eyesight may not be able to read the headings.

Title page. A box says whether the book starts with a title page, with its name, the logo and the PRODUCT, VERSION and COMPANY variables.

Words at the top and bottom of every page. The top is on every page after the first, and empty means the book's title. The bottom sits beside the page number, and empty means nothing. Variables are filled in. Each can be up to 120 letters.

Paper size. Letter or A4, for the PDF and the Word file.

Language. A code such as *en*, *fr* or *pt-BR*. The PDF, the Word file, the e-book and the help site all say the language of the book. A screen reader uses it to choose its voice. The window says the language's name, and warns in words if the code is not one.



A. Paper size: Letter or A4.
B. Language of the book: A code such as en, fr or pt-BR.
C. Choose a logo picture: A PNG or JPEG for the title page.

Logo. A PNG or JPEG for the title page and the help site's header. It needs a description for readers who cannot see it.

The book check lists an accent colour that is too light for text on white, and a logo with no description (topic 6.1, page 68).

7.9 Making an e-book

An **EPUB** is a book for e-readers and reading apps. Unlike a PDF, it reflows to fit any screen and any size of letters, and the reader can change the font and the colours.

To make an e-book:

1. Click **Export** in the navy bar, or choose **Export...** on the **File** menu.
2. Choose **E-book (EPUB)**.

3. Click **Export** in the window.
4. Choose where to save the .epub file.
5. Open the file in the reading app your readers use.

> ϟ **POWER TIP**
> OpenExplainer has not run the e-book through EPUBCheck or tried it in every reading app. Open it in the app your readers use before you publish it.

What is in it

The e-book has a title page if your look has one, the contents, and one page for each chapter with its topics. It has the index and the links between topics. It has the pictures with their descriptions as alternative text, and the callout lists under them. Boxed notes say their kind in words. Comments are never included.

There are **no page numbers**. The pages of an e-book depend on the reader's screen, so a link says *(4.2)* and not a page.

What it says about accessibility

The file records what the book really offers. That is a contents list, headings to move by, an index when there is one, and descriptions of the pictures when every picture has one. It does not claim to meet any standard, because the book has not been checked against one. If a picture has no description, the file says how many have none.

7.10 Translating your book

You can send the words of your book to a translator. The translation comes back as a **copy** of the book in the other language. Your own book is never changed.

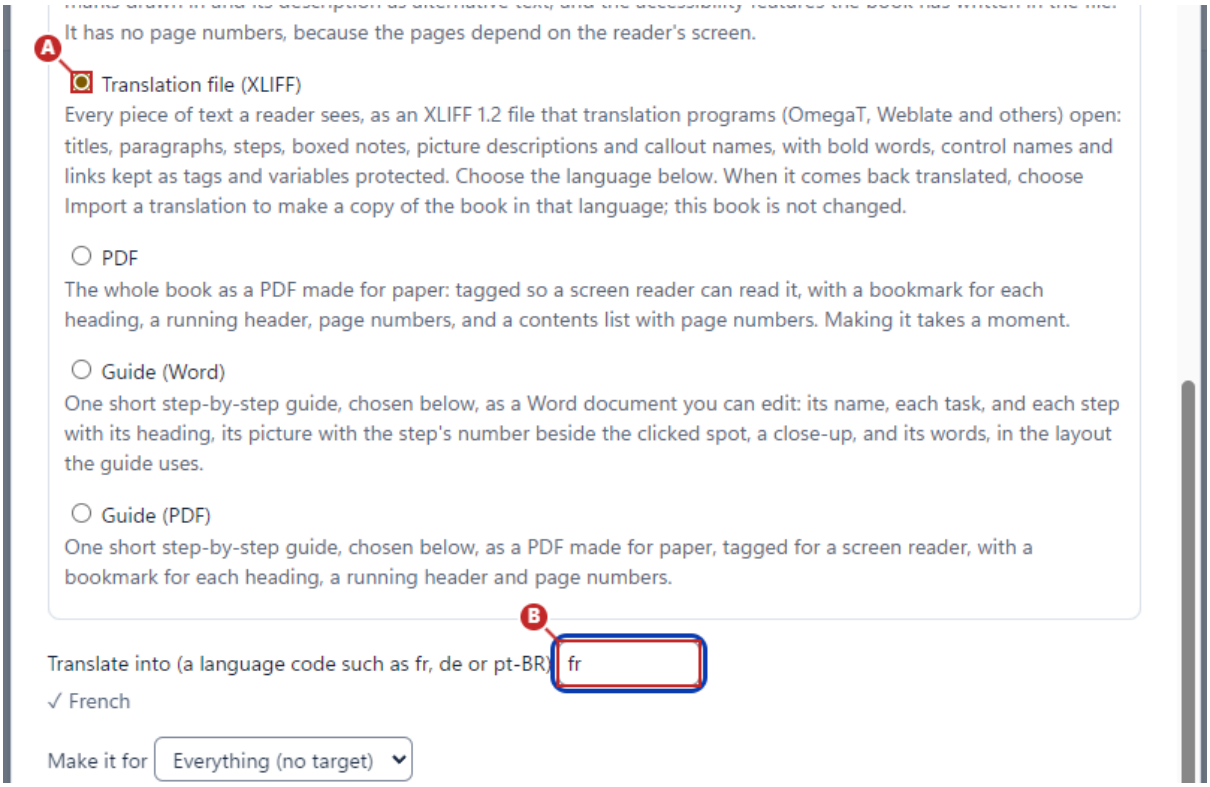
First check that **Look of the book** has the language your book is written in (topic 7.8, page 85).

Sending the words out

To send the words to a translator:

1. Click **Export** in the navy bar, or choose **Export…** on the **File** menu.
2. Choose **Translation file (XLIFF)**.

3. Type a language code such as fr, de or pt-BR in the box called *Translate into*.



A. Translation file (XLIFF): Every piece of text a reader sees, for a translation program.

B. Translate into: A language code such as fr, de or pt-BR.

Result: A line under the box says the name of the language.

4. Click **Export** in the window.
5. Choose where to save the .xlf file.

   Result: The window says how many pieces of text, and about how many words, the translator will see.

XLIFF is the standard file that translation programs such as OmegaT and Weblate read and write. Every piece of text a reader sees is in it: titles, paragraphs, steps and their results, boxed notes, picture descriptions and callout names. Bold words, control names and links are kept as tags that the translator cannot break, and variables such as PRODUCT are protected. The file is XLIFF 1.2.

Bringing the translation back

> ϟ **POWER TIP**
> Only XLIFF 1.2 is read. If your translation program saved version 2, save the file again as 1.2.

To bring the translation back:

1. On the **File** menu, choose **Import a translation…**.
2. Choose the file that came back from the translator.

   Result: A window says how many pieces were translated and what was not.
3. Click **Make the copy**.
4. Choose a name for the new project.

