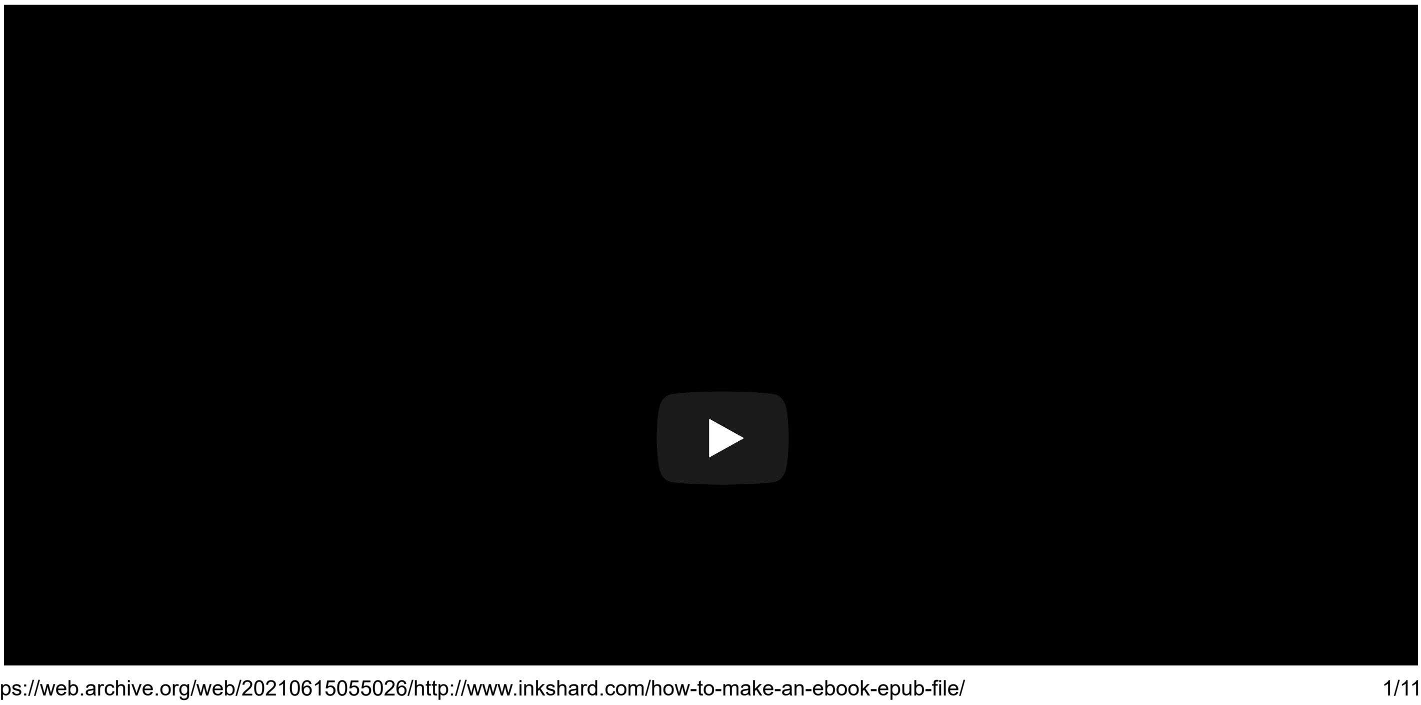
The Wayback Machine - https://web.archive.org/web/20210615055026/http://www.inkshard.com/how-to-make-an-ebook-epub-f…

INKSHARD

a blog from author Eric Muss-Barnes

How To Make An eBook EPUB File

by Eric Muss-Barnes on 3 September 2013



[redacted]

Many authors get really confused when creating an eBook as to what kind of format to use. eBooks can be made as a ton of different formats:

.aeh (used by *Archos* eReaders)

.lrx (used by *Sony* eReaders)

.ibooks (used by *Apple* eReaders)

.pkg (used by *Newton* eReaders)

.mobi (used by *Amazon Kindle* eReaders)

.epub (used by just about everyone else, including *Barnes & Noble NOOK* eReaders)

There are actually even more formats than those. That's just a small sample. So, which one should you make?

Well, the only formats you need to create are EPUB and MOBI. Forget the others. EPUB is quickly becoming the industry standard and 90% of the eReaders on the market can open EPUB files. There is also a very simple conversion tool to change your EPUB into a MOBI. So you really only need to make an EPUB, convert it to MOBI, and your book will be accessible on 99% of the eReaders out there, including NOOK and Kindle.

Today on InkShard, I want to teach you to do something awesome! Something I have never seen anywhere else on the Internet. I'll show you how to create an EPUB file that can be published!

