

# MARKDOWN SYNTAX

Let's face it: Writing content for the web is tiresome. WYSIWYG editors help alleviate this task, but they generally result in horrible code, or worse yet, ugly web pages.

**Markdown** is a better way to write **HTML**, without all the complexities and ugliness that usually accompanies it.

Some of the key benefits are:

1. Markdown is simple to learn, with minimal extra characters so it's also quicker to write content.
2. Less chance of errors when writing in Markdown.
3. Produces valid HTML output.
4. Keeps the content and the visual display separate, so you cannot mess up the look of your site.
5. Write in any text editor or Markdown application you like.
6. Markdown is a joy to use!

John Gruber, the author of Markdown, puts it like this:

*The overriding design goal for Markdown's formatting syntax is to make it as readable as possible. The idea is that a Markdown-formatted document should be publishable as-is, as plain text, without looking like it's been marked up with tags or formatting instructions. While Markdown's syntax has been influenced by several existing text-to-HTML filters, the single biggest source of inspiration for Markdown's syntax is the format of plain text email.*

— John Gruber

Without further delay, let us go over the main elements of Markdown and what the resulting HTML looks like:

## Tip

Bookmark this page and the [official Commonmark reference](#) for easy future reference!

## Paragraphs

In Markdown your content usually spans the whole available document width. This is called a block. Blocks are always separated by whitespace to their adjacent blocks in the resulting document.

Any text not starting with a special sign is written as normal, plain text paragraph block and must be separated to its adjacent blocks by empty lines.

Lorem ipsum dolor sit amet, graecis denique ei vel, at duo primis mandamus.

Et legere occurreret pri, animal tacimates complectitur ad cum. Cu eum inermis inimicus efficiendi. Labore officiis his ex, soluta officiis concludaturque ei qui, vide sensibus vim ad.

## Result

Lorem ipsum dolor sit amet, graecis denique ei vel, at duo primis mandamus.

Et legere occurreret pri, animal tacimates complectitur ad cum. Cu eum inermis inimicus efficiendi. Labore officiis his ex, soluta officiis concludaturque ei qui, vide sensibus vim ad.

## Headings

A good idea is to structure your content using headings and subheadings. HTML-headings from `h1` through `h6` are constructed with a `#` for each level.

In Hugo you usually don't use `h1` as this is generated by your theme and you should only have one such element in a document.

```
# h1 Heading
## h2 Heading
### h3 Heading
#### h4 Heading
##### h5 Heading
##### h6 Heading
```

### Result

# H1 HEADING

## h2 Heading

### h3 Heading

#### h4 Heading

##### h5 Heading

###### h6 Heading

## Horizontal Rules

To further structure your content you can add horizontal rules. They create a “thematic break” between paragraph blocks. In Markdown, you can create it with three consecutive dashes `---`.

Lorem ipsum dolor sit amet, graecis denique ei vel, at duo primis mandamus.

```
---
```

```
Et legere occurreret pri, animal tacimates complectitur ad cum. Cu eum inermis inimicus efficiendi. Labore officiis his ex, soluta officiis concludaturque ei qui, vide sensibus vim ad.
```

#### Result

Lorem ipsum dolor sit amet, graecis denique ei vel, at duo primis mandamus.

Et legere occurreret pri, animal tacimates complectitur ad cum. Cu eum inermis inimicus efficiendi. Labore officiis his ex, soluta officiis concludaturque ei qui, vide sensibus vim ad.

## Text Markers

### Bold

You can show importance of a snippet of text with a heavier font-weight by enclosing it with two asterisks `**`.

```
I am rendered with bold text
```

#### Result

I am rendered with **bold text**

### Italics

You can emphasize a snippet of text with italics by enclosing it with underscores `_`.

```
I am rendered with italicized text
```

#### Result

I am rendered with *italicized text*

## Strikethrough

In GFM (GitHub Flavored Markdown) you can do strikethroughs by enclosing text with two tildes `~~`.

```
~~Strike through this text~~
```

#### Result

~~Strike through this text~~

## Text substitution

This Markdown dialect supports an extension to combine multiple punctuation characters to single typographic entities. This will only be applied to text outside of code blocks or inline code.

Double quotes ``"``` and single quotes ``'``` of enclosed text are replaced by `***double curly quotes***` and `***single curly quotes***`.

Double dashes ``--`` and triple dashes ``---`` are replaced by en-dash `***-***` and em-dash `***-***` entities.

Double arrows pointing left ``<<`` or right ``>>`` are replaced by arrow `***<<***` and `***>>***` entities.