   Result: A new project file is saved. The book you had open is unchanged.
5. On the **File** menu, choose **Open project…**, and choose the copy.

   Result: Your book appears in the new language.

Anything the translator left untranslated stays in the original language and is counted. Links go where the original links went, even if the translator moved the words. The window also tells you about links or variables that were lost in the translation. It also tells you about a file that was made from a different book.

Pictures in the other language

The copy's pictures are the original pictures. To show the translated program, take the pictures again, or use **Replace the picture file…**, which is in the **…** menu under each picture. It puts another PNG or JPEG in the picture's place and keeps its description, its callouts and its marks. The new picture must be exactly the same size, because callouts and marks are placed in pixels. One that is not is refused, and the window gives both sizes.

CHAPTER 8

Using AI help

In This Chapter

- Knowing what AI help will and will not do
- Turning on the built-in model
- Reviewing what the AI drafts
- Telling the AI how your book should read

AI help in OpenExplainer is **optional**. Everything in this guide works by hand with no AI, and the book you make is the same either way. This chapter explains what the AI can do for you if you turn it on, and what it is never allowed to do.

8.1 Knowing what AI help will and will not do

AI help writes **drafts**: first versions of short pieces of text that you then read, change and accept, or throw away. It does not write your book, and nothing it writes goes into the book until you accept it.

> ✎ **DESIGN TIP**
>
> A draft can be wrong. It can describe how to do a step when it should only name the task, or turn a step into a rule. The program checks what it can, for example that a drafted callout rests on something real, but you are the author. Read every draft before you accept it.

What it can draft

A **name and a sentence** for each callout on a picture.

A **chapter opener**, which is two or three sentences at the start of a chapter.

A **chapter summary** for a topic such as Summary.

Suggested boxed notes: up to three sentences that a reader could miss, each with a short boxed note written from it.

A numbered procedure from a click-by-click capture (Chapter 4, page 42) is written by the program itself from what you clicked, with no AI at all.

What it is shown

The model is given only **words**: the clicked item's name, kind and description, the text of your own topics, and your writing-style notes. It is never given a picture, anything you typed in another program, or the contents of a password box. Those are never kept in a project in the first place.

Where it runs

The built-in model runs on **your PC**. It needs no account and, once it has been downloaded, no internet. The program that runs it is told to listen on your PC only, so nothing you write or capture leaves it.

The dialog also shows a third choice, **My own key** (ChatGPT, Claude or Gemini). It is switched off and says that it is not built yet.

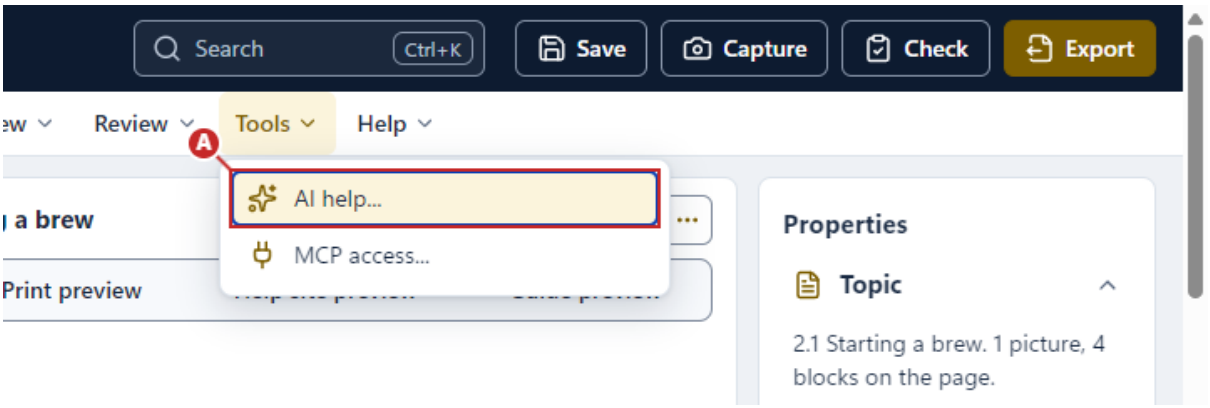
8.2 Turning on the built-in model

The built-in model is **Qwen 3.5** with 4 billion parameters. It is a small language model from Alibaba Cloud, squeezed into a smaller file so that it runs on an ordinary PC. It is about 2.7 GB, so it is not inside the installer. It is downloaded the first time you turn it on, and kept on your PC, outside your projects.

> **TIME SAVER**
> The model is a 2.7 GB download. Check the free disk space that the window reports before you start.

To turn on the built-in model:

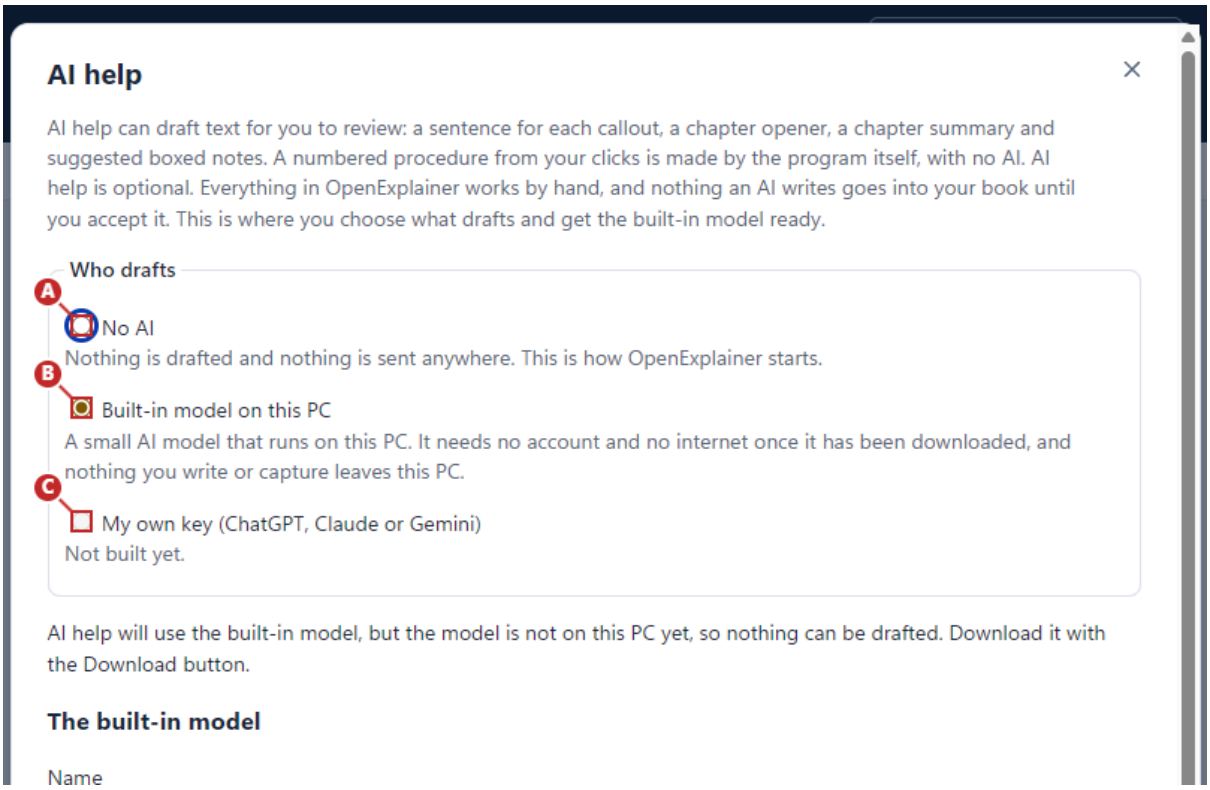
1. On the **Tools** menu, choose **AI help...**.



A. AI help...: Opens the window where you choose who drafts.

Result: A window called AI help opens. Under *Who drafts*, the choice is *No AI*, which is how OpenExplainer starts.

2. Choose **Built-in model on this PC**.



A. No AI: How OpenExplainer starts.

B. Built-in model on this PC: A small model that runs on this PC.

C. My own key: Switched off. It says it is not built yet.

Result: The window shows the model's name, its size, where it comes from, its licence, and whether it is on this PC. It also says how much free disk space and memory the PC has.

3. Click **Download the model**.

   Result: A progress bar shows the numbers in words, such as *1.2 GB of 2.7 GB*.

4. When the download has finished, click **Test the AI**.

   Result: The model writes one short sentence, so that you can see that it works. The first time can take half a minute while the model loads.

5. Click **Save**.

   Result: Your choice is kept. It is not part of any project.

Pause, resume and cancel

Pause keeps what has arrived, and **Resume** carries on from where it stopped. A dropped connection, or closing the program, also continues from where it stopped when you come back. **Cancel and delete the partial download** throws away what has arrived.

When the file is complete, the program checks it against a fixed fingerprint (a SHA-256 value). It then says *Ready*, or that the file is damaged and must be downloaded again. **Check the file** repeats that check. **Delete the model** removes the file and frees the space. **Show the licence text** shows the model's licence, which is Apache 2.0.

Good to know

The program that runs the model keeps going for about 25 seconds after OpenExplainer closes. It holds about 2 GB of memory until it stops by itself. Nothing is harmed.

The model was not downloaded for this edition, so the steps from the download onwards are as the earlier edition and the program's own words describe them (topic 1.5, page 7).

8.3 Reviewing what the AI drafts

A draft is never put straight into your book. It waits in the **Drafts** list, with a note in words that it is a draft and who made it, until you decide.

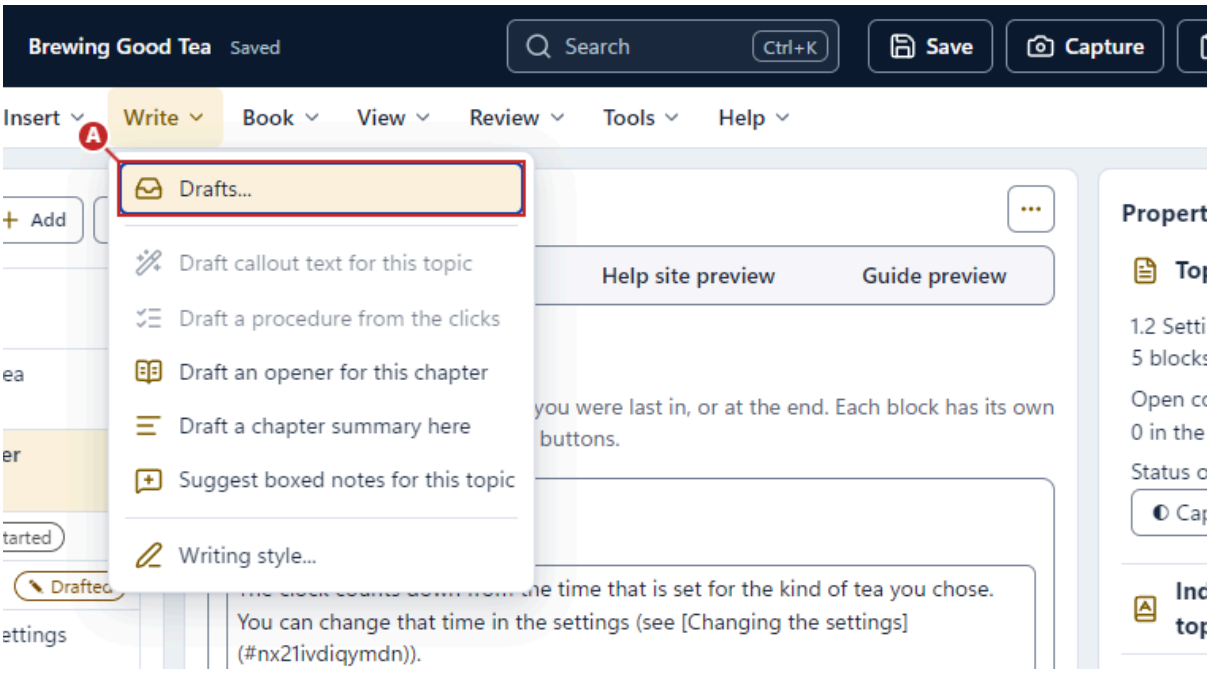
> ✳ **BRIGHT IDEA**
> Drafts are in no preview and in no export. A draft you never accepted cannot reach a reader.

Four commands on the **Write** menu ask for a draft: **Draft callout text for this topic**, **Draft an opener for this chapter**, **Draft a chapter summary here** and **Suggest boxed notes for this topic**. **Draft callout text** in the **…** menu under a picture works on that picture's callouts. The **…** button of the page holds the opener, the summary and the suggested boxed notes too.

To review a draft:

1. Choose one of the four draft commands on the **Write** menu.

2. On the **Write** menu, choose **Drafts...**. The same command is on the **Review** menu.



A. Drafts...: Opens the list of drafts that wait for you. The same command is in the Review menu.

3. Read each draft, and change it if you want to.
4. Click **Accept** or **Reject**.

   Result: Accept puts the words where they belong. Reject throws the draft away.

What you see for each draft

Each draft says where it is and who made it. A callout sentence shows the sentence now and an editable draft. A fold-out shows exactly what the AI was given. A procedure shows each step in its own box with its picture number and what it was made from. A step with no name has a flag. An opener or a summary shows where it will go and its words in a box that you can edit. Suggested boxed notes show the sentence they rest on beside the note, so you can compare them.

