

Syntax Examples

tropf

ABSTRACT

Demonstration of possible syntax using groff, sorted by macro package.

1. Plain

Plain groff provides a small base for the language. Though mainly it governs the general look and feel of the source code.

1.1. Code Structure

Newlines have meanings. To create a line purely for source code structure, without it having a meaning for the output use **a single . char**.

```
.PP
Some text
.
.
.PP
More Text
```

1.2. Control Sequence

Generally all commands begin with a dot. This can be changed using the `.cc` macro.

```
.cc ~
. now the dot can be used like any other character
.
commands now begin with a ~
~PP
a paragraph
.
call cc without arguments to reset
~cc
```

1.3. Inline Formatting

To switch the font inline, use the sequence `\f[]` or alternatively `\f if is only a single character`. With `` being one of these:

Code	Result
R	roman (regular)
I	<i>italics</i>
B	bold
C	constant width

BI	<i>bold italics</i>
CI	<i>constant width italics</i>
CB	constant width bold
CR	constant width roman
CBI	<i>constant width bold italics</i>

The above table looks like that in the source:

```
[...]
CI | \f[CI]constant width italics
CB | \f[CB]constant width bold
CR | \f[CR]constant width roman
[...]
```

1.4. Escape Sequences

groff provides plenty of special characters via various escape sequences. A complete list can be found on the man page **groff_char(7)**.

Sequence	Result	Description
<code>\e</code>	<code>\</code>	backslash
<code>\[Fo]</code>	<code><<</code>	left guillemet
<code>\[Fc]</code>	<code>>></code>	right guillemet
<code>\[fo]</code>	<code><</code>	single left-pointing angle
<code>\[fc]</code>	<code>></code>	single right-pointing angle
<code>\[em]</code>	<code>—</code>	em-dash
<code>\[en]</code>	<code>-</code>	en-dash
<code>\[pc]</code>	<code>·</code>	period centered

2. ms

The largest and most influential macro package.

2.1. Preamble

Technically all parts of the Preamble are optional. In practice it should at least contain a title.

```
.TL
This is a title
```

Any number of authors (including none) can be specified. My convention is to put that **before** the abstract.

```
.AU
trof
.AU
another co-author
```

The abstract should only be a couple of sentences.

```
.AB
This document is on $topic.
It gives a general overview and shows an example.
.AE
```

Generally I add the `.DA` macro right after the abstract. It prints the date of document generation on every page. Combined with `make` only regenerating updated documents, this automatically prints the last

edited date for every document.

› That apparently does not work on the html export.

2.2. Document Structure

2.2.1. Headings

Headings are specified by a level, 1 being the highest. They are numbered and added to a TOC automatically. For example this section has the following source for its headings:

```
.NH 1
ms
.NH 2
Preamble
.NH 2
Document Structure
.NH 3
Headings
.NH 3
Paragraphs
.NH 2
Footnotes
```

In theory the `.SH` macro can create an unnumbered subheading. Don't do that though, the styling of that sucks.

2.2.2. Paragraphs

There are multiple paragraph types. The default that almost always should be used is `.PP`. Instead of complicated text descriptions, below will be some examples.

This paragraph uses `.PP`. Tempora a perferendis non. Officia sint quo eligendi harum. Enim et molestiae autem nemo ut soluta. Quam laboriosam fuga impedit sapiente quia consequuntur. Perspiciatis rerum voluptate nisi animi. Exercitationem corrupti asperiores similique aut recusandae fuga quis. Accusantium enim distinctio non nulla. Voluptates molestiae facere eos neque qui magnam. Dicta aspernatur aut et praesentium est aut. Inventore et officia ex et placeat odit quas.

This paragraph uses `.LP`. Tempora a perferendis non. Officia sint quo eligendi harum. Enim et molestiae autem nemo ut soluta. Quam laboriosam fuga impedit sapiente quia consequuntur. Perspiciatis rerum voluptate nisi animi. Exercitationem corrupti asperiores similique aut recusandae fuga quis. Accusantium enim distinctio non nulla. Voluptates molestiae facere eos neque qui magnam. Dicta aspernatur aut et praesentium est aut. Inventore et officia ex et placeat odit quas.

This paragraph uses `.QP`. Tempora a perferendis non. Officia sint quo eligendi harum. Enim et molestiae autem nemo ut soluta. Quam laboriosam fuga impedit sapiente quia consequuntur. Perspiciatis rerum voluptate nisi animi. Exercitationem corrupti asperiores similique aut recusandae fuga quis. Accusantium enim distinctio non nulla. Voluptates molestiae facere eos neque qui magnam. Dicta aspernatur aut et praesentium est aut. Inventore et officia ex et placeat odit quas.

This paragraph uses `.XP`. Tempora a perferendis non. Officia sint quo eligendi harum. Enim et molestiae autem nemo ut soluta. Quam laboriosam fuga impedit sapiente quia consequuntur. Perspiciatis rerum voluptate nisi animi. Exercitationem corrupti asperiores similique aut recusandae fuga quis. Accusantium enim distinctio non nulla. Voluptates molestiae facere eos neque qui magnam. Dicta aspernatur aut et praesentium est aut. Inventore et officia ex et placeat odit quas.

