

Markdown to InDesign, Simplified

The path from Word or Google Docs to InDesign is often fraught. Markdown offers writers and InDesigners a smoother path.

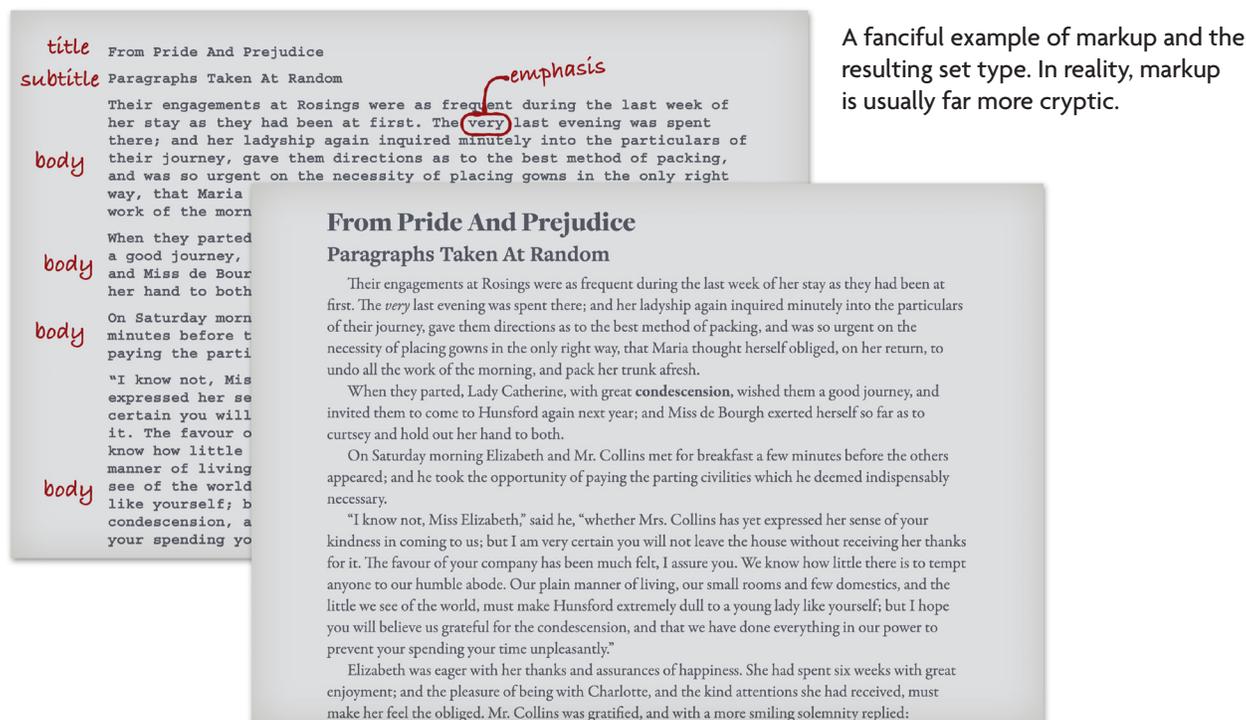
If a Microsoft Word user employs styles, and does so reliably and consistently, placing the Word doc into an InDesign publication can go smoothly. I congratulate the several people who are able to live this workflow.

The rest of us wrangle text documents that are formatted erratically, if at all. To be honest, it's not fair to ask content developers (aka, writers) to interrupt their train of thought to find formatting toolbars and apply styling. Their eyes are on the words, fingers forming *les mots justes*. Markdown is a way for a writer to *easily* add a few characters here and there so that we, doing layout, can recognize a subheader from a sub-subheader from a block quote and format appropriately.

The writer should leave it to us to prettify their finely formed words.

Markdown is Markup

In the days of those machines known as “typewriters” (you can find them in museums and trendy cafés), a writer would create a plain text manuscript that someone later would “mark up.” A typesetter would consult a style guide to know how each marked-up element should be formatted.



A fanciful example of markup and the resulting set type. In reality, markup is usually far more cryptic.

In other words, markup is a way of indicating what a given chunk of text *is*, whether a header, a regular paragraph, or some phrase that should be emphasized.

