Paragraphs and Breaks

Markdown accepts text on consecutive lines as a hard-wrapped paragraph.

Put a blank line in between to start a new graph.

If you want a break: end your line with two spaces.

Hard to see in print, but it's there.

Headers

<table>
<thead>
<tr>
<th>h1</th>
<th>h2</th>
<th>Setext-style</th>
</tr>
</thead>
<tbody>
<tr>
<td># h1</td>
<td></td>
<td>atx-style</td>
</tr>
<tr>
<td>## h2</td>
<td></td>
<td>opening hashes determine header level</td>
</tr>
<tr>
<td>###### h6 ##</td>
<td></td>
<td>closing hash(es) optional</td>
</tr>
</tbody>
</table>

Blockquotes

> Blockquote.
> > Second paragraph in the blockquote, nested.
> > ## This is an h2 in a blockquote

Phrase Emphasis

*em* or _em_ or un*fucking*believable
**strong** or __strong__
***em and strong*** or ___em and strong___

Lists

* Candy. or + Candy. or - Candy.
* Gum. + Gum. - Gum.

1. Ordered          > start the list with number 1
2. List             > otherwise number order doesn’t matter
3. Items

* A list item.
  With multiple paragraphs.
  > And a blockquote
* Another List item with a hard wrapped 2nd line.

  10 PRINT "and a code block"

Horizontal Rules

--- or *** or ___ > on a line by itself

Links

An [inline link](http://xrl.us/ "optional title").

A [reference link][id]. > [id] defined elsewhere

[id]: http://example.com/ "optional title"
or [id]: <http://example.com/> (optional title) or [id]: http://example.com/longish/path/to/resource "optional title"

[Google][] > implicit link name shortcut
[Google]: http://google.com

<http://example.com/> > creates linked URL
<address@example.com> > creates encoded mailto

Images

![Alt text](/path/to/img.jpg "Optional title")

![Alt text][id] > reference-style
[id]: url/to/image "Optional title attribute"

Code

Backticks format and auto-escape &, < and > in code like `<blink>`, `&mdash;` and `&#8212;`.

Preformatted code is also easy.&trade;
<strong>Just indent 4 spaces.</strong>

Double-backticks delimit literal backticks:
```
```
```
```
```
A backtick-delimited string: ``` `foo` ```

Escaping

Backslash (\) escapes the following characters:
\ ` * _ { } [ ] ( ) # + - . !
\*this text is surrounded by literal asterisks*```