This paragraph uses `.IP`. Its speciality is being usable for lists. You can use it directly, though generally rather directly use the macros for lists. Tempora a perferendis non. Officia sint quo eligendi harum. Enim et molestiae autem nemo ut soluta. Quam laboriosam fuga impedit sapiente quia

consequuntur. Perspiciatis rerum voluptate nisi animi. Exercitationem corrupti asperiores similique aut recusandae fuga quis. Accusantium enim distinctio non nulla. Voluptates molestiae facere eos neque qui magnam. Dicta aspernatur aut et praesentium est aut. Inventore et officia ex et placeat odit quas.

2.3. Highlighting

Text can also be styled using macros provided. This is in general *more easy* to type, though it breaks the flow of text in the source. One huge advantage is, that it does not require resetting the font back to roman.

```
Text can also be styled using macros provided.
This is in general
.I "more easy"
to type, though it breaks the flow of text in the source.
One huge advantage is,
that it does not require
resetting the font back to roman.
```

The general way the formatting macros work in the ms package is like so:

```
.<MACRO> "text" "postfix" "prefix"
```

Using bold text as an example this would result in:

```
prefixtextpostfix
```

There is no space between the postfix, prefix and the actual text. This is mainly handy for formatting things at the **end of sentences**.

```
There is no space between the postfix,
prefix and the actual text.
This is mainly handy for formatting things at the
.B "end of sentences" "."
```

› **This behaviour is different from the man macros.**

<MACRO> can be any of these:

<MACRO>	Result
B	bold
R	roman (regular)
I	<i>italics</i>
CW	constant width
BI	<i>bold italics</i>

› Macros are less expressive than the inline styles.

2.4. Footnotes

Sometimes there is an addition to be made in a text. This can be done using a footnote¹.

¹ Well, to be honest, this is just an example. There is nothing important to say.

```
.PP
Sometimes there is an addition to be made in a text.
This can be done using a footnote\**.
.FS
Well, to be honest, this is just an example.
There is nothing important to say.
.FE
```

› Footnotes don't really work in html and look not good in ascii/utf8. Avoid them.

3. www

The www macro package adds some macros for authoring web pages.

3.1. URLs

The .URL macro can be used to create links. It is fairly straight forward, as this example link to xkcd (<https://xkcd.com>).

```
It is fairly straight forward, as
.URL "https://xkcd.com" "this example link to xkcd" "."
```

The general syntax is the following, with the text and postfix parameters being optional.

```
.URL "URI" "text" "postix"
```

3.2. Lists

Lists should generally be preceded by a paragraph of some sort.

› This seems odd. But the html device *is* odd.

3.2.1. Unordered Lists

- Exhibit A
- Exhibit B

```
.PP
.ULS
.LI
Exhibit A
.LI
Exhibit B
.ULE
```

3.2.2. Ordered Lists

- 1 Item 1
- 2 Item 2

```
.PP
.OLS
.LI
Item 1
.LI
Item 2
.OLE
```

4. tbl

Tables are handled by a preprocessor in groff named **tbl(1)**. The man page on **tbl** is a really good explanation, so here is just a template to copy and edit.

text	right bold	centered?
A	<i>italics</i>	left
BC	<i>centered</i>	looooooong text
Sum	<i>loooooooooooooong text</i>	not summed

```
.TS
nospaces center tab(|);
lFC rB c
lFC cI l.
text    | right bold    | centered?
=
A    | italics    | left
BC   | centered    | looooooong text
-
Sum  | loooooooooooooong text | not summed
.TE
```

5. Custom

Macros which are added by my custom macro package.

5.1. Code Listings

Code Listings can be introduced using two macro pairs. One also changes the control character from `.` to `~`, while the other leaves them be.

5.1.1. CB and CE

```
I can
use macros here

produced by

.CE
I can
.B "use macros here"
.CB
```

5.1.2. CBT and CET

This section also
.B disables
the . control character.
Though the ~
(tilde)
character still allows macro usage.
Some macros like .B (actually now ~B)
will produce strange text though:
.ie 1 . nr par*prev-font 6
bold
.
.el .ft B

produced by

```
.CBT
This section also
.B disables
the . control character.
Though the ~
~ft I
(tilde)
~ft
character still allows macro usage.
Some macros like .B (actually now ~B)
will produce strange text though:
~B bold
~CET
```

› Code blocks started by .CBT must be ended with ~CET, because .CBT changed the control character to ~.

5.2. Notes

Notes can be easily added using the .NOTE macro. It also accepts up to nine arguments to add a short note easily.

- › A Note looks like this
- › Alternatively the multiline style can be utilised.

```
.NOTE A Note looks like this
.NOTE
Alternatively the multiline style
can be utilised.
```

5.3. Other

- The Link color is automatically set to the document default color.
- The .LOREM macro will insert a couple of words to use for testing.

6. See also

Get used to read man pages.

- **groff(1)**: lists good documentation, pay attention to the "See Also" section of the man page.
- **groff(7)**: general groff language documentation

- **groff_ms**(7): ms macro package
- **groff_www**(7): www macro package
- **tbl**(1): excellent groff table explanation
- my custom macros `<./src/tropf.tmac>` (source)
- The source of this page `<./src/syntax.groff>` and the other pages `<./src/>` of this website.
- The Groff and Friends HOWTO `<https://www.troff.org/TheGroffFriendsHowto.pdf>`
- **fortune**(6): you need that