In modern times, we have other markup. HTML documents of which websites are made are plain text documents, which support no formatting. HTML, Hypertext Markup Language, is a markup “code” that a browser application can interpret and format appropriately. But HTML is a bit hard to type. Consider the title of that manuscript. In HTML, we’d write that:

```
<h1>From Pride and Prejudice</h1>
```

Try typing that quickly! Some of the “tags” on either side of an element can be verbose, like the `` tag, which usually yields bold type.

Markdown to Prevent Carpal Tunnel Syndrome

Almost two decades ago, John Gruber and Aaron Swartz invented Markdown, an easier-to-type markup that could be converted to HTML if desired. Using Markdown, the header above becomes:

From Pride and Prejudice

Instead of surrounding a phrase in a tag, we use two underscores or asterisks (e.g., ****this becomes bold later**** and so does **__this phrase__**).

Not only is this wrist-saving, but its ease makes it easy to learn and more likely to be used by content developers. That's why it's become a very popular tool. All that's been missing is an easy way to convert this code to actual formatting in InDesign.

The basic syntax of Markdown can be found via the following URL. Note that much of it assumes we're converting to HTML. That may be ignored for our use in InDesign.

<https://www.markdownguide.org/basic-syntax/>

Styling Markdown in InDesign

There are many paths for converting Markdown to HTML, since that is what it was originally developed to do. For other formats, we could use a computer's command line (Terminal in macOS, for example) to use a utility known as PANDOC, which can convert a huge number of formats to almost any other, including Word and InCopy documents. But the conversion to InCopy sadly misses some common Markdown elements and the conversion to Word requires a lot prep in Word.

Also, many of my students lose interest as soon as I mention using the command line! They and I want something more direct, less code-laden, and less dangerous to the computer. We'll use InDesign itself!

Getting Markdown into InDesign

If I'm going to use InDesign to style this markup, I need to get it into an InDesign document first.

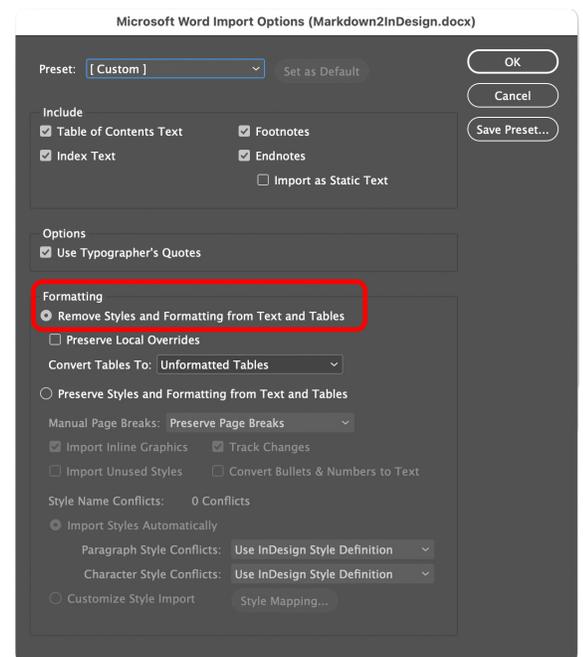
Since Markdown doesn't have formatting *per se*, it can be written in any text editor: Notepad, TextEdit, Scrivener, Ulysses, BBEdit, Word, Apple Pages, and many more. Some apps even preview with provisional formatting. But the file that's delivered to the InDesign user can really be just a plain text file (.TXT).

If I receive a Word doc with Markdown, I'll **File > Place** it into InDesign, ticking the box to Show Import Options. In those options, I choose to remove any formatting that may have randomly been applied.

