`crumbs`: LaTeX Package for Navigation Crumbs[*]

Yegor Bugayenko
`yegor256@gmail.com`

2022-10-03, 0.4.1

1 Introduction

This package helps you add navigation crumbs to your document, which is most useful for presentations and slide decks:

```
\documentclass{article}
\usepackage{crumbs}
\usepackage{fancyhdr}
  \fancyhf{}
  \pagestyle{fancy}
  \fancyhead[L]{\crumbs / \subcrumbs}
\begin{document}
\section{Introduction}
Some text here.
\section{Related Works}
Some other text here.
\end{document}
```

`\crumb` You may redefine `\crumb{}` and `\subcrumb{}` commands, if you want your crumbs to look nicer, for example:

```
\renewcommand\crumb[2]{{
  \ifnum\value{section}=\value{crumbi}
    \color{orange}
  \fi
  #1
}}
```

The command will be called by `\crumbs` with two arguments. The first one will be the short name of the crumb, the second one will be the long one. The counter `crumbi` will be equal to the position of the crumb being rendered, starting from zero.

`\subcrumb` The same for `\subcrumb{}`, but the comparison should be done not between `section` and `crumbi`, but between `subsection` and `subcrumbi` counters.

---

[*] The sources are in GitHub at yegor256/crumbs

`\section` The first parameter of both `\crumb{}` and `\subcrumb{}` is the optional parameter of `\section{}` and `\subsection{}` calls respectively. The second one is the mandatory parameter of them, which is the full text of the section and sub-section. When you use `crumbs` package, you can use `\section` and `\subsection` with an optional argument, which will be used in the crumbs, e.g.:

```
\section[Details]{More Details About The Method}
\subsection[SLR]{Systematic Literature Review}
```

2 Package Options

`dry` You may turn on the “dry” mode, which will prohibit the package to renew `\section{}` and `\subsection{}` commands. You will still be able to use the crumbs mechanism through `\crumbection{}` and `\subcrumbection{}` commands, for example:

```
\documentclass{article}
\usepackage[dry]{crumbs}
\begin{document}
\crumbection[Intro]{Introduction} % Add it to \crumbs
\section{Introduction} % Print it to the page
Hello, world!
\end{document}
```

`increment` You can also instruct it to increment `\section{}` and `\subsection{}` on each call to `\crumbection{}` and `\subcrumbection{}` respectively. Just use the `increment` package option.

3 How It Works

On the first run, a new file `jobname.crumbs` is created, where all `\section` and `\subsection` commands are being logged.

On the second run, the file is loaded and two commands are being created: `\crumbs` and `\subcrumbs`. The first one is a collection of `\crumb{}` calls, while the second one is a collection of `\subcrumb{}` ones.

4 Implementation

`\crumbs` It is implemented with the help of `catchfile` package. First, we define special commands for special characters inside the `jobname.crumbs` files, as suggested here:

```
\makeatletter\newcommand\crumbs@back{\@backslashchar}\makeatother
```

`\appendtofile` Then, we define a new command `\appendtofile`:

```
\RequirePackage{catchfile}
  \newwrite\appendwrite
  \ExplSyntaxOn\cs_new:Npn\appendtofile#1#2{%
    \begingroup
    \IfFileExists{#1}%
      {\CatchFileDef{\filecontent}{#1}{\endlinechar=`^^J\catcode\endlinechar=12\relax}}% keep
      {\let\filecontent\empty}%
    \immediate\openout\appendwrite=#1\relax
    \immediate\write\appendwrite{\unexpanded\expandafter{\filecontent} \text_purify:n{#2}}%
    \immediate\closeout\appendwrite
    \endgroup
  }\ExplSyntaxOff
```

`\subcrumbs` The command `\subcrumbs` prints all sub-crumbs for the current subsection:

```
\newcommand\subcrumbs{%
  \ifnum\value{section}=0\else%
    \expandafter\ifcsname crumbs\romannumeral\the\value{section}\endcsname%
      \csname crumbs\romannumeral\the\value{section}\endcsname{}%
    \fi%
  \fi%
}
```

`\crumbection` The command `\crumbection` registers a new crumb:

```
\makeatletter\newcommand\crumbection[2][]{%
  \def\temp{#1}%
  \ifx\crumbs\empty%
    \appendtofile{\jobname.crumbs}{%
      \crumbs@back ifx\crumbs@back crumbs\crumbs@back empty%
        \crumbs@back gappto\crumbs@back crumbs\{\crumbs@back setcounter\{crumbi\}\{0\}\}%
      \crumbs@back fi%
      \crumbs@back gappto\crumbs@back crumbs\{\crumbs@back stepcounter\{crumbi\}\crumbs@back c
    }%
  \fi%
  \ifcrumbs@increment\stepcounter{section}\fi%
}\makeatother
```

`\subcrumbection` The command `\subcrumbection` registers a new subcrumb:

```
\makeatletter\newcommand\subcrumbection[2][]{%
  \def\temp{#1}%
  \ifx\crumbs\empty%
    \appendtofile{\jobname.crumbs}{%
      \crumbs@back ifcsname crumbs\romannumeral\the\value{section}\crumbs@back endcsname\crumb
        \crumbs@back expandafter\crumbs@back newcommand\crumbs@back csname crumbs\romannumeral
          \crumbs@back setcounter\{subcrumbi\}\{0\}%
        \}%
      \crumbs@back fi%
      \crumbs@back expandafter\crumbs@back gappto\crumbs@back csname crumbs\romannumeral\the\v
        \crumbs@back stepcounter\{subcrumbi\}%
        \crumbs@back subcrumb{\ifx\temp\empty \{#2\}\else \{#1\}\fi}{\{#2\}}\}%
    }%
  \fi%
  \ifcrumbs@increment\stepcounter{subsection}\fi%
}\makeatother
```

`\section` We also redefined `\section` and `\subsection` commands:

```
\makeatletter\ifcrumbs@dry%
  \message{Since "dry" mode is ON we don't renew "section" and "subsection" commands}%
\else%
  \NewCommandCopy\crumbs@oldsection\section
  \renewcommand\section[2][]{%
    \crumbection[#1]{#2}%
    \crumbs@oldsection{#2}%
  }
  \NewCommandCopy\crumbs@oldsubsection\subsection
  \renewcommand\subsection[2][]{%
    \subcrumbection[#1]{#2}%
    \crumbs@oldsubsection{#2}%
  }
\fi\makeatletter
```

`\crumb` We define default implementations of `\crumb` command:

```
\ifcsname crumb\endcsname%
  \message{The command "crumb" is already defined}%
\else%
  \newcommand\crumb[2]{#1}%
\fi%
```

`\subcrumb` We define default implementations of `\subcrumb` command:

```
\ifcsname subcrumb\endcsname%
  \message{The command "subcrumb" is already defined}%
\else%
  \newcommand\subcrumb[2]{#1}%
\fi%
```

`crumbi` We also define a few counters:

```
\newcommand*\crumbs{}
\newcounter{crumbi}
\newcounter{subcrumbi}
```

Finally, we load the file into the document:

```
\AtBeginDocument{\IfFileExists{\jobname.crumbs}{\input{\jobname.crumbs}}{}}
```

5 *

Change History

v0.2.0
    General: Initial version 2

6 *

IndexNumbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.