Go to the picture (or topic) takes you to what the draft is about. Use Tab to move through the entries and Enter on a button. After Accept or Reject the keyboard moves to the next entry, and Escape closes the list.

What the program checks

The AI is asked for an answer in a fixed shape. An answer that does not fit is thrown away with a plain message. It is never half-used. A drafted step may only name controls that were in your capture, and an unclear label becomes *needs a name* instead of a guess. A drafted opener or summary is thrown away and tried once more if it is a list or has quotation marks. The same happens if it names something that is not in the chapter, or says *you will learn*.

No draft was asked for in this edition, because no model was downloaded (topic 1.5, page 7).

8.4 Telling the AI how your book should read

The **Writing style** window has a box called **Notes** for the AI. Say in your own words how the book should read. The AI reads your notes before it writes a draft.

To write style notes for the AI:

1. On the **Write** menu, choose **Writing style...**.
2. Type your notes in the **Notes** box.
3. Click **Save**.

   Result: The notes are saved with the project.

The box holds up to 2,000 letters, and it says how many you have used.

Good notes are short and concrete. For example: *Speak to the reader as you. Say click for buttons and choose for menu items. Keep sentences short.* You can also say which words to use for which things.

The notes are saved with the project, so a colleague who opens it gets the same style. Leave the box empty if you have no notes. With AI help off, the Notes box does nothing.

The second box in the same window, **Words to avoid**, is a plain list, not an instruction to the AI. It works with AI help off (topic 6.2, page 70).

CHAPTER 9

Letting an assistant do the work

In This Chapter

- Turning on MCP access
- Connecting an assistant
- Knowing what an assistant can do

MCP, the Model Context Protocol, is the way an AI assistant such as Claude uses a program's tools. With MCP turned on, an assistant can document a program **while nobody is at the keyboard**. It makes a project and takes pictures of windows and clicks. It writes the pages, names the callouts, builds the index, runs the book check and makes any export.

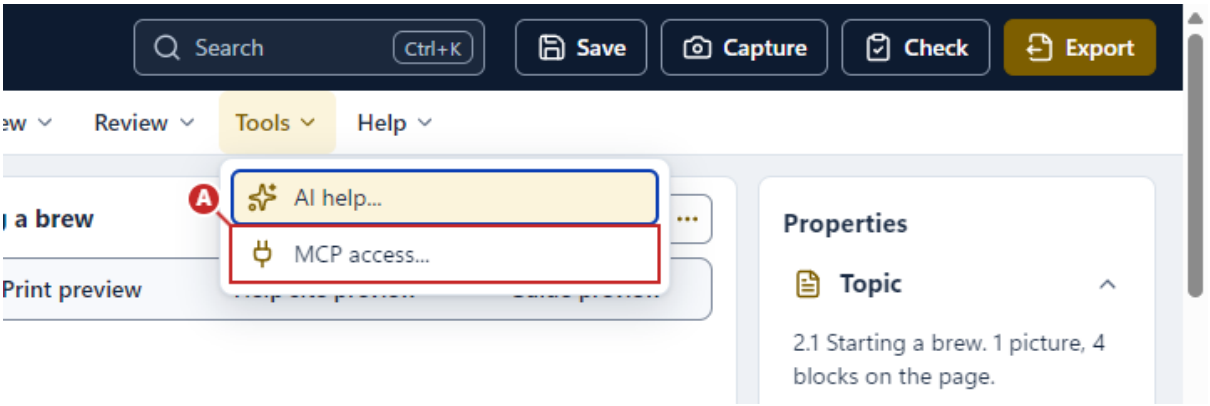
This is optional, and it is **off** until you turn it on. An assistant used these tools, working inside one folder, to write the text of this guide and to take the pictures of its own windows.

9.1 Turning on MCP access

MCP access has three settings: whether it is on, which **folder** an assistant may use, and whether it may capture a **whole screen**.

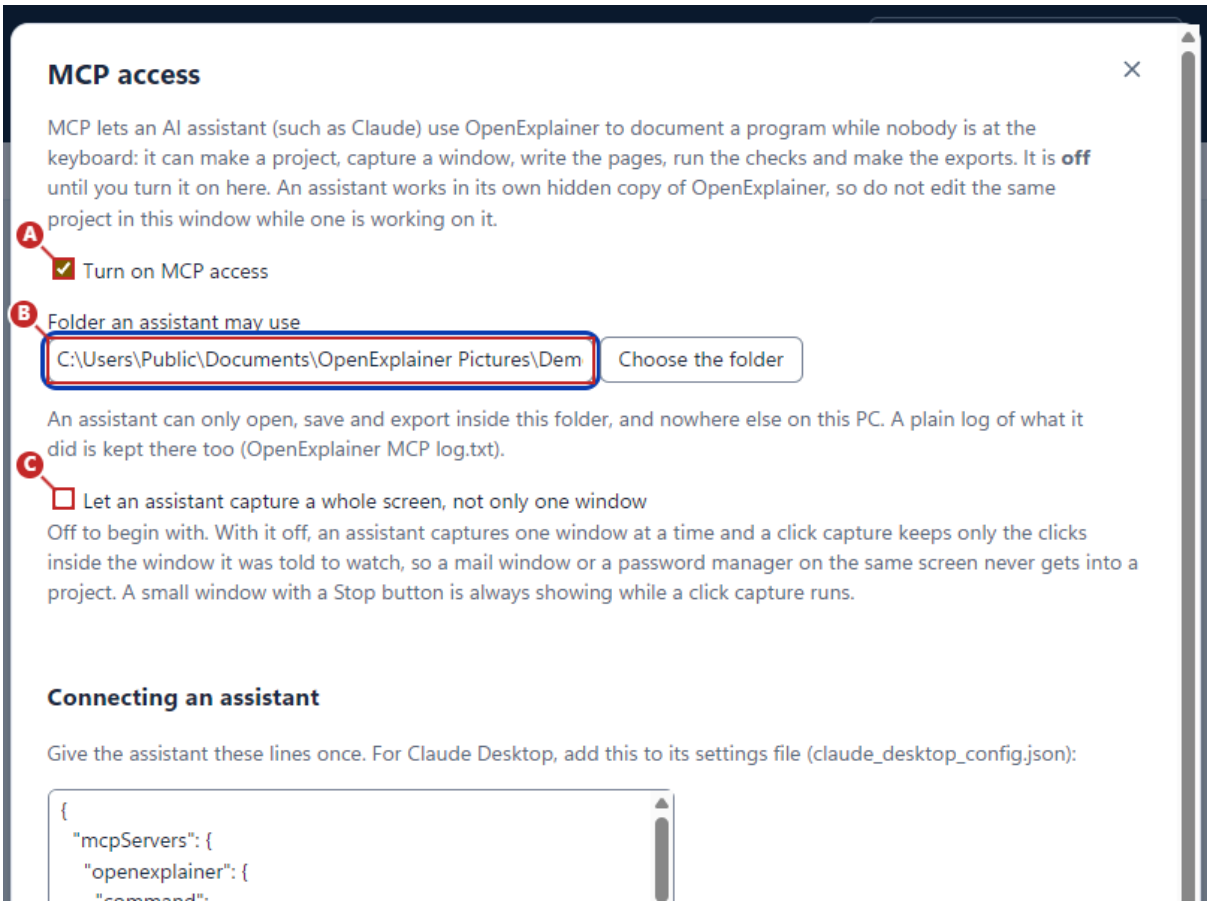
To turn on MCP access:

1. On the **Tools** menu, choose **MCP access...**.



A. MCP access...: Opens the window for letting an AI assistant use the program.

2. Tick **Turn on MCP access**.



A. Turn on MCP access: Off until you tick it.

B. Folder an assistant may use: An assistant can only work inside this folder.

C. Let an assistant capture a whole screen: Off to begin with.

3. Type the **Folder an assistant may use**, or click **Choose the folder**.

   Result: An assistant can only open, save and export inside this folder, and nowhere else on this PC. OpenExplainer will not turn MCP on until you have chosen a folder that exists.

4. Leave **Let an assistant capture a whole screen, not only one window** unticked, unless you need it.

5. Click **Save**.

> **⚠ CAUTION**
> If you turn whole-screen capture on, anything on your screen can end up in a picture. Close private windows first, and look at every picture afterwards.

The settings are kept in a small file called mcp.json in OpenExplainer's settings folder (see topic 10.2, page 104). They are read each time an assistant connects, so a change takes effect for the next assistant that connects.

The folder and the log

Every use of a tool is written to a plain file called **OpenExplainer MCP log.txt** in the folder. It says when, which tool, whether it was OK or REFUSED, and the first line of the answer. What was written on a page is not in the log.

Whole screens

With whole-screen capture off, an assistant takes **one window at a time**. A click capture keeps only the clicks inside the window it was told to watch. A mail window or a password manager on the same screen never gets into a project.

The picture shows a made-up folder. The settings window opened on the settings that were already on the PC, and the folder in the picture was typed in only for the picture, and never saved.

9.2 Connecting an assistant

An assistant has to be told how to start OpenExplainer. The MCP access window gives you the lines to copy for each assistant. You do this once for each assistant.

Claude Desktop

In the MCP access window, click **Copy these settings**, and add them to Claude Desktop's settings file, claude_desktop_config.json. They name the program and the word --mcp. The block looks like this, with the path of your own openexplainer.exe: { "mcpServers": { "openexplainer": { "command": "the path of openexplainer.exe", "args": ["--mcp"] } } }

Claude Code

Click **Copy this command** and run it once in a terminal. It is *claude mcp add openexplainer*, then two dashes, then the path of openexplainer.exe in quotation marks, then --mcp.

What happens when it connects

The assistant starts OpenExplainer with the word **--mcp**. The program then runs with a window that nobody can see and that is not in the taskbar. The assistant talks to it by standard input and output, one line at a time. When the assistant disconnects, the program ends. If MCP is off, the assistant is refused when it connects, with the words that say how to turn it on.

An assistant works in its own hidden copy of OpenExplainer.

> **⚠ CAUTION**
> Do not edit the same project in the window while an assistant is working on it. Each change is saved as it is made, so the last one to save would overwrite the other's work. Keep the project closed in the window, or work in a different project.

9.3 Knowing what an assistant can do

An assistant has tools in five groups. Every tool checks everything first, so one that cannot be done says why in plain words and changes nothing.

The tools

Projects: list, create, open and save projects.

Reading: get the outline, and get a topic with its page, its pictures, what was clicked, its callouts, its index terms and its See also list. Check the book, and list the guides, the snippets, the open comments and the snapshots.

Writing: add, change, move and delete topics. Write a page or add blocks to it. Describe a picture and name its callouts. Make a callout from a click. Set the variables and the index terms, and update the index. Create, change and delete a guide. Choose which part of a picture is shown. Set the book's settings: the zoom, the automatic zoom for print, the click circles, the writing style and the words to avoid.

Capture: list the open windows. A Save or Open window of the program being documented is listed too. Capture one window, or a rectangle (only when whole screens are allowed). Start, check and stop a click capture. Look at a picture, and remove one.

Exports: make a help site, JSON, Markdown, Word, PDF, or a guide as Word or PDF.

A page is written as blocks: a heading, a paragraph, a picture, a numbered procedure or a callout box. Text uses the same marks as the editor: two stars for bold, one star for italic and double square brackets for a control name.

What it does when it captures

The assistant does the clicking in the program with its own tools. OpenExplainer watches and records, exactly as it does for a person. It records a picture from the moment before each click, and a note of what was clicked. It never records anything typed, and never a password box's contents. Each kept picture has its click note and a first callout at the item that was clicked.

To say what was clicked, OpenExplainer asks Windows what is under the pointer. It gets the window under the pointer ready as the pointer first moves over it. So even the first click in a program built on a web browser engine is usually named. A click that lands on the label of a button is named after the button. If another program's see-through window is in the way, OpenExplainer asks the window that was really clicked instead.

The small **Stop** window during a click capture and the **Cancel** window during any other capture are always visible. A person who comes back can stop the capture at once. A copy of OpenExplainer does not capture its own windows.

What it cannot do yet

It does not click or type in the program being documented. The assistant has to do that with its own tools. It does not use the built-in AI to draft text: the assistant writes the text itself. It works on one project at a time for each connection.

Some things can only be done in the program's window. These are moving or resizing a callout's outline, drawing marks on a picture, and choosing one picture's own click circle. An assistant can name a callout and describe it. A click that landed on the window itself gives a small box round the click with no name. You can change its place by hand in the Callouts window (topic 5.2, page 57).

> **✳ BRIGHT IDEA**
>
> Ask an assistant to describe every picture and every callout as it goes. It can see each picture through the tools, and a description written while the picture is fresh is better than one written at the end.

Newer tools

An assistant can also bring in a Word manual, link topics and make See also lists.

It can add and delete callouts on any picture, bring in a picture, and replace a picture with another of the same size.

It can set the look and the language of the book, and write snippets, conditions and targets.

It can take, compare, restore, rename and delete snapshots.

It can make a Word review copy and bring the reviewers' comments back.

It can make an e-book or a translation file, and import a translation as a copy of the book.

Anything that deletes or replaces work makes the assistant say it is sure. Examples are deleting a topic, a snippet or a snapshot, and restoring a snapshot. The answer says what was done. A restore first keeps a snapshot of the book as it was.