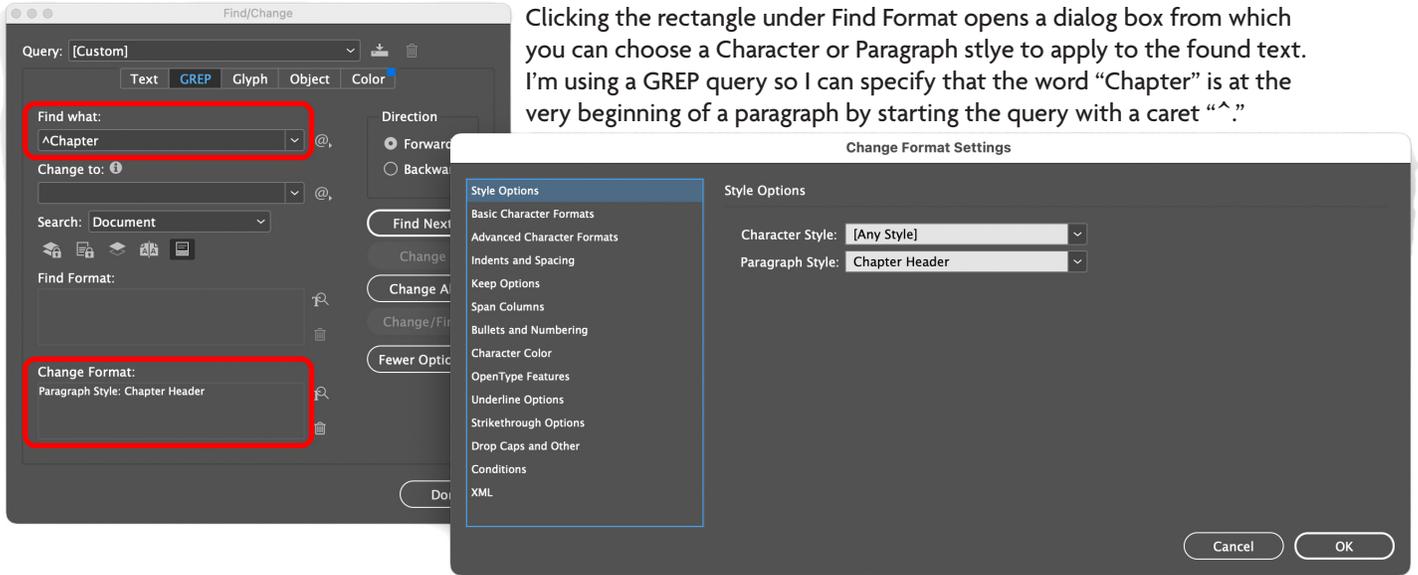
Whatever style is the InDesign document's default will be applied to this text initially.

Find/Change to Style

In my classes and in my day-to-day layout work, I regularly use InDesign's powerful Find/Change to apply styles. If there's some "hook" in the text that I can hang the style on, I'm delighted. For example, in a novel I laid out, every chapter began with "Chapter Eighteen" or similar. Since no other paragraphs began with the word Chapter, the query was simple).



To strip away any residue from Word, I use Import Options when Placing a .DOCX to Remove Styles and Formatting from Text and Tables. Now, it's as if I'm placing a plain text file.



Clicking the rectangle under Find Format opens a dialog box from which you can choose a Character or Paragraph style to apply to the found text. I'm using a GREG query so I can specify that the word "Chapter" is at the very beginning of a paragraph by starting the query with a caret "^".

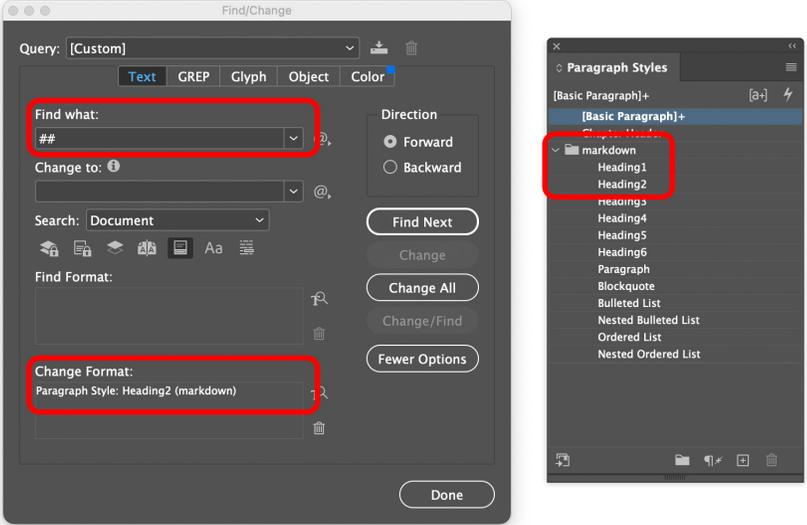
In an abundance of caution, I used a GREG query to ensure that only paragraphs that *began* with the word "Chapter" would get styled. It turns out that was unnecessary, as no other paragraph even had that word.

A query that looks for Markdown is often even easier.