Three consecutive dots ``...`` are replaced by an ellipsis `***...***` entity.

### Result

Double quotes `"` and single quotes `'` of enclosed text are replaced by **“double curly quotes”** and **‘single curly quotes’**.

Double dashes `--` and triple dashes `---` are replaced by en-dash `–` and em-dash `—` entities.

Double arrows pointing left `<<` or right `>>` are replaced by arrow `«` and `»` entities.

Three consecutive dots `...` are replaced by an ellipsis `...` entity.

## Lists

### Unordered

You can write a list of items in which the order of the items does not explicitly matter.

It is possible to nest lists by indenting an item for the next sublevel.

You may use any of `-`, `*` or `+` to denote bullets for each list item but should not switch between those symbols inside one whole list.

- Lorem ipsum dolor sit amet
- Consectetur adipiscing elit
  - Vestibulum laoreet porttitor sem
  - Ac tristique libero volutpat at
- Nulla volutpat aliquam velit
  - Phasellus iaculis neque
  - Purus sodales ultricies
- Faucibus porta lacus fringilla vel

### Result

- Lorem ipsum dolor sit amet

- Consectetur adipiscing elit
  - Vestibulum laoreet porttitor sem
  - Ac tristique libero volutpat at
- Nulla volutpat aliquam velit
  - Phasellus iaculis neque
  - Purus sodales ultricies
- Faucibus porta lacus fringilla vel

## Ordered

You can create a list of items in which the order of items does explicitly matter.

It is possible to nest lists by indenting an item for the next sublevel.

Markdown will automatically number each of your items consecutively. This means, the order number you are providing is irrelevant.

- ```
1. Lorem ipsum dolor sit amet
3. Consectetur adipiscing elit
    1. Integer molestie lorem at massa
    7. Facilisis in pretium nisl aliquet
99. Nulla volutpat aliquam velit
    1. Faucibus porta lacus fringilla vel
    1. Aenean sit amet erat nunc
17. Eget porttitor lorem
```

### Result

1. Lorem ipsum dolor sit amet
2. Consectetur adipiscing elit
  1. Integer molestie lorem at massa
  2. Facilisis in pretium nisl aliquet
3. Nulla volutpat aliquam velit
  1. Faucibus porta lacus fringilla vel
  2. Aenean sit amet erat nunc
4. Eget porttitor lorem

## Tasks

In GFM (GitHub Flavored Markdown) you can add task lists resulting in checked or unchecked non-clickable items

- ```
- [x] Basic Test
- [ ] More Tests
  - [x] View
  - [x] Hear
  - [ ] Smell
```

### Result

- ☒ Basic Test
- ☐ More Tests

- ☒ View
- ☒ Hear
- ☐ Smell

## Definitions

This Markdown dialect supports an extension to add definition lists. Definition lists are made of terms and definitions of these terms, much like in a dictionary.

A definition list in Markdown Extra is made of a single-line term followed by a colon and the definition for that term. You can also associate more than one term to a definition.

If you add empty lines around the definition terms, additional vertical space will be generated. Also multiple paragraphs are possible

```
Apple
: Pomaceous fruit of plants of the genus Malus in the family Rosaceae.
: An American computer company.

Orange
: The fruit of an evergreen tree of the genus Citrus.

    You can make juice out of it.
: A telecommunication company.

    You can't make juice out of it.
```

### Result

```
Apple
  ◦ Pomaceous fruit of plants of the genus Malus in the family Rosaceae.
  ◦ An American computer company.

Orange
  ◦ The fruit of an evergreen tree of the genus Citrus.
    You can make juice out of it.
  ◦ A telecommunication company.
    You can't make juice out of it.
```

## Code

### Inline Code

Inline snippets of code can be wrapped with backticks ```.

In this example, `<div></div>` is marked as code.

### Result

In this example, `<div></div>` is marked as code.

## Indented Code Block

A simple code block can be generated by indenting several lines of code by at least two spaces.

Be impressed by my advanced code:

```
// Some comments
line 1 of code
line 2 of code
line 3 of code
```

### Result

Be impressed by my advanced code:

```
// Some comments
line 1 of code
line 2 of code
line 3 of code
```

## Fenced Code Block

If you want to gain more control of your code block you can enclose your code by at least three backticks ````` a so called fence.

In GFM (GitHub Flavored Markdown) you can also add a language specifier directly after the opening fence, ````js`, and syntax highlighting will automatically be applied according to the selected language in the rendered HTML.

