Chapter 1

A Header

theory Demo = Main:

1.1 A Section

1.1.1 A subsection

A subsubsection

Here some text with some antiquotations:

′a list, x # x, x # x = y # y, any text

\[ P ;; \forall a \text{ list}. \ P \text{ list } \Rightarrow \ P (a \ # \ \text{list}) \] \Rightarrow \ P \text{ list} \]

Keywords are printed bold, rest is just copied verbatim into the document:

lemma \( a = a \)

— not a difficult proof

— note that the double quotes do not appear in the ouptput

proof —

but we could still want to have a longer text in here and do \LaTeX\ tricks:

\begin{itemize}
  \item show \( a = a \) by force
\end{itemize}

qed

end
Chapter 2

More On Locales

locale agroup = group +
assumes com: x · y = y · x

We are now in the agroup context where assumption com: x · y = y · x is visible without any further premises.
All inherited and proved theorems of the group context are available as well:

x · y · z = x · (y · z)

1 · x = x

x⁻ · x = 1

x · 1 = x

x · x⁻ = 1

etc.

Outside the context, these theorems would look like this. (for fun we replace \(\implies\) by \(\rightarrow\) in \LaTeX).

agroup prod one inv \(\rightarrow\) prod x y = prod y x

semi prod \(\rightarrow\) prod (prod x y) z = prod x (prod y z)

group prod one inv \(\rightarrow\) prod one x = x

group prod one inv \(\rightarrow\) prod (inv x) x = one

group prod one inv \(\rightarrow\) prod x one = x

group prod one inv \(\rightarrow\) prod x (inv x) = one

Changing existing output syntax:

syntax (latex output)

Cons :: 'a ⇒ 'a list ⇒ 'a list (-/- [66,65] 65)
Now existing function definitions look different:

\[
\begin{align*}
map f \emptyset &= \emptyset \\
map f (x\cdot xs) &= f x \cdot map f xs
\end{align*}
\]

Creating new symbols and changing output syntax:

**syntax** (*latex*)

\[
\text{notEx} :: (\forall a \Rightarrow \text{bool}) \Rightarrow \text{bool} \quad \text{(binder } \exists 10)\]

**translations**

\[
\neg \exists x. P \iff \neg(\exists x. P)
\]

**lemma** (*∀ x. ¬P x*) = (*¬∃ x. P x*) by blast