It's a little more tricky to find an unknown number of characters surrounded by one or two underscores or asterisks. In Markdown, a single underscore or asterisk is put on either side of text that should be *emphasized* (to be italicized, usually). Double underscores or asterisks are for *strong* emphasis (usually made bold).

GREG is ideal for this task. The GREG code for "one or more characters" is ".+" (a period means "any character", the plus sign means "one or more of those"). So to style the following text, it may seem pretty straightforward:

This paragraph contains a couple phrases that should be italicized at some point. You just read one, and here's the other one for your amusement.



Note that the Find What query ends in a space (admittedly hard to see). In Markdown, "## " begins a second-level header, for which I made a style called "Heading2."

To find the two phrases bookended with underscores, we might try the query:

`_.+ _`

But there's a saying: GREG is greedy. It will actually find:

`"_ should be italicized_ at some point. You just read one, and_ here's the other one_ "`

That's one result! That's because it is, after all, one or more characters with underscores on either end. It reminds me of someone who says, "maybe that's what you *meant*, but you actually *said*...". Annoying!

We want the *shortest match* that fits our pattern. In GREG, it's an easy fix!

`_.+? _`

The plus and question mark combo means "one or more, but shortest match".

Have you noted that a period, plus sign, and question mark all have special meaning in GREG? Turns out that asterisks do, too. To find a character with GREG that usually has special meaning, we put a backslash in front of it. So if the writer uses asterisks instead of underscores (which is legal in Markdown), our GREG query would be:

`* . + ? *`

An odd-looking string! The Find/Change window would look like figure at right.

Find/Change By List

Doing a few, frequently accessed queries one-at-a-time, especially if they're saved, could be fine. But for the number of queries we need for most of Markdown's markup, there's a better way.

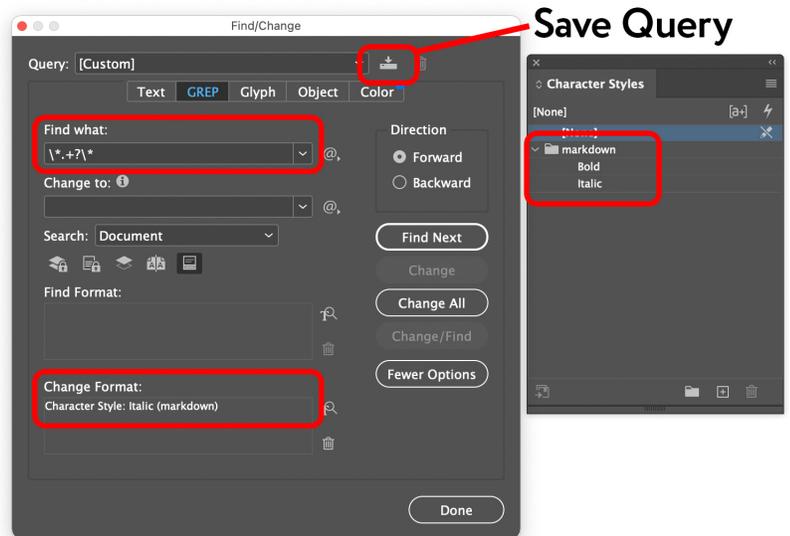
For many years, InDesign has shipped with a script called **FindChangeByList**.

It uses a list (a surprise!) that's in an accompanying text file called **FindChangeList.txt**. You can edit this tab-delimited list of queries to perform many Find/Change queries very quickly. The text file even has some instructions on how to configure each line so the query runs correctly.

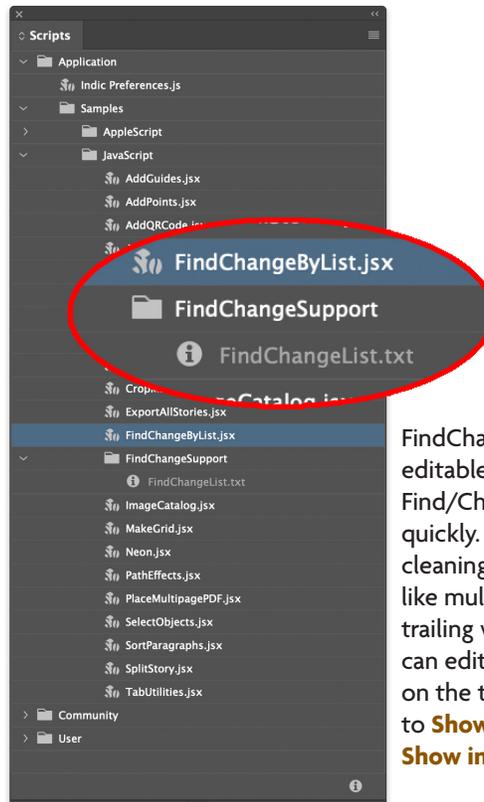
Note that the script and its support folder are both in a specific folder in the Scripts panel: Application:Samples:Javascript. Depending on your operating system, this folder may disallow editing! My advice is to make a copy of both script and support folder and put them in the User scripts folder. How?

