The \texttt{problemset} package

Andrew McKenzie & Matt Menzenski
http://people.ku.edu/~a326m085

February 8, 2014

Contents

1 About problemset 1

2 Using problemset 2
  2.1 The \texttt{problemset} environment 2
  2.2 The \texttt{\entry} command 2
  2.3 Example 3

3 Formatting problem sets 3
  3.1 Global formatting via options 3
  3.2 Local formatting via \texttt{fproblemset} 4
  3.3 Declarations for local formatting 4
  3.4 Columns 4

1 About problemset

The \texttt{problemset} package is designed to typeset problem sets for linguistics. It is compatible with any of \LaTeX{}’s major document classes.

To call the package, type \texttt{\usepackage{problemset}} in the preamble of your document.

The \texttt{problemset} package calls for the \texttt{tipa} package.
2 Using problemset

Using this package mainly involves two parts

1. The problemset environment
2. The \entry command

Linguistic data sets generally look like this:\footnote{These examples are from the Kiowa language (Kiowa-Tanoan, Oklahoma, USA).}

\begin{itemize}
\item a. sá: ‘break (shatter)’
\item b. há.tsò ‘how’
\item c. sép ‘descend.SG’
\item d. pʰː ‘stop’
\end{itemize}

We can divide any problem set into three columns:

1. the problem prompts (a., b., c., etc.)
2. the data
3. the glosses

2.1 The problemset environment

\begin{verbatim}
\begin{problemset}
\end{problemset}
\end{verbatim}

The data and glosses …

\begin{verