

<link href="faq.html#forrest">

Use this to [link](#) to another document and go to the named anchor. This will open the new document in the same browser window.

<jump href="faq.html">

Use this to jump to another document and optionally go to a named anchor within that document. This will open the new document in the same browser window. So what is the difference between link and jump? The jump behaves differently, in that it will replace any frames in the current window. This is the equivalent of `<a . . .`

`target="_top">`

<fork href="faq.html">

Use this to fork your webbrowser to another document. This will open the document in a new, unnamed browser window. This is the equivalent of `<a . . .`

`target="_blank">`

- Oh, by the way, a definition list `<dl>` was used inside the previous list item. We could put another
 - unordered list
 - inside the list item

Or even tables..	inside tables..
or inside lists, but I believe this liberty gets quickly quite hairy as you see.	

Table 1: A sample nested table

So far for the in-line elements, let's look at some paragraph-level elements.

FIXME (SN):

The `<fixme>` element is used for stuff which still needs work. Mind the author attribute!

Note:

Use the `<note>` element to draw attention to something, e.g. ...The `<code>` element is used when the author can't express himself clearly using normal sentences ;-)

Note:

Sleep deprivation can be the result of being involved in an open source project. (a.k.a. the `<warning>` element).

Apart from unordered lists, we have ordered lists too, of course.

1. Item 1
2. Item 2
3. This should be 3 if my math is still OK.

1.1. Using sections

You can use sections to put some structure in your document. For some strange historical reason, the section title is an attribute of the `<section>` element.

1.2. Sections, the sequel

Just some second section.

1.2.1. Section 2.1

Which contains a subsection (2.1).

1.3. Showing preformatted source code

Enough about these sections. Let's have a look at more interesting elements, `<source>` for instance:

```
// This example is from the book _Java in a Nutshell_ by David Flanagan.
// Written by David Flanagan. Copyright (c) 1996 O'Reilly & Associates.
// You may study, use, modify, and distribute this example for any purpose.
// This example is provided WITHOUT WARRANTY either expressed or implied.

import java.applet.*;    // Don't forget these import statements!
import java.awt.*;

public class FirstApplet extends Applet {
// This method displays the applet.
// The Graphics class is how you do all drawing in Java.
public void paint(Graphics g) {
g.drawString("Hello World", 25, 50);
}
}
```

Please take care to still use a sensible line-length within your source elements.

1.4. Using tables

And now for a table:

heading cell	heading cell
data cell	data cell

Table 1: Table caption

Not much of attributes with `<table>`, if you ask me.

1.5. Using figures

And a figure to end all of this.



project logo

2. Changes since document-v11

doc-v12 enhances doc-v11 by relaxing various restrictions that were found to be unnecessary.

- Links ((`link`|`jump`|`fork`) and inline elements (`br`|`img`|`icon`|`acronym`) are allowed inside `title`.
- Paragraphs (`p`|`source`|`note`|`warning`|`fixme`), `table` and `figure`|`anchor` are allowed inside `li`.
- Paragraphs (`p`|`source`|`note`|`warning`|`fixme`), lists (`ol`|`ul`|`dl`), `table`, `figure`|`anchor` are allowed inside definition lists (`dd`) and tables (`td` and `th`).
- Inline content (`strong`|`em`|`code`|`sub`|`sup`|`br`|`img`|`icon`|`acronym`|`link`|`jump`|`fork`) is allowed in `strong` and `em`.

© 2002 Apache Forrest