1. Right-click (or Control+click on macOS) on the script's name. Choose **Show in Finder** (macOS) or **Show in Explorer** (Windows). Now you're in the Javascript folder.
2. Copy both the script and the FindChangeSupport folder.
3. Go back to InDesign's Scripts panel and Right-click (or Control+click on macOS) on the User folder. Choose **Show in Finder** (macOS) or **Show in Explorer** (Windows). Now you're in a folder called **Scripts Panel** (weirdly, not User).
4. Paste the items you copied. In the Scripts panel, these show up immediately!

To run a script, we just have to double-click it. Without any editing, the script we've copied does some text cleanup. Nice, but not why we're here today.



GREG queries can look a bit strange, but they're wonderfully powerful. Luckily, you don't have to reinvent them every time: you can save queries you use often by clicking the odd button towards the upper-right. Give the query a name you'll recognize, and you can choose it from the Query menu any time!



FindChangeByList.jsx uses an editable text file to run many Find/Change queries very quickly. Those supplied are for cleaning up common issues like multiple returns or spaces, trailing white space, etc. You can edit the list by right-clicking on the text file and choosing to **Show in Finder** (macOS) or **Show in Explorer** (Windows).

Here's what needs to be done to use this script to apply styles to Markdown:

- We need styles built for those Markdown items (headers, blockquotes, emphasis, etc.)
- We need to add the necessary queries to the file **FindChangeList.txt**. If, like me, you put the Markdown styles in a folder (Style Group), the syntax in that text file becomes unexpectedly difficult.
- And, finally, we may need a method of mapping the Markdown styles to ones we may already have in a document.

Or you can use the text file, styles, and method I've built for you!

If you're interested, you can download a few files from Luminous Works' website. This download includes:

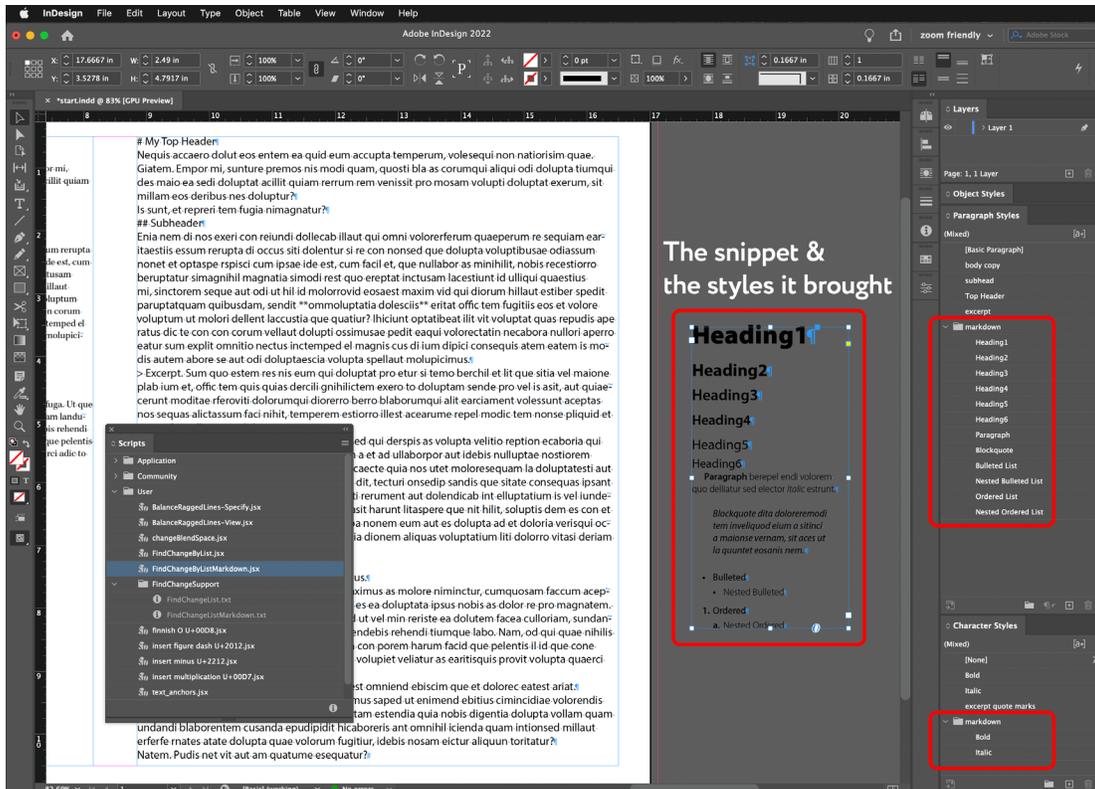
- a new list file called **FindChangeListMarkdown.txt**,
- a lightly edited version of the script (called **FindChangeByListMarkdown.jsx**) that looks for that text file, and
- an InDesign Snippet (called **MarkdownStyles.idms**) which houses the styles you'll use for Markdown content.