CHAPTER 10

Reference

In This Chapter

- Using the keyboard
- Where your files are kept
- When something goes wrong
- Glossary

This chapter gives short answers to the questions that come up while you use OpenExplainer. It covers the keyboard, where things are kept, what to do when something does not work, and what the words mean.

10.1 Using the keyboard

OpenExplainer is meant to be fully usable with the keyboard alone and with a screen reader. Every button can be reached with Tab, and every window closes with the keyboard.

Everywhere

Tab and **Shift+Tab** move between controls. **Enter** or **Space** presses a button. **Ctrl+S** saves the project. **Ctrl+K** searches for a command by its name (topic 2.4, page 14), and **Ctrl+/** lists every key. **F10**, or **Alt** pressed and let go on its own, goes to the menus, and pressed again goes back to where you were. **F6** moves to the next pane (chapters and topics, the page, the properties) and **Shift+F6** to the one before, and the new pane is announced. On a narrow window the panes stack, and F6 still moves between them.

In the menus and the Add and ... buttons

Down, **Enter** or **Space** opens a menu on its first command, and **Up** opens it on its last. **Up**, **Down**, **Home** and **End** move in a menu, and a letter jumps to the next command that starts with it. **Left** and **Right** move between the menus under the navy bar. **Escape** or **Tab** closes a menu and goes back to where it was opened from.

In the list of chapters and topics

Up and **Down** move between items, and **Home** and **End** go to the first and the last. **Right** opens a chapter and **Left** closes it. **Left** on a topic goes to the chapter it is in. **F2** renames the selected item and **Delete** removes it, after asking. **Alt with Up or Down** moves the item, and **Alt with Right or Left** indents or outdents it.

On the page

Each block is a labelled box. A new block goes after the block the keyboard was last in, and the keyboard goes into it. Every block has **Move up**, **Move down**, **Comments** and **Delete** buttons. A button that cannot be used is switched off. The tabs above the page (Edit, Print preview, Help site preview and Guide preview) move with **Left** and **Right**. **Home** and **End** go to the first and the last.

In windows

Escape closes any window and keeps nothing you changed, the same as the Cancel button. Nothing is kept until you press the window's Save, OK or similar button. Every window also has a **Close** button in its corner that does the same. **Tab** and **Shift+Tab** move through a window, and from the last control back to the first. When a window closes, the keyboard goes back to where it was before the window opened.

When you point at a picture (in the Callouts, Picture view and Drawing windows), **Tab** to the outline, the marker or the region on the sample. An **arrow key** moves it by 1 pixel and **Shift with an arrow** by 10. **Control with an arrow** makes an outline wider or taller (Right or Down) or narrower or shorter (Left or Up). **Escape during a drag** puts everything back. The number boxes always hold the exact numbers.

Screen readers

The title of the window names the project and says *(unsaved changes)* in words, and the navy bar says *Saved*, *Unsaved changes* or *Not saved yet* in words too. Status and progress are said in words, for example after a move, a delete or a save. Colour is never the only way to tell things apart. Every status has a word and a symbol, every kind of callout box says its name, and a colour check says its result in words. A picture's description is read as its alternative text.

If you find something you cannot do with the keyboard, or that a screen reader does not say, that is a fault in OpenExplainer. It is not a fault in how you use the program. Tell the author.

The newer windows

History, Look of the book, Reuse and the windows that choose a topic to link to work the same way as the others. Tab goes round inside a window and never leaves it. Escape closes it and puts the keyboard back where it was. The lists are list boxes with labels. What a button did is said in a line under it, so a screen reader reads it. The first stop of the Tab key in the main window is a link, *Skip to main content*.

> **⚠ CAUTION**
> These windows have been checked by program: every control has a name, the keyboard cannot be trapped, and every stop shows where it is. A person has not yet listened to them with a screen reader. The same is true of the navy bar, the menus and the command search: they were checked by program, and used here through the browser engine's debugging port, and no person has used them with a screen reader yet (topic 1.5, page 7). A checklist for doing that is among the project's documents.

10.2 Where your files are kept

OpenExplainer keeps your work and its own settings in different places, so that moving or deleting one does not harm the other.

Your project

A project is **one file** named after the project, with the ending **.openexplainer**. It holds the whole book: the text, the pictures, the callouts and marks, and the settings. Keep it wherever you like, and copy it like any file. To keep a safe copy, use **Save as**.

The program

The installer puts the program in your own folder, **%LOCALAPPDATA%\OpenExplainer**. It needs no administrator rights. It adds a Start menu entry, a desktop shortcut and an entry in Windows' Apps list, where it can be uninstalled. It also tells Windows, for you only, that a file ending in .openexplainer opens in OpenExplainer.

Settings

Settings are in **%APPDATA%\org.openexplainer.app**: the AI choice (ai.json) and the MCP settings (mcp.json). They are not part of any project.

The AI model and other data

The built-in AI model, about 2.7 GB, is kept in **%LOCALAPPDATA%\org.openexplainer.app**, with the browser engine's own profile and a log of the AI program. Uninstalling the program leaves this folder alone, which is right for a big download, and the interactive uninstaller offers to delete it. **Delete the model** in the AI help window removes the model by itself.

Temporary files

While a PDF is made, a work folder called **openexplainer-pdf** is created in the PC's temporary folder. It is cleared away after each export, and any leftover is cleared the next time.

To type one of those paths, paste it into the address bar of File Explorer. For example, %APPDATA% opens your settings area.

A project also holds your **snapshots** (topic 2.7, page 21). The format of the file has a version number, now 21. A project from an older version is upgraded when you open it. One from a newer version is refused in words, so it is never opened and saved without parts of it. A translation (topic 7.10, page 88) is a new project file, and the book you translated is unchanged.

10.3 When something goes wrong

I cannot find a command

Press Ctrl+K, type a few letters of its name, and press Enter (topic 2.4, page 14). A command that is dimmed in its menu cannot be used yet. **Draft a procedure from the clicks**, for example, needs a topic with a picture taken click by click (topic 2.3, page 12).

Windows warns about the installer

Windows SmartScreen may warn when you run the installer. The first versions were not signed. Alpha 3 is signed by Jason Wrench, by its release notes, but SmartScreen may still warn on a first download, because the signer is new to it. Only run an installer that you got from the project's own page.

The window I want is not in the list

In the Capture window, click **Refresh the list**. A window that is minimized has to be restored first, and the list says so.

A capture is taking too long

A capture is normally over in a fraction of a second. If it is not, press the **Cancel** button in the small Cancel window. The keyboard may be in the other program, which is why that window has its own button.

A click has no name

OpenExplainer keeps the name of a clicked item only for buttons, menus and similar controls, and only when Windows can say what it is. It gets a window ready the first time the pointer moves over it. So the first click in a program built on a web browser engine is usually named too. A click that lands on the writing or the picture inside a button counts as a click of the button.

