\bfseries (font)

Font switch to set the following text in bold face.

```
text \bfseries bold text \mdseries normal text again
```

text **bold text** normal text again

\itshape (font)

Font switch to set the following text in italics.

```
text \itshape italic text \upshape normal text again
```

text *italic text* normal text again

\scshape (font)

Font switch to set the following text in small caps.

```
text \scshape Small Capitals \upshape normal text again
```

text Small Capitals normal text again

\slshape (font)

Font switch to set the following text as slanted.

```
\sffamily text \slshape slanted text \upshape normal text again
```

text *slanted text* normal text again

\upshape (font)

Font switch to set the following text to upright.

```
\itshape italic \scshape text \upshape upright text
```

italic *TEXT* UPRIGHT TEXT

\ulcshape (font)

Font switch to undo small caps.

```
\itshape italic \scshape text \upshape upright \ulcshape text
```

italic *TEXT* UPRIGHT text

\textbf{⟨*text*⟩} (font)

Font command to set ⟨*text*⟩ in bold face.

```
text \textbf{bold text} normal text again
```

text **bold text** normal text again

\textit{⟨*text*⟩} (font)

Font command to set ⟨*text*⟩ as italic.

```
text \textit{Felis gattus} normal text again
```

text *Felis gattus* normal text again

If the font does not have italics, slanted is used.

\textsl{⟨*text*⟩} (font)

Font command to set ⟨*text*⟩ as slanted.

```
\sffamily text \textsl{Felis gattus} normal text again
```

text *Felis gattus* normal text again

If the font does not have a slanted face, italics are used.

\textulc{⟨*text*⟩} (font)

Font command to set ⟨*text*⟩ as non-small caps.

```
\scshape text \textulc{non-small caps} and SC text again
```

TEXT non-small caps AND SC TEXT AGAIN

\cdr{⟨*code*⟩} (meta)

Metacommand to print ⟨*code*⟩ and then run it.

```
The \cs{colorbox} command takes two parameters:\par
text \cdr{\colorbox{red!12!yellow!80}{text}} text
```

The `\colorbox` command takes two parameters:
text `\colorbox {red!12!yellow!80}{text}` ↦ text text

\cs{⟨*control sequence name*⟩} (meta)

Metacommand to pre-pend the escape character (\\) and print ⟨*control sequence name*⟩.

```
The command \cs{textit}\marg{text} will set \meta{text}
in italics.
```

The command `\textit{`⟨*text*⟩`}` will set ⟨*text*⟩ in italics.

`\marg{`⟨*argument*⟩`}` (meta)

Metacommand to print the mandatory ⟨*argument*⟩ of a command in braces. See `\oarg`→P. 5.

```
The command \cs{foo}\marg{arg1}\marg{arg2} will process
\meta{arg1} and \meta{arg2} \ldots
```

The command `\foo{`⟨*arg1*⟩`}{`⟨*arg2*⟩`}` will process ⟨*arg1*⟩ and ⟨*arg2*⟩ …

`\oarg[`⟨*argument*⟩`]` (meta)

Metacommand to print the optional ⟨*argument*⟩ of a command in brackets. See `\marg`.

```
\cs{newcommand}\{\cs{foo}\}\sqbrackets{2}\oarg{default}
\marg{code} will use \meta{default} as \#1 if no optional
argument \meta{oarg} is specified: \cs{foo}\oarg{oarg}
\marg{marg} \textit{versus} \cs{foo}\marg{marg}.
```

`\newcommand{\foo}[2][`⟨*default*⟩`]{`⟨*code*⟩`}` will use ⟨*default*⟩ as #1 if no optional argument ⟨*oarg*⟩ is specified: `\foo[`⟨*oarg*⟩`]{`⟨*marg*⟩`}` *versus* `\foo{`⟨*marg*⟩`}`.

`\brackets{`⟨*argument*⟩`}` (meta)

Metacommand to print its ⟨*argument*⟩ inside curly brackets ("braces"). The ⟨*argument*⟩ may be empty. See `\sqbrackets`.

```
\cs{foo}\brackets{text} \cs{foo}\brackets{} \cs{foo}\marg{}
```

`\foo{text} \foo{} \foo{`⟨⟩`}`

`\sqbrackets{`⟨*argument*⟩`}` (meta)

Metacommand to print its ⟨*argument*⟩ inside square brackets ("brackets"). The ⟨*argument*⟩ may be empty. See `\brackets`.

```
\cs{foo}\sqbrackets{text} \cs{foo}\sqbrackets{} \cs{foo}\oarg{}
```

`\foo[text] \foo[] \foo[`⟨⟩`]`

```
\begin{dispListing}                                                    (meta)
  ⟨environment content⟩
\end{dispListing}
```

Environment to print its contents as a code listing. See `dispExample`→P. 4.

```
  \begin{dispListing}
  Some text and \cs{foo}\oarg{arg1}\marg{arg2}
  \end{dispListing}
```

```
    Some text and \cs{foo}\oarg{arg1}\marg{arg2}
```

```
\begin{dispExample}                                                    (meta)
  ⟨environment content⟩
\end{dispExample}
```

Environment to print its contents as a code listing and as resulting output. See `dispListing`.

```
\begin{dispExample}
  Some text, \textit{italic} \textsc{small caps}
\end{dispExample}
```

produces:

```
  Some text, \textit{italic} \textsc{small caps}
```

Some text, *italic* SMALL CAPS

The `\textit{}` command has a matching switch-pair: an on-switch `\itshape` (italic shape) and an off-switch `\upshape` (upright shape).

Listing 1: `Example listing`

```
\begin{docCommand}[
doc no index, % no index entries for this example
doc name = oarg,
doc parameter = \oarg{argument},
doc description=meta,
] {}%name
{}%parameters
Metacommand to print the optional \meta{argument} of a command in
↪  brackets. See \refCom{marg}.
\begin{dispExample}
  \cs{newcommand}\{\cs{foo}\}\sqbrackets{2}\oarg{default}
  \marg{code} will use \meta{default} as \#1 if no optional
  argument \meta{oarg} is specified: \cs{foo}\oarg{oarg}
  \marg{marg} \textit{versus} \cs{foo}\marg{marg}.
\end{dispExample}
\end{docCommand}
```

`\oarg`[⟨*argument*⟩] (meta)

Metacommand to print the optional ⟨*argument*⟩ of a command in brackets. See `\marg`→P. 3.

```
\cs{newcommand}\{\cs{foo}\}\sqbrackets{2}\oarg{default}
\marg{code} will use \meta{default} as \#1 if no optional
argument \meta{oarg} is specified: \cs{foo}\oarg{oarg}
\marg{marg} \textit{versus} \cs{foo}\marg{marg}.
```

`\newcommand{\foo}[2]`[⟨*default*⟩] {⟨*code*⟩} will use ⟨*default*⟩ as #1 if no optional argument ⟨*oarg*⟩ is specified: `\foo`[⟨*oarg*⟩] {⟨*marg*⟩} *versus* `\foo`{⟨*marg*⟩}.