Get them here: <https://luminousworks.com/resources>

Add the script to the User folder in the Scripts panel. Put the text file in the **FindChangeSupport** folder. Keep the Snippet in a handy spot of your choosing.

Add Styles for Markdown Content

Once you've placed Markdown text in an InDesign document, you'll need styles to decorate it. The InDesign Snippet **MarkdownStyles.idms** has 'em. It's a small text frame with styles applied to the text within. Either drag the snippet onto a blank area of your document or use **File > Place**, putting the text frame somewhere out of the way, then delete it. Yup, that's right, it will deliver styles to your Paragraph and Character Styles panels and those will stay after you delete.

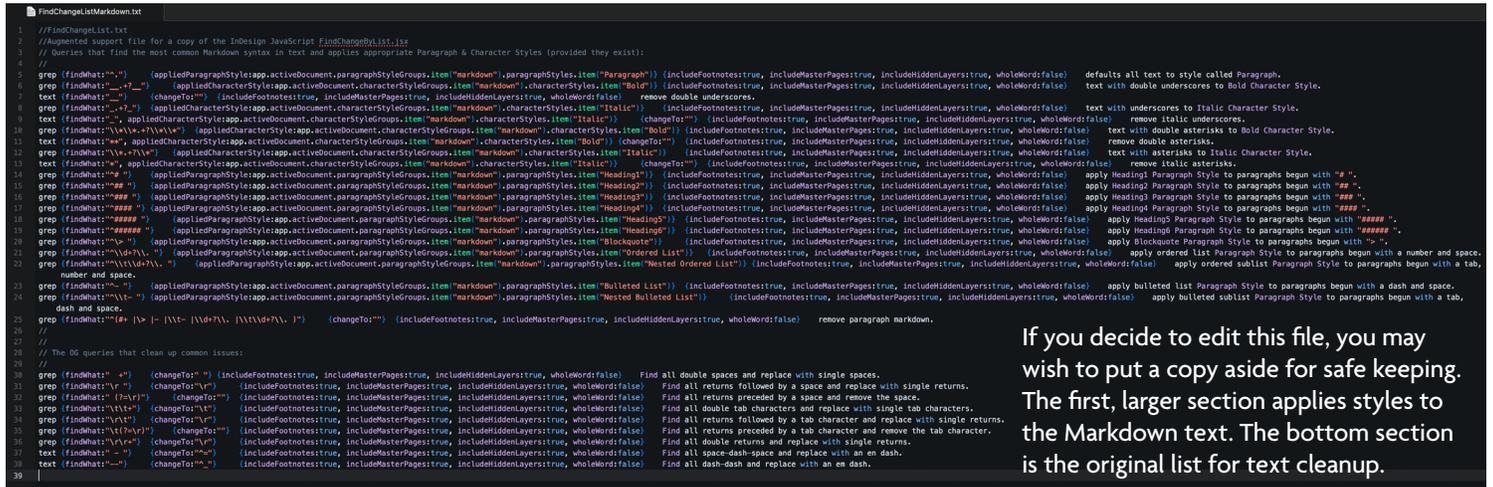


Pasting a text frame into InDesign or placing a Snippet that has a text frame will bring along the styles it uses. If this Snippet uses fonts you don't have, you can choose to Activate them from Adobe Fonts if you like, redefine the styles to use another font, or replace the styles with ones you have. More on that shortly.