When another program's see-through window is over the window, such as a writing helper's overlay, OpenExplainer asks the window you clicked what is there. The click's note says so.

If a click still has no name, name the callout yourself in the Callouts window. This happens when the item is a list item, a link or a text box, whose names are never kept. It also happens when the pointer was moved and clicked in one go.

A callout is in the wrong place or too big

When the click landed on the window itself and not on a control, the first callout is a small box round the click, with no name. Open **Callouts** and give it a name. Then change the four numbers of the outline, or drag it on the sample, to put it round the real control. A project made with an earlier version of OpenExplainer may still have a callout that covers the whole window. The same steps fix it.

A picture prints too small

Choose **Picture view…** in the **…** menu under the picture and crop it, or zoom in on the click, or turn on the automatic zoom in **Zoom settings…** on the **Book** menu. The print preview lists every picture that will print too small.

The contents of the PDF has no page numbers

The page numbers are read out of the finished file. If that is not possible, the contents is left without numbers, never with wrong ones, and the export says so. Make the PDF again.

An assistant is refused

MCP access is off until you turn it on. On the **Tools** menu, choose **MCP access…**, tick **Turn on MCP access**, choose the folder, and click Save. Remember that an assistant can only work inside the folder you chose.

Draft callout text says it cannot

AI help needs the built-in model. On the **Tools** menu, choose **AI help…**, choose **Built-in model on this PC**, and check that the model is downloaded and says *Ready*. Everything else works without it.

A project does not open when I double-click it

The installer makes the ending .openexplainer open in OpenExplainer, but Windows may keep another program for it if one was chosen before. Right-click the project's file, choose **Open with**, choose OpenExplainer, and choose to always use it. If OpenExplainer was installed by another person on the PC, it is not set up for you: use **Open project…** on the **File** menu.

A project was made by a newer version

A project saved by a newer version of OpenExplainer is refused with a message that asks you to update the program. Older projects are upgraded by themselves when you open them.

A snapshot cannot be taken

The book keeps up to 20. On the **File** menu, choose **History…**, delete one you no longer need, and try again.

A translated file is refused

OpenExplainer reads XLIFF 1.2 only. If the window says the file is XLIFF 2, save it again as 1.2 in the translation program. If it says the file does not say which language it is in, set the target language in the translation program and save again.

A picture is refused when I replace its file

The new picture must be exactly the same size as the old one, because callouts and marks are placed in pixels. The window gives both sizes. Make the new picture the same size, or import it as a new picture and place its callouts again.

Not all the reviewers' comments came in

The window counts the comments it could not place. The file may be a copy of another book, or the text a comment was made on may have been taken out of the book. Comments that you wrote yourself in the review copy are left as they are, and a comment already in the book is not added twice.

10.4 Glossary

Accessibility check. An optional look at the book for things that make it harder to use, such as a picture with no description. See the book check.

Alternative text. The words that stand for a picture for a reader who cannot see it. In OpenExplainer it is the picture's description, with the words of its marks.

Book check. The optional list of things worth fixing in a book (topic 6.1, page 68).

Callout. A lettered marker on a picture, with an outline round a part and a line to the marker. Letters belong to pictures (topic 5.2, page 57).

Callout box. A boxed note on a page: a Design Tip, Caution, Bright Idea, Power Tip, Time Saver or New in this version (topic 3.4, page 30).

Chapter. A top-level part of the book. It can hold topics.

Click circle. A transparent circle drawn on the spot where a click was made. It is drawn when the book is made, not into the picture (topic 4.5, page 51).

Command search. Ctrl+K, a window that finds any command by its name, its menu or what it is for (topic 2.4, page 14).

Condition. A name for content that is only for some readers or versions. A target can hide it (topic 3.9, page 37).

Control name. The label of a button, menu or box, written between double square brackets. It is shown in bold and used by the index and the book check (topic 3.5, page 31).

Description. What a picture shows, in words (topic 5.1, page 55).

Draft. A piece of text that the AI wrote and that is waiting for you to accept or reject it (topic 8.3, page 94).

EPUB. A book for e-readers that reflows to fit any screen (topic 7.9, page 87).

Guide. A few pages for one task, made from topics already in the book (topic 7.6, page 81).

Help site. The book as a folder of web pages with search (topic 7.4, page 80).

Index. A list of terms with the topics they are in (topic 7.7, page 84).

Language. The language the book is written in, as a code such as en or fr. A screen reader uses it to choose its voice (topic 7.8, page 85).

Link. Words in a paragraph that go to another topic. It follows the topic when it moves (topic 3.8, page 35).

Look of the book. The fonts, accent colour, title page, header, footer, paper size, language and logo of the outputs (topic 7.8, page 85).

MCP. The Model Context Protocol, the way an AI assistant uses a program's tools (Chapter 9, page 97).

Marks. Arrows, text boxes, rectangles and numbered markers drawn over a picture (topic 5.4, page 65).

Menus. The nine menus under the navy bar hold every command (topic 2.3, page 12).

Navy bar. The bar across the top of the window, with the project's name, the Search box and the buttons Save, Capture, Check and Export (topic 2.2, page 10).

Picture view. Which part of a picture is shown: the whole, a region, or a zoom on the click (topic 5.3, page 61).

Procedure. A numbered list of steps. The numbers follow the order, and each procedure starts at 1 (topic 3.3, page 29).

Project. One file with the ending .openexplainer that holds the whole book.

Review copy. A Word file of the book, with your open comments in it, for reviewers to add their comments to (topic 6.3, page 73).

See also. A list of related topics at the end of a topic (topic 3.8, page 35).

Snapshot. A dated copy of the whole book, kept inside the project file, that you can go back to (topic 2.7, page 21).

Snippet. A piece of a page that you write once and use in many topics (topic 3.9, page 37).

Status. Where a topic is in its life: Not started, Captured, Drafted, Reviewed or Needs recheck (topic 3.7, page 34).

Target. A named output that leaves out content with certain conditions (topic 3.9, page 37).

Topic. A part of a chapter, with a page of its own. A topic can hold more topics.

Variable. A name in double curly brackets that stands for a value kept once for the whole book, such as PRODUCT (topic 3.6, page 32).

XLIFF. The standard file that translation programs read and write (topic 7.10, page 88).

