The keyindex package*

Richard Zach
rzach@ucalgary.ca

May 1, 2013

1 Introduction

The `keyindex` package provides functionality for producing an index without directly entering index entries into the text using the `index` command, but instead by looking up short keys and printing a predefined string in the main text and adding a corresponding index entry. The standard use case is the production of an index of name: Rather than having to write in the text, e.g., "`Einstein\index{Einstein, Albert}`" every time, write "`\keyindex{Einstein}`," which produces "Einstein" in the text and an index entry under "Einstein, Albert". Of course, the correct index entries must be collected somewhere.

2 Interactions

`keyindex` requires the `ifthen` package.

3 Usage

`\keyindex` `\keyindex{`⟨*key*⟩`}[`⟨*index option*⟩`]` prints the text associated with index key ⟨*key*⟩ (using the hook `\keyindexprint`) followed by an index entry for ⟨*key*⟩. The index entry is generated using the hook `\keyindexcommand` and the optional argument ⟨*indexoption*⟩ is added to the argument of `\keyindexcommand` after a `|`. For instance, `\keyindex[(]{Einstein}` might produce

`Einstein\index{Einstein, Albert|(}`.

`\keyindexfile` `\keyindexfile{`⟨*file*⟩`}` will load the indexkey definitions from ⟨*file*⟩`.kix`, which is assumed to contain only alist of lines of the form

`\keyindexentry {`⟨*key*⟩`}{`⟨*print text*⟩`}{`⟨*index text*⟩`}`

.

*This document corresponds to keyindex 0.2, dated 2013/05/01.

4 Implementation

4.1 Setup

```
\RequirePackage{ifthen}
```

`\keyindexfile{#1}` sets name of index entry definition file to use default is keyindex.kix

```
\newcommand*{\keyindexfile}[1]{%
  \renewcommand*{\@keyindexfile}{#1}}
\newcommand*{\@keyindexfile}{\jobname.kix}
```

`\keyindexprint{#1}` prints and index key; default: just prints argument.

```
\newcommand*{\keyindexprint}[1]{#1}
```

`\missingkeyindexformat{#1}` print index key when not defined by default, `\textbf`

```
\newcommand*{\missingkeyindexprint}[1]{%
  \textbf{#1}}
```

`\keyindexcommand{#1}` Command to add index entry; by default, `\index{#1}` but can be redefined

```
\newcommand*{\keyindexcommand}[1]{\index{#1}}
```

At the beginning of the document, we read the index key definition file

```
\AtBeginDocument{%
  \InputIfFileExists{\@keyindexfile}{%
    \PackageInfo{keyindex}{Using index key definition file
      \@keyindexfile.}}{%
    \PackageWarning{keyindex}{No index key definition file
      \@keyindexfile!}}}
```

`\keyindex[#1]{#2}` `#1` optional argument to add to index entry `#2` index key Calls `\kix@<index key>{|#1}` or `\kix@<index key>{}` if `#1` empty

```
\newcommand*{\keyindex}[2][]{\@ifundefined{kix@\detokenize{#2}}{%
    \PackageWarning{keyindex}{Index key \detokenize{#2}
      undefined}%
    \missingkeyindexprint{#2}}{%
    \ifthenelse{\equal{#1}{}}{%
      \@nameuse{kix@\detokenize{#2}}{}}{%
      \@nameuse{kix@\detokenize{#2}}{|#1}}}}

\let\name\keyindex
```

`\keyindexentry{#1}{#2}{#3}` `#1` key `#2` text to be printed `#3` txt to be indexed defines a command `\idx@<key>##1` which calls `\@keyindex#2#3##1`

```
\newcommand*{\keyindexentry}[3]{%
  \@ifundefined{kix@\detokenize{#1}}{%
    \@namedef{kix@\detokenize{#1}}##1{%
      \@keyindex{#2}{#3}{##1}}}{%
    \PackageWarning{keyindex}{Duplicate definition for keyindex key
      \detokenize{#1} ignored}}}
```

`\@keyindex#1#2#3` prints `#1` indexes `#2#3`

```
\newcommand*{\@keyindex}[3]{%
  \keyindexprint{#1}%
  \keyindexcommand{#2#3}}
```