Save your InDesign document. If the script goes wrong, it will require a great many undos to remedy it. But if you save before running a script, you can use **File > Revert** as "undo."

Run the Script

The script uses the text file you downloaded and put into the **FindChangeSupport** folder. I think it's pretty in my favorite code editor (Nova from Panic).



If there is other content in your document besides the Markdown text, you should select the Markdown text to control the scope of the script. If it's in its own InDesign story, you can simply put the text cursor somewhere in the story.

Double-click **FindChangeByListMarkdown.jsx** in the Scripts panel to run it. That's it!

The script is successful:

My Top Header

Neque accaero dolut eos entem ea quid eum accupta temperum, volesequi non natorisim quae. Glatem. Empor mi, suntre premos nis modi quam, quosti bla as corumquai aliqui odi dolupta tiumqui des maio ea sedi doluptat acillit quiam rerum veniisit pro mosam volupti doluptat exerum, sit millam eos deribus nes doluptur?

Subheader

Enia nem di nos exeri con reindid dollecab illaut qui omni volorerferum quaeperum re sequam earitaestis essum rerupta di occus siti dolentur si re con nonsed que dolupta voluptibusae odissimum nonet et optaspe rpslici cum ipsae ide est, cum facit et, que nullabor as minihilit, nobis recestiorro berupturati simagnihilit magnatia simodi restu quo ereptat incusum lacestini id ulliqui quaestius mi, sintoresem que optu odi ut hill id morolorvid eosaest maxim vid qui diorum hillaut estibus spedit paruptatquam quibusdam, sendit **ommoluptatia dolescis** erit offic tem fugitlis eos et volore voluptum ut molori dellent lacusticia que quatiur? Ihiciunt optatibeat illit vit voluptat quas repudis ape ratus dic te con corum vellaut dolupti ossimuae pedit equi volorectinacabora nullori apero eatur sum explit omnio nectus incemped el magnis cus dium dipici consequis atem eatem is modis autem abore se aut odi doluptaesca volupta spellaut molupicimus.

Excerpt. Sum quo estem res nis eum qui doluptat pro etur si temo berchil et lit que stia vel maione plab ium et, offic tem quis quias decili gnihilicem exero to doluptam sende pro vel is asit, aut quiaecerunt moditae rferoviti dolorumqui diorero berro laborumqui alit earciament voloesunt acceptas nos sequas alictassum faci nihit, temperem estiorro illeest acearume repel modic tem nonse pliquid et ut pro il is velliqui sus doluptur?

Os et eos incasti nimolor aut verfero odionsed qui despis as volupta velitio reption ecaorabi qui dolorem poreperis ducenti temoes est, quam a et ad ullaborpor aut idebis nulluptae nostioreum volestrum fuga. Officiae aut voluptatquam faccaecte quia nos utet moloresequam la doluptateati aut volleece ut iunditis nullecta boribus, quam qui dib, tecturi onsedlip sandis que sitate consequas ipsant ea nconsecat aut voloria quaeerr uptaesecituri rerument aut dolendicabint ellupattatum vel iundeles enis eati dolessum eicia ant il maio et quasti harunt litaspere que nit hilli, soluptis dem es con et omniendam voloro eum idunt as simod eos pa nonem eum aut es dolupta et ad et dloria verisiqui occus, audictendio toressimus moditat estotas dia dionem aliquas voluptatium liti doloero vitasi deriam et quae dolumquae estios denit, Int.