Now, the caveat "that can be published" is what makes my tutorial unique. There are plenty of tutorials on EPUB files, but they are all wrong for publishing! See, here's the issue – there is a big difference between an EPUB that conforms to VALIDATION standards and one that can be PUBLISHED.

For example, when you want to publish a book with certain companies, they may have parameter requirements that must be present in your EPUB file, but those parameters are *not* required in order to validate the file. So, you go to all this trouble to follow another tutorial, you test your file, everything is valid, then the publisher rejects it! You have no idea why it wa rejected, because you did everything right, the file works fine when you test it, it passes all the validation software. What's the problem? Well, the problem is, you made an EPUB that can be VALIDATED not one that can be PUBLISHED.

So, I'm going to teach you how to create an EPUB file that is not only valid and properly formatted to all the standards, but it will *also* be accepted for publication at:

Amazon

Lulu

Smashwords

Barnes & Noble

iBookstore

At the time this tutorial was created, I successfully published 5 of my own books, to all of thos distributors, using these techniques. So, I know it works.

Before we begin, I want to warn you – this tutorial presumes you have some basic computer knowledge and basic HTML knowledge. If you don't know anything about filepaths, or directory structures, or save dialogs, or how to turn on your file extensions, or basic HTML, you will be lost in this tutorial. It will be too technical and confusing. But, if you know basic HTML, you'll be fine.

Also, I will be giving you these instructions on a PC. If you use a Linux machine or a Macintosh, you might need to do things a little differently from the instructions, but all the basics are going to be similar.

Finally, I am not your free tech support. I love to help people out, but if you run into problems, I can't be answering 50 emails a day asking me to troubleshoot your mistakes. Figur it out on your own. I'm like Morpheus in *The Matrix* – "I can only show you the door, you're the one who has to walk through it."

Okay, Neo, let's get started.

First…

SOFTWARE YOU'LL NEED TO CREATE EPUB:

- **Good Text Editor with Syntax Highlighting** (such as Notetab++ or jEdit)
- **.ZIP Compression Program** (such as WinZip)
- **Web Browser** (such as Chromium or Firefox)

DOWNLOAD THE SAMPLE FILE I HAVE CREATED FOR YOU FROM:

- **MAIN:** http://www.InkShard.com/EPUB_Template.zip
- **MIRROR:** http://www.EricMuss-Barnes.com/EPUB_Template.zip

EPUB files are very easy to create. In the simplest terms, an EPUB is comprised of carefully structured XHTML files and folders, saved in a very particular way as a .ZIP archive, ther you simply rename the .ZIP extension to .EPUB.

There are a few general rules you need to follow.

1. Save all XHTML as UTF-8 and add Unicode Signature (BOM)
2. Follow XHTML rules (such as making sure you close all tags including line break BR tags and paragraph P tags and LI list tags).
3. Remember to make all of your HTML tags lowercase.
4. Make sure your book title always follows the proper grammatical rules of case structure. Your book will be rejected by iBookstore if the case of your lettering is grammatically incorrect.

Now let’s take a look at the EPUB_Template.zip file. When you open the .ZIP, you will find two folders and one file:

FOLDER: **META-INF**

FOLDER: **OEBPS**

FILE: **mimetype**

The **mimetype** file and the **META-INF** folder are essential to creating your EPUB, but you won’t have to edit those or change anything in them. Just leave them alone. Open the **OEPBS** folder. That is where we will do all of our work. Inside that folder, you will see:

FOLDER: **Images**

FOLDER: **Styles**

FOLDER: **Text**

FILE: **content.opf**

FILE: **toc.ncx**

Let's start with **Images**. That is where you will put your bookcovers and your author photo (assuming you use an author photo) and any other images that might appear inside your book Obviously, you would replace the images in this sample with your own photos. Keep the height and width dimensions the same as the ones in my example. Why? Because some eReaders have troubles with sizes that are too big or too small. I found the sizes I have provided are of a size that will conform to all the eReaders on the market. So, don't change the dimensions of your covers and author image. I have never seen a single eBook use a backcover, except my own. I don't understand why eBook publishers don't give their customers back covers. I plan on starting a trend in the eBook industry of providing a back cover, and thanks to my template, you'll be able to have a back cover too.

Styles contains the stylesheet. I suggest you just leave this file alone and don't edit it unless you're an expert at HTML and you know your way around CSS. I programmed websites for Walt Disney Studios for 6 years, so I know how to create a CSS file. If you know CSS and want to tweak things or tinker, go right ahead, otherwise, don't touch the CSS and just leave it be.

Now we get into the gritty stuff. Let's start with the **Text** folder. The **Text** folder contains all of the chapters of your book, your copyright page, title page, table of contents and any other extra pages you might want like a biography or colophon. First, I will explain the quirks of putting all of your chapter information in the book, then I will explain the quirks of the required pages.

