

Guide for XHTML

--version 0.9.9 / writer: @devster / last edit: Fri Apr 25 2025 15:30:10 GMT+0700 (Indochina Time)

- [Summary](#)
- [How to use](#)
- [Header](#)
- [Paragraph](#)
- [Span](#)
- [Bold](#)
- [Italic](#)
- [Underline](#)
- [Mark](#)
- [Bullet List](#)
- [Number List](#)
- [Link](#)
- [Image](#)
- [Small Text](#)
- [Block](#)
- [Code Block](#)
- [Line](#)
- [Variable](#)
- [Attribute](#)
- [Block Quote](#)
- [Quote](#)
- [xTag](#)
- [Table](#)
- [Direct html](#)
- [Default classes](#)
- [Final Note](#)

Attention

This guide is good to read in XHTML file because you see how to do it in the plain text. But you may switch to see html to realize the conversion.

Summary

The XHTML is a very simple and clean markup language. It allows you to make your document in plain text style, but easily to convert into html. So you can read both in XHTML or HTML formats.

Example of an xtml file

```
# This is my title
```

```
This is my paragraph which normally putting some interesting story into this part so  
people can read it. Humanity uses text and languages to communicates among our society.  
Go up top The languages and text and the way we writing are the art, and brain of human being.
```

Following is a list:

- . Thailand
- . Singapore
- . Malaysia

That's it. This is a simple form when people write something.

And when you convert this into html you'll see something similar to this guide. So the xtml is nice to both worlds, the plain text, and the web.

How to use

You write your document in `your-doc.xml` file which is a general text file, can be read by any text editor. But if you like to read in html format, you just do:

```
$ node xtml your-doc.xml output.html
```

This way you get html file in which you can read in that style. So the picture, links and those stuff work in the html file.

Title

We have *Title* and *Sub Title* here. The Title takes `#` and it gives you the `<h1>` in html. The Sub Title takes `##` and is the `<h2>` in html.

```
# This is the biggest title or <h1>
## this is the second title or <h2>
### This is heading 3 or <h3>
```

! the `###` just added to xtml feature.

We have made little styles to the title: the title 1 is `<h1>`, the title 2 is `<h2>` and being *italic* and title 3 `<h3>` with underlined.

Paragraph

For paragraph you don't have to do anything so just type your story in each paragraph. Normally, you may put blank lines above and below. That's it.

Just type your words and we regard it as the paragraph right away (or `<p>` in html)

Span

In html we have `...` the xtml also has things like `{this is your span or part of text within your paragraph}` . Please notice that there's no gap between the `{` and the `}` this allows xtml to understand this is the `` for you.

[Go up top](#)

If you want to set `id` or `class` you can do so like: `{ssssssssssss ; #id.class}` . So just put the `;` and then set id and class as you want.

Bold

Put 2 stars for the **bold text** as:

puts 2 stars to `**bold this text**`

Italic

Uses 2 slashes for *italic text* like:

the text after this is `//italic cccccccc//`

Underline

Double underscores for underline your text:

following text is to be `__underlined ssssssss__`

Mark

You can mark or highlight text by putting `[m] [m]` then you get something like `<mark> </mark>` in html.

Bullet List

Put the `.` and list your points along down until you finish, just put a blank line after it and so you get the bullet lists.

```
. Thailand  
. Singapore  
. Malaysia
```

In html will be:

- Thailand
 - Singapore
 - Malaysia
-

Number List

[Go up top](#)

Put number, a space, and list your points and end with a blank line then you get the number list or in html is the `` .

```
1 Thailand
2 Singapore
3 Malaysia
```

In html:

```
1. Thailand
2. Singapore
3. Malaysia
```

Link

We have 2 styles of link. One with the description, this one not showing the URL so the reader won't be headache. Another one, no description just go straight the URL. Do like this:

```
[[ click this ; $url = http://nex-era.xyz ]]
[[ $url = http://nex-era.xyz ]]
```

[click this](#)
<http://nex-era.xyz>

The first one we defined the text for reader to click and it brings her to another page or part of this page. Another one is we don't specify the clickable text just define the URL for them to click.

Personally, I'm preferring to just show the reader URL so she knows what she is about to click.

Image



Simple put in a brackets like above and define where your pic file is. It will show in width 100% by default. So picture above comes by:

```
[[ $img = lisa.png ]]
```

The image tag above it will generally put in the `<p>...</p>` so you'll get some margins the browser preset for the `<p>` . To prevent it staying inside the `<p>` you can just put in html tag `` like:

```

```

This way you get pure picture, with no margin and all that. And you can also style it just like html file.