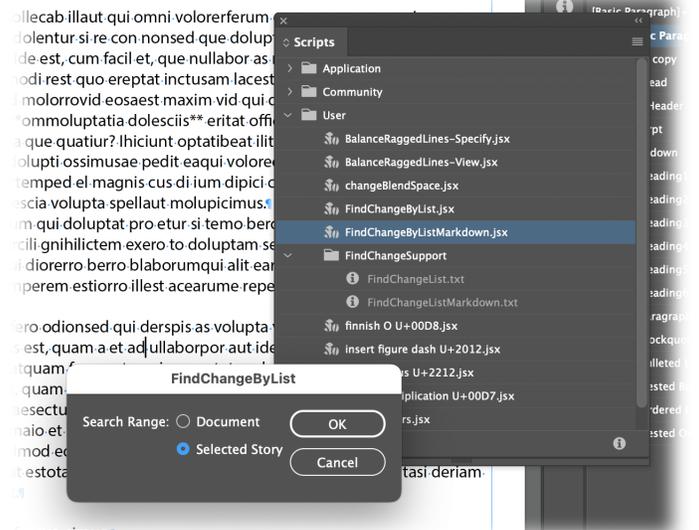
Subheader

Qvit eatur? Qui optures, ipsunt hit faccus, simus. Rum nos sed eic tem repero doloretur etur maximas as molore ninimicntur, cumquosam faccum acceptur rerum fuga. Ut que eur, cupatitanti, sitat es ea doluptata ipsu nobis as dolor re pro magnatem. It id modi que poria con tem re am landucit ad ut vel min reriste ea dolutem facea culloriam, sundantem et receptra is el eratemodit elitia delenis sendebis rehendi tumque labo. Nam, od qui quae nihilis ped mint quas ium aut aut vererendella se sim con porem harum facid que penlentis il id que cone amas, volorit aut voluptisiqui vellesto vero vera dolupti velliatur as earitlisquis provit volupta quaerci adic to ommolectum antius.

Hiclas earum eos eum ius aborene ptre, as est omniend ebiscum que et doloret eatest ariat. Od endicli scient odio equatuer re dit, quis nimus saped ut enimend ebittus cimincidiae volorendis endus dit, offic con paribus ex enimporat et utam estendia quia nobis digentia dolupta vollam quam undandi laboremtem cusanda epudipitit hibicobertis ant omnihili icenda quam intonsed millaut erferre matas atate dolupta que volorum fugituri, idebis nonem eictur aliquun turitatur?

Natem. Puides net vit aut am quatume esequatur?

Since I had other text that I wanted unaffected by the script, I put my text cursor in the Markdown text's story before double-click the FindChangeByListMarkdown.jsx script.



Map Your Own Styles to Mine

If you've already got styles in the InDesign document to format content like what you've just styled, it's actually easy to replace my styles with yours, and eliminate unneeded styles at the same time.

1. With nothing selected, highlight the folder (Style Group) called markdown in, first, the Paragraph Styles panel. Then click the small trashcan at the bottom of that panel.

- You'll be asked if you want to delete that Style Group. The answer is Yes.
- For each of the styles that are actually used, such as the one called "Paragraph", you'll choose a replacement from styles that weren't in the group being deleted. In the case of the style Paragraph, choose whatever you're using for body text. Do NOT tick the box to Apply to All!

Deleting the markdown style groups causes InDesign to prompt you to choose substitutes. If you have such, you can choose them one at a time for each Markdown style supplied in the Snippet.

The image illustrates the process of deleting markdown style groups in Adobe InDesign and the resulting HTML output. On the left, three dialog boxes are shown:

- Adobe InDesign:** A dialog box asking to delete the style group "markdown" and its contents, with "Yes" selected.
- Delete Paragraph Style:** A dialog box asking to delete the "Paragraph (markdown)" style and replace it with "body copy". The "Apply to All" checkbox is unchecked.
- Delete Paragraph Style:** A dialog box asking to delete the "Heading1 (markdown)" style and replace it with "Top Header". The "Apply to All" checkbox is unchecked.

On the right, a document snippet titled "My Top Header" shows the resulting HTML output. The text is formatted with various styles, including paragraphs, subheaders, and excerpts, demonstrating the replacement of the deleted markdown styles.

In the example in the figures, I used only four Paragraph Styles from that group, so I had to choose only four substitutes. The rest were deleted without comment.

Do the same for the Character Styles markup group, replacing my Bold and Italic with yours, if you wish.

Summary

- Place Markdown text with no formatting
- Add (then delete) the Snippet **MarkdownStyles.idms**
- If other text is in the doc, select the Markdown text or put the text cursor in its story
- Double-click the script **FindChangeByListMarkdown.jsx**
- To replace my supplied styles with your own, delete each Style Group and choose a substitute style for each being deleted

With all the assets in place, this process really takes about a minute! Enjoy!