See [Code Highlighting](#) for additional documentation.

```
```js
grunt.initConfig({
  assemble: {
    options: {
      assets: 'docs/assets',
      data: 'src/data/*.json,yml',
      helpers: 'src/custom-helpers.js',
      partials: ['src/partials/**/*.hbs,md']
    },
    pages: {
      options: {
        layout: 'default.hbs'
      },
      files: {
        './': ['src/templates/pages/index.hbs']
      }
    }
  }
});
```
```

### Result

```
grunt.initConfig({
  assemble: {
    options: {
      assets: 'docs/assets',
```

```

    data: 'src/data/*.json,yml',
    helpers: 'src/custom-helpers.js',
    partials: ['src/partials/**/*.hbs,md']
  },
  pages: {
    options: {
      layout: 'default.hbs'
    },
    files: {
      './': ['src/templates/pages/index.hbs']
    }
  }
}
};

```

## Tables

In GFM (GitHub Flavored Markdown) you can create tables by adding pipes as dividers between each cell, and by adding a line of dashes (also separated by bars) beneath the header. Note that the pipes do not need to be vertically aligned.

```

Option	Description
data	path to data files to supply the data that will be passed into templates.
engine	engine to be used for processing templates. Handlebars is the default.
ext	extension to be used for dest files.

```

### Result

| Option | Description   |
|--------|---|
| data   | path to data files to supply the data that will be passed into templates. |
| engine | engine to be used for processing templates. Handlebars is the default.    |
| ext    | extension to be used for dest files.                                      |

## Aligned Columns

Adding a colon on the left and/or right side of the dashes below any heading will align the text for that column accordingly.

```

Option	Number	Description
data	1	path to data files to supply the data that will be passed into templates.
engine	2	engine to be used for processing templates. Handlebars is the default.
ext	3	extension to be used for dest files.

```

### Result

| Option | Number | Description |
|--------|--------|-------------|
|--------|--------|-------------|



| Option | Number | Description   |
|--------|--------|---|
| data   | 1      | path to data files to supply the data that will be passed into templates. |
| engine | 2      | engine to be used for processing templates. Handlebars is the default.    |
| ext    | 3      | extension to be used for dest files.                                      |

## Blockquotes

For quoting blocks of content from another source within your document add `>` before any text you want to quote.

Blockquotes can also be nested.

```
> Donec massa lacus, ultricies a ullamcorper in, fermentum sed augue. Nunc augue augue,
aliquam non hendrerit ac, commodo vel nisi.
>
> > Sed adipiscing elit vitae augue consectetur a gravida nunc vehicula. Donec auctor odio non
est accumsan facilisis. Aliquam id turpis in dolor tincidunt mollis ac eu diam.
>
> Mauris sit amet ligula egestas, feugiat metus tincidunt, luctus libero. Donec congue finibus
tempor. Vestibulum aliquet sollicitudin erat, ut aliquet purus posuere luctus.
```

### Result

*Donec massa lacus, ultricies a ullamcorper in, fermentum sed augue. Nunc augue augue, aliquam non hendrerit ac, commodo vel nisi.*

*Sed adipiscing elit vitae augue consectetur a gravida nunc vehicula. Donec auctor odio non est accumsan facilisis. Aliquam id turpis in dolor tincidunt mollis ac eu diam.*

*Mauris sit amet ligula egestas, feugiat metus tincidunt, luctus libero. Donec congue finibus tempor. Vestibulum aliquet sollicitudin erat, ut aliquet purus posuere luctus.*

## Links

### Autolink

In GFM (GitHub Flavored Markdown) absolute URLs will automatically be converted into a link.

This is a link to `https://example.com`.

### Result

This is a link to <https://example.com>.

### Basic Link

You can explicitly define links in case you want to use non-absolute URLs or want to give different text.

```
[Assemble](http://assemble.io)
```

#### Result

[Assemble](http://assemble.io)

## Link with Tooltip

For even further information, you can add an additional text, displayed in a tooltip on hovering over the link.

```
[Upstage](https://github.com/upstage/ "Visit Upstage!")
```

#### Result

[Upstage](https://github.com/upstage/)

## Link References

Links can be simplified for recurring reuse by using a reference ID to later define the URL location. This simplifies writing if you want to use a link more than once in a document.

```
[Example][somelinkID]
[somelinkID]: https://example.com "Go to example domain"
```

#### Result

[Example](https://example.com)

## Footnotes

Footnotes work mostly like reference-style links. A footnote is made of two things, a marker in the text that will become a superscript number and a footnote definition that will be placed in a list of footnotes.

Usually the list of footnotes will be shown at the end of your document. If we use a footnote in a notice box it will instead be listed at the end of its box.

Footnotes can contain block elements, which means that you can put multiple paragraphs, lists, blockquotes and so on in a footnote. It works the same as for list items, just indent the following paragraphs by four spaces in the footnote definition.

```
That's some text with a footnote[1]
```

```
[^1]: And that's the footnote.
```

```
That's some more text with a footnote.[^someid]

[^someid]:
    Anything of interest goes here.

    Blue light glows blue.
```

#### Result

That's some text with a footnote<sup>1</sup>

That's some more text with a footnote.<sup>2</sup>

- 
1. And that's the footnote. ↵
  2. Anything of interest goes here.
 

Blue light glows blue. ↵

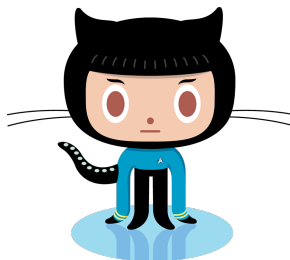
## Images

### Basic Images

Images have a similar syntax to links but include a preceding exclamation mark.

```
! [Spock] (https://octodex.github.com/images/spocktocat.png)
```

#### Result



### Image with Tooltip

Like links, images can also be given a tooltip.

```
! [Picard] (https://octodex.github.com/images/jean-luc-picat.jpg "Jean Luc Picard")
```

#### Result





## Image References

Images can also be linked by reference ID to later define the URL location. This simplifies writing if you want to use an image more than once in a document.

```
![[La Forge]][laforge]
```

```
[laforge]: https://octodex.github.com/images/trekkie.jpg "Geordi La Forge"
```

👁 Result



## Further Image Formatting

This theme allows additional non-standard formatting by setting query parameter at the end of the image URL.

## Resizing

Add query parameter `width` and/or `height` to the link image to resize the image. Values are CSS values (default is `auto`).

```
![[Minion]](https://octodex.github.com/images/minion.png?width=20vw)
```

👁 Result



```
! [Minion] (https://octodex.github.com/images/minion.png?height=50px)
```

👁 Result



```
! [Minion] (https://octodex.github.com/images/minion.png?height=50px&width=40vw)
```

👁 Result



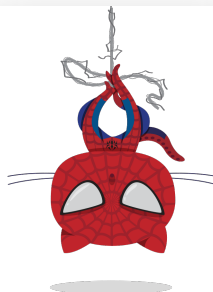
## CSS Classes

Add a query parameter `classes` to the link image to add CSS classes. Add some of the predefined values or even define your own in your CSS.

### Shadow

```
! [Spidertocat] (https://octodex.github.com/images/spidertocat.png?classes=shadow)
```

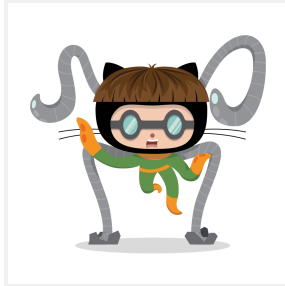
👁 Result



## Border

```
![[Dr0ctocat]](https://octodex.github.com/images/droctocat.png?classes=border)
```

### Result



## Left

```
![[Supertocat]](https://octodex.github.com/images/okal-eltocat.jpg?classes=left)
```

### Result



## Right

```
![[Riddlocat]](https://octodex.github.com/images/riddlocat.jpg?classes=right)
```

### Result



## Inline

```
! [Spidertocat] (https://octodex.github.com/images/spidertocat.png?classes=inline)
! [DrOctocat] (https://octodex.github.com/images/droctocat.png?classes=inline)
! [Supertocat] (https://octodex.github.com/images/okal-eltocat.jpg?classes=inline)
! [Riddlocat] (https://octodex.github.com/images/riddlocat.jpg?classes=inline)
```

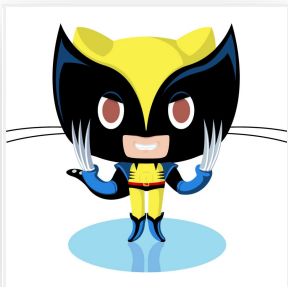
### Result



## Combination

```
! [X-tocat] (https://octodex.github.com/images/xtocat.jpg?classes=shadow,border,left)
```

### Result



## Lightbox

Add the query parameter `lightbox=false` to the image link to disable the lightbox.

```
! [Homercat] (https://octodex.github.com/images/homercat.png?lightbox=false)
```

### Result



