GitHub Markdown Cheat Sheet

|  |
| --- |
| **Headers** |

# This is an <h1> tag

## This is an <h2> tag

###### This is an <h6> tag

|  |
| --- |
| **Emphasis** |

\*This text will be italic\*

\_This will also be italic\_

\*\*This text will be bold\*\*

\_\_This will also be bold\_\_

\*You \*\*can\*\* combine them\*

|  |
| --- |
| **Lists** |

Unordered

\* Item 1

\* Item 2

\* Item 2a

\* Item 2b

Ordered

1. Item 1

2. Item 2

3. Item 3

\* Item 3a

\* Item 3b

|  |
| --- |
| **Tables** |

You can create tables by assembling a list of words and dividing them with hyphens - (for the first row), and then separating each column with a pipe | :

First Header | Second Header

------------ | -------------

Content cell 1 | Content cell 2

Content column 1 | Content column 2

|  |
| --- |
| **Blockquotes** |

As Grace Hopper said:

> I’ve always been more interested

> in the future than in the past.

|  |
| --- |
| **Backslash Escapes** |

Markdown allows you to use backslash escapes to generate literal characters which would otherwise have special meaning in Markdown’s formatting syntax.

\\*literal asterisks\\*

Markdown provides backslash escapes for

the following characters:

\ backslash () parentheses

` backtick # hash mark

\* asterisk + plus sign

\_ underscore - minus sign (hyphen)

{} curly braces . dot

[] square brackets ! exclamation mark

|  |
| --- |
| **Links** |

http://github.com - automatic!

[GitHub](http://github.com)

|  |
| --- |
| **Images** |

![GitHub Logo](/images/logo.png)

Format: ![Alt Text](url)

|  |
| --- |
| **Task Lists** |

- [x] this is a complete item

- [ ] this is an incomplete item

- [x] @mentions, #refs, [links](), \*\*formatting\*\*, and <del>tags</del> supported

- [x] list syntax required (any unordered or ordered list

supported)

|  |
| --- |
| **Emoji** |

To see a list of every image supported, check out

www.emoji-cheat-sheet.com

GitHub supports emoji!

:+1: :sparkles: :camel: :tada:

:rocket: :metal: :octocat:

|  |
| --- |
| **Fenced Code Blocks** |

Markdown converts text with four leading spaces into a code block; you can wrap your code with ``` to create a code block without the leading spaces. Add an optional language identifier and your code will get syntax highlighting.

```javascript

function test() {

console.log("look ma’, no spaces");

}

```