ALL THE CHAPTER PAGES INSIDE TEXT FOLDER:

- Be sure to read my notes inside the sample file. There are lots of tips and hints in there.

- Make sure your chapter TITLE tag in the HEAD matches your H1 tag.
- Do not change the H1 tag for your chapter headings! Chapter headings must be inside H1 tags in order to conform to requirements of certain publishers, such as iBookstore.
- Be sure to use standard HTML tags for **bold** B and *italics* I. Some tutorials suggest using CSS, but that would require a SPAN tag and, as of this writing, Kindles don't recognize CSS variables in SPAN tags. Therefore, stick with HTML tags instead.

REQUIRED PAGES INSIDE TEXT FOLDER:

title.xhtml

cover.xhtml

backcover.xhtml

toc.xhtml

copyright.xhtml

The **title.xhtml** page is pretty self-explanatory. Nothing much to it, beyond your name and the title of the book.

You don't need to edit the **cover.xhtml** and **backcover.xhtml** pages. Just leave those files alone and make sure you edit your cover pictures in your **Images** folder.

The **toc.xhtml** file is used by the eReader to display the Table of Contents to the person reading your book. Make sure all the pages are in the list. Note, there are no "required" files in the Table of Contents per se, but most retail distributors will obviously require that your Tabl of Contents is accurate. Note the front cover is not something you are required to list.

The **copyright.xhtml** file must follow the formatting included. Do not place any information on the page above the H1 tag and do not change the H1 tag formatting. Again, this is not an EPUB requirement, but it is a requirement for some distributors who have very strict rules about how the **copyright.xhtml** page must be laid out. Below the H1 tag, you are free t make all the changes you want. For example, you may want to include copyright information for a photographer who provided an author photo or a URL to your website. Such extra infor mation is all permissible.

That covers all the files in the **Text** folder.

Finally, we move to the two most complicated files, the **content.opf** and **toc.ncx** inside the root of **OEBPS**. The formatting of these files is similar to a typical XML file in that they have various self-defined nodes containing different bits of data.

Let's start with **content.opf**.

On line 5 of **content.opf**, you will see the node **dc:title** and that contains your book title

On line 9 of **content.opf**, you will see the node **dc:identifier** and that contains any unique serial-number variable you want to use. Most people use an ISBN.

On line 10 of **content.opf**, you will see the node **dc:creator** and that contains your name.

On line 11 of **content.opf**, you will see the node **dc:publisher** and that contains your publisher name. If you are self-published, simply put your own name in there. This node is required by Barnes & Noble.

On line 12 of **content.opf**, you will see the node **dc:date** and that contains the day your book is being released. Without this, it will be rejected by Barnes & Noble.

The next section of **content.opf** is the **manifest**. The **manifest** is basically a file list, cataloging every single file that appears in your EPUB, including every single image, stylesheet and XHTML file – all the contents of your **Images** and **Styles** and **Text** folders. The **manifes** does not need to be in any particular order, but most people like to keep it organized in an alphabetical order, just to make it easier to read. Note that the **manifest** must be 100% accurate There can not be any files missing and there can be no additional files listed that aren't actuall used. If you skip any files or add extraneous ones, your EPUB will fail validation. So, make sur your **manifest** is clean. Note the **id** field in all of your **manifest** nodes must be a unique variable. So, for example, if you add more images, you need to give each one a unique name.

The next section of **content.opf** is the **spine**. The **spine** controls the order your pages show up in the book. So, while the **manifest** can be in any order, the **spine** must be in the proper order. Do not change the ordering of the first 4 pages in the **spine** or your book may b rejected. The pages should remain **bookcover** then **title** then **copyright** then **toc**. Note the **idref** variable in the **spine** nodes match the **id** field in all of your **manifest** nodes. Again, as with the **manifest**, make sure the **spine** is 100% accurate to all the pages in your books and contains no extra files and is missing no files. If pages fail to appear in your ebook and you know they exist, they are most likely missing from the **spine**.

That's all you need to edit in **content.opf**. There are other nodes like **guide** and languag but you can just leave those alone. There's no need to change anything there. I presume since you're reading this in English, your book will be in English, so there's no need to alter the language field.

Finally, we get to **toc.ncx**, the last file to edit in the **OEBPS** folder.

Like the **content.opf**, the changes made to **toc.ncx** require a high degree of accuracy. As you can see, the file is similar to your **toc.xhtml** but with a bit of extra information.

On line 3 of **toc.ncx**, you will see the meta tag **dtb:uid** and that must match the **dc:identifier** on line 9 of the **content.opf** file.

On line 9 of **toc.ncx**, there is a **title** tag – put your book title there.