Index

Symbols and numbers

--mcp 99
.openexplainer 19

A

A region I choose 61
About this version and licence 7
accent colour 85
accept and reject 94
accessibility 55
accessibility check 68
Add 10, 12, 16, 25, 27, 29, 30, 37, 53
Add a callout 57
Add a condition 37
Add a target 37
Add an arrow 65
Add chapter 12, 25
Add step 29
Add topic under selected 12, 25
AI help 12, 91, 105
AI model 92
ai.json 104
alternative text 55
Apache 2.0 licence 4
arrows 65
assistant tools 100
automatic zoom 61

B

blocks 27
bold 31
Book 10, 32, 37, 51, 61, 81, 84, 85, 105
book check 68
bookmarks 78
boxed notes 6, 30
Bring in review comments 12, 73
Building the index 84
Built-in model on this PC 92, 105

C

Callout A 57
Callout box 12, 30
callouts 57, 105
Cancel window 105
Capture 10, 12, 47, 53, 100
capture a rectangle 44
capture a window 42
Capture this rectangle 44
Capture this rectangle (Enter) 44
Caution box 30
chapters 10, 12, 25
Chapters and topics 10
Check 10, 68
Check the book 12, 68
Choose a logo picture 85
Choose conditions for this topic 37
Claude Code 99
Claude Desktop 99
click capture 47
click circle 51
Click circle settings 12, 51
Close 14
close-up 81
Colour 51
comments 27, 34, 102
COMPANY variable 32
conditions 37
Conditions of this topic 10, 37
consistency 70
contrast 65
control name 31
conventions 6
copyright 7
Create 16
crop 61
Ctrl+S 19

D

Delete 12, 25
description of a picture 55
Description of the picture for click 1 49
Design Tip box 30
disk space 5
docx 79
Draft a chapter summary here 12, 94
Draft a procedure from the clicks 12, 29, 49, 105
Draft an opener for this chapter 12, 94
Draft callout text 94
Draft callout text for this topic 12, 94
drafts 12, 91, 94
Drawing 65

E

e-book 87
Edit 10, 12
empty project 16
EPUB 87
EPUBCheck 87
Export 10, 12, 14, 78
Exporting Markdown and JSON 81

F

F6 10, 102
File 6, 10, 16, 19, 21, 40, 73, 78, 79, 80, 81, 87, 88, 105
Find and replace in titles 12, 25
Finding a command in the menus 12
Finding what is still missing 68
Finding your way around the window 10
Folder an assistant may use 97
Font of the text 85
fonts 85

G

glossary 107
Go to it 68
Go to the block 34
Guide preview 10, 12, 75, 81
guides 12, 81

H

header and footer 85
Heading 12, 27
Help 10, 14
help site 80
Help site preview 10, 12, 75
History 12, 21, 105
How the pages are written 6

I

Import a translation 12, 88
import an export 12, 81
Import from Word 7, 12, 40
Import pictures 12, 53
importing pictures 53
Indent 12, 25
index 12, 80, 84
index terms 84
Index terms for this topic 10, 84
Index terms, one on each line 84
Insert 10, 27, 42, 44, 47, 53
installer 9
italic 31
Item: how far across it starts 57

J

JPEG 53
JSON 81
JSON with images 81

K

Keep click 2 49
Keyboard keys 12, 14
keyboard shortcuts 102
Kind of box 30
Knowing what AI help will and will not do 91
Knowing what an assistant can do 100

L

language 85
language code 88
Language of the book 85
Leaving notes and tracking progress 34
Let an assistant capture a whole screen 97
letters on pictures 57
licence 7
Link to a topic 35
links 35
llms.txt 80
logo 85
Look of the book 7, 12, 79, 85, 88

M

Make a guide from a chapter 81
Make it for 37
Making a help site 80
Making a PDF for print 78
Making a Word file 79
Making an e-book 87
Markdown 81
Marking bold, italic and control names 31
marks 65
MCP 97
MCP access 12, 97, 105
MCP folder 97
MCP log 97
mcp.json 104
menus, capturing 44
model download 92
model folder 104
More info 9
Move down 12, 27, 57, 102
Move up 12, 25, 27, 57, 102
My own key 91, 92

N

Name 57
Name of the guide 81
named styles 65
Narrow the list 35
Needs a description 55
New project 12, 16
New variable name 32
No AI 92
Notes 70
Notes box 96
notes for the writer 75
numbered procedure 12, 29
numbering 25

O

Open project 12, 19, 105
open source 4
Open windows 42
Outdent 12

P

Page 10
panes 10
paper size 85
Paragraph 12, 27
password box 47
PDF 75, 78
Picture 12
Picture for step 1 29
Picture view 61, 105
Place the marker for me 57
PNG 53
Pointing at part of a picture 57
Previewing before you export 75
print preview 10, 12, 75
privacy 47
procedures 29
PRODUCT 32
PRODUCT variable 32
Project variables 12, 32
Properties 10
Properties pane 10

Q

quick start guide 81
Qwen 92

R

Refresh the list 105
Rename 12, 25
Replace the picture file 88
Restore the chosen topic 21
Restore the whole book 21
restoring 21
result line 29
reuse 12, 37
Reusing text and building variants 37
Reusing the product name everywhere 32
Review 10, 34, 68, 94
review copy 73
Review your clicks 49
reviewers 73
Reviewing what the AI drafts 94
Run anyway 9

S

Save 10, 12, 19, 81
Save as 6, 12, 19, 104
saving 19
Screen 44
screen reader 55, 102
screenshot 42
search 80
Search box 14
Search for a command 12
Searching for a command, and the list of keys 14
See also 35
See also for this topic 10, 35
Select on the screen 44
Selected block 10
Setting the look of your book 85
settings folder 104
setup file 9
Show only terms that contain 84
Show the click circle in the book 51
Showing where the reader clicks 51
Size 51
SmartScreen 9
SmartScreen warning 105
snapshots 21
Snippet 12
snippets 37
Start capturing clicks 47
Start from 16
status of a topic 34
Status of this item 34
steps 29
still pictures 4
Stop and keep the clicks 47
Suggest boxed notes for this topic 12, 94
Suggest terms 84
system requirements 5

T

tagged PDF 78
targets 37
Telling the AI how your book should read 96
temporary files 104
Text box 31
Text of the box 30
The … button 25, 57, 61
The book starts with a title page 85
The four number boxes 44
The navy bar and the menus 10
The rectangle 44
The whole picture 61
title page 85
toolbar 10
Tools 10, 92, 97, 105
Topic 10, 34
topics 25
Translate into 88
translation 88
Translation file (XLIFF) 88
troubleshooting 105
Turn on MCP access 97
tutorials 4

U

uninstalling 9
Unresolved comments 12, 34
unsaved changes 19
Update Field 79
Update the index from the book 84
updating 9

V

variables 32
version 7
version history 21
VERSION variable 32
View 10, 75
view of a picture 61

W

Wait before capturing 42
web pages 80
What it says 37
What the file holds 73
What to do 29
whole-screen capture 97
window 10
Windows 11 5
Word 35
Word file 79
Word import 40
Word review copy 7, 73
Words to avoid 70
Write 10, 29, 49, 70, 94, 96
Writing boxed notes 30
writing style 12, 70, 96

X

XLIFF 88

Z

zoom 61
Zoom in on the click automatically 61
Zoom settings 12, 51, 61, 105
Zoom size in the book 61
Zoomed in on the click 61