Small Text

Surrounding the text with the `[s]` you'll get small text. The small text mark can govern the italice or other format inside, see below.

[Go to top](#) is your small text[s]

```
[s]//this is italic and then small//[s]
[s]**you can also have bold and small together too**[s]
```

Your codes will show in html like:

```
this is your small text
this is italic and then small
you can also have bold and small together too
```

Block

We can do things like `<div>` as in html by using the double square brackets like below.

```
[[
## your heading 1 or 2 can be in this block

paragraph and other formatings can be here, bold, italic, bullet, numbering list.

]]
```

and you'll get these in your html file:

```
<div>
  <h2>ssssssssssssssssssss</h2>
  <p>ssssssssssssssssssssssssssss</p>
</div>
```

If you like to put id to the `<div>` you can do so by putting `#xyz` and if like setting classes, you may put `#yourId.your-class-a.class-b` whatever.

Code Block

You can put programming code block by doing this:

```
[c]

let x = 1000
let y = x + 3000

<html>.....</html>
<p>.....</p>

[c.]
```

! In your real xtml file you have to put the `[c]` and `[c.]` at the first character otherwise xtml won't understand it. In this guide we put them at position 2 to avoid its conversion.

Line

Puts the `-----` for at least 30 characters then you get the line (or `<hr>` tag in html).

Variables

Defines variable set in the file as:

[Go up top](#)

```
{{  
    var1 = value.....  
    var2 = value2.....  
    ...  
}}
```

And then in any where on your file, you can say like: `{{ myVarname }}` so the var value shows up there at the double brackets.

For example if you put this in your file:

```
{{ $date }}
```

you'll get:

Fri Apr 25 2025 15:30:10 GMT+0700 (Indochina Time)

Right now we have what can be called the global vars, like: `$date` and `$timeStamp` . So any where in your xml text you put `{{ $date }}` it publishes the java script date style (from the js code like `new Date`).

You can define the Var any where in the doc but all the Vars will be assembled into 1 main object, which has only 1 per file. So all your Vars must be unique because it is each property of the main var.

List of default variables in xtml so far

variable	do	example
<code>\$date</code>	get java script style date format	Fri Apr 25 2025 15:30:10 GMT+0700 (Indochina Time)
<code>\$timeStamp</code>	get js timeStamp format	1745569810714
<code>\$randomInt</code>	get random integer 5 digits	22060
<code>\$randomWords</code>	get random words 8 characters	ffdCnhDE
<code>\$mixWordNum</code>	get random mix words & num, 12 chars	JOkSYcZ3n0Vl
<code>\$isoTimeCondense</code>	get ISO date format but condensed	202504251530

For `$randomInt`, `$randomWords`, `mixWordNum` if you want different length from what it given, you can use the `[[...]]` by saying:

`[[getRandomInt(16)]]` and then you'll get like:

5588625340983441

The other function names are: `getRandomWord()`, `getMixWordNum()` , check out [xTag section](#) for more info.

Attribute

At the end of any line you can put `id` or `classes` to that line like:

this is your line..... `[#xyz.red.green]`

[Go up top](#)

And you'll get this in your converted html:

```
<p id="xyz" class="red green">.....</p>
```

can be like `[#id]` ...for only id

or like `[.class-a.class-b]` ...for classes, you can put any class as you want

The square brackets at the end of line allows you to set id and/or classes. You may set only id, or class, up to you.

Block Quote

Do this:

```
"
```

```
/**this is the quote..... something is going to happen and may not be that
htings. The USA is doing something headache to the world. What'sre you dong, guys?
Yeah!**//
```

```
Is this going to be OK? I'm not sure but yeah, it is going to happen any time soon. The
WW3? No, but can be. Who knows, man!
```

```
"
```

Output will show:

this is the quote..... something is going to happen and may not be that htings. The USA is doing something headache to the world. What'sre you dong, guys? Yeah!

Is this going to be OK? I'm not sure but yeah, it is going to happen any time soon. The WW3?
No, but can be. Who knows, man!

Again, the `#id` and `.class` are optional, if you set the class in your template will effect to this block quote style.

In the block quote you can put other formatting tags like `**..** //..// __..__` whatever.

Quote

Puts `["]the quote from someone["]` will give your `<q>your text</q>` in html format. Output is like "this is a quote" from someone.

xTag

This is the `[[...]]` as you can see we used it for `$url`, `$img` but we can do few more.

Besides to making `[[$img = ...]]` and link `[c] ...`

`[c]` we may put java script into it like:

```
[[ Date.now() ]]
[[ 5149 ]]
```



```
[[ getRandomInt(16) ]]
```

So your document can calculate or do some programming things if you know java script.

There're functions you may use from inside the xml like:

- `getRandomInt(LENGTH)`
- `getRandomWords(LENGTH)`
- `getMixWordNum(LENGTH)`
- `getIsoTimeCondense(LENGTH)`

For example if you say: `3 + 5 = [[3 + 5]]`

Then you get like:

`3 + 5 = 8`

Or you say `this is my code = [[getMixWordNum(16)]]` then you get:

`this is my code = rFAXmCGbxuJagAdW`

Or say: `this is my ISO condensed code = [[getIsoTimeCondense()]]` and you get:

`my ISO code = 202504251530`

which is the condensed style ignoring the dash '-' gooding for applying doc# all those stuff.

Or you like to do js code like: `this is my code = [[Date.now().toString().slice(-5)]]` . Then you get:

`this is my code = 10717`

Summary for this kind of `[[...]]` thing. You can do a lot in your document if you know java script. And the last note, if you just put a text inside it, then you just get a plain `[[this is plain text]]` , nothing happens.

Table

Simply do this to make table in xml:

```
[t]
```

```
# | Product      | Price      | Qty | Total
1 | Mobile Phone | 500.00     | 5   | 46532
2 | Coffee       | 5,456,745  | 2   | 46
```

```
[t.]
```

And you get table in html like this:

#	Product	Price	Qty	Total
1	Mobile Phone	500.00	5	46532
2	Coffee	5,456,745	2	46

You can also set id & classes by:

```
[t]#id.class-a.class-b
```

[Go up top](#)

```
...
[t.]
```

The beauty of applying `xml` in your table

If you put your variables into table, it's beauty. Like:

```
{{
product = coffee
price   = 325.25

}}

[t]

# | Product      | Price      | Qty | Total
1 | {{ product }} | {{ price }} | 3   | [[ 3 * {{ price }} ]]
2 | sssss        |            |     |
```

[t.]

And this is what you get in html:

#	Product	Price	Qty	Total
1	Coffee	325.25	3	975.75
2	sssss			

So beauty 'right! OK, now you can name your variables and calculate them and put them in your nice table in html with ease. So your quotation, or any doc that can utilize these things will be at breeze for you. WOW!

Align your table data cells

You can align your data cells to left, center or right. See this:

```
[t]

Product      | Price
Coffee       | 1,000.00 .r
noodle       | 500.45 .r
steak        | 80.00 .r

[t.]
```

And you get this:

Product	Price
Coffee -----	1,000.00
noodle	500.45
steak	80.00

This only works for the table cells, not applying to the table head as normally the table head will be aligned center which is nice already.

So the table cells supports classes and you can set like: `.r` , `.c` or `.l` and actually other classes that existed in the `xml` default, or the classes in your template. But `.r`, `.c` would be useful for table cells alignment especially when you put numbers into the cells.

Note: `.r` = right align, `.c` = center and the `.l` = left but left already default to all table cells any way.

Direct Html

We can put html tags directly in the xtml file so it just pass to the html file when you convert. Simply like that. So in using the xtml, it's an art of mixing between the xtml tags and html tags but it still works in the browser when you converted them.

So you can put things like:

```

<div>
.....
.....
</div>
```

And all these pass to your browser when you view html version. Yeah!

Default classes

xtml gives you some default classes which you can use but when you not taking any template (means: you didn't set any tamplet variable in the `{{...}}` block). See following:

class	do
indent, indent-16	set margin-left = 16px
sans-serif, sans	font to sans-serif
mono	set font to monospace
red	text color = red
green	text color = green
blue	text color = blue
yellow	yellow
orange	orange
brown	brown
gray	gray
pink	pink
purple	purple
indigo	indigo color, code = #5c6bc0
l	text align left
c	text aligns center
r	aligns right

So in your document you can put like `this text is green [.green]` then you get the green out of it. Like below:

This text is green
[Go up top](#)

Or you like to make an indent for a block of your content you may do like:

```
[[.indent
```

```
So now this is a div of my text and I want it indented a little so it feels different  
from other part of my story.
```

```
]]
```

Then it goes:

```
So now this is a div of my text and I want it indented a little so it feels different from other part of my  
story.
```

Easier ha! :-D

Final Note

So we encourage you to write in xtml style and then convert to html for reading. But still, if you techie guys and hating fancy stuff, you can just read it in plain xtml format, fine. :-D

Have fun writing.

I personally will use this thing to write web pages, so I can put fancy things in my template and then I just write the story in this format, then convert to a nice web page before upload to server. Make life easier.

Good luck, guys.