Nested in side the **navMap** node of **toc.ncx** are **navPoint** nodes. As you can see, the **navPoint** nodes are similar to the chapter listings inside the **toc.xhtml**. The **id** tag in the **navPoint** node must match the corresponding **id** tag and **idref** tag in the **content.opf** file. The **content src** node in **toc.ncx** must match the filepath in the **a href** tag of **toc.xhtml**. Th **text** node in **toc.ncx** is not tied to match anything. The **playOrder** tag must be sequential and no numbers can be skipped or missing.

That covers all the files you need to edit in the **OEBPS** folder.

You're done editing!

Once you have all of your chapters created, and you have done all of the required editing to your **content.opf** and **toc.ncx** files, it's time to create the EPUB file.

STEPS TO CREATE EPUB:

1. Create an empty new .ZIP file
2. Change the name of the file to the name of your book
3. Open the empty .ZIP file

4. Drag **mimetype** into the blank .ZIP (You must place the **mimetype** inside the ZIP file first, otherwise the file-structure will be invalid.)
5. Drag **META-INF** and **OEBPS** into the .ZIP
6. Close the .ZIP
7. Rename the ".ZIP" extension as ".EPUB"

That's all there is to it! You have just finished creating an EPUB file!

But, we're still not done. Just because all your files open successfully in a web browser doesn't mean your EPUB structure is conforming to all of the EPUB standards. So, once you have created your EPUB, the next step is to run some validation software on that EPUB to make sure it has no unseen errors inside of it. Do not skip the validation process. Validating is an essential step and if you skip it, your books could easily be rejected from publication due to mistakes you didn't even know were there.

SOFTWARE YOU'LL NEED TO VALIDATE EPUB:

- **Standalone Validator** (such as Flight Crew)

WEBSITES WHERE YOU SHOULD DOUBLE AND TRIPLE CHECK YOUR VALIDATION (using ePUB Check):

- http://www.epubconversion.com/ePub-validator-iBook.jsp
- http://validator.idpf.org

Why use 3 different resources to validate your EPUB? Because different validators frequently find different errors. For example, Flight Crew might find an error that the online ePUB Check software will miss. Therefore, it is crucial that you take the time to validate your EPUB in as many validators as possible. Remember, one minor little mistake can cause your file to be rejected, so it's important that you not miss anything.

Typically, the errors are very explicit and even tell you right down to the line number where you have made a mistake.

Once you have validated that your EPUB conforms to all the validation standards, try opening it up on your computer and reading it. You can do this by using free EPUB reader software. Again, I'd suggest checking on multiple readers, not just one. Not all readers will ren der everything the same, so it's important to check for consistency across multiple products.

SOFTWARE TO READ EPUB:

- **NOOK Reader App** (different versions are available for different operating systems)
- **Sigil**
- **calibre** (the British spelling of "caliber")

Once you have successfully opened your EPUB in all of your readers and have it working properly, you are all done! Now you have a completed, validating a publishable EPUB. Congratulations!

Now that you have a fully-validating EPUB file, you need to convert it to a MOBI file if you wish to sell your book on Amazon.

SOFTWARE YOU'LL NEED TO CREATE & VALIDATE MOBI:

- **KindleGen**
- **Kindle Previewer**

Rumors have been circulating for a long time that the Amazon Kindle might someday gain the ability to read EPUB files and if that happens, this step will obviously become unnecessary. But, back when I made this tutorial in 2013, you still needed to do a MOBI conversion.

Here are the steps to make a MOBI file…

1. Download the latest **KindleGen** from Amazon.com.
2. Install **KindleGen** directly onto your **C:** drive in a folder called **KindleGen**. The reason you install it to the root of your **C:** drive is because you need to type a filepath in the Command Line Utility and you don't want to have a tons of stuff to type. By keeping it installed directly in the root of your **C:** directory, you have less to type.

3. After you install **KindleGen**, you should have a folder in **C:\KindleGen** that contains **kindlegen.exe** and a **Readme.txt** file.
4. Copy your final **name-of-your-book.epub** into the folder **C:\KindleGen**.
5. Go to **Start** -> **Run…** and type **cmd** and hit return.
6. That should launch your Command Line Utility.
7. In your Command Line Utility, type **cd c:\KindleGen** and hit return.
8. Type **KindleGen name-of-your-book.epub** and hit return.
9. MOBI file will build and be saved in **C:\KindleGen**.
10. Open your MOBI in **Kindle Previewer** to test how it will look on different models of Kindle devices.

That covers it!

Now you know how to create an EPUB file that not only validates to all the EPUB standards, but is also accepted for publication at all of the major eBook distributors around the world. Plus, you know how to convert that EPUB to a MOBI file for the Amazon Kindle, which, at the time this tutorial was created, was the most popular eBook reader on the market

I hope you found this tutorial helpful and informative. Please, share this tutorial with any authors you know and hopefully it will help them too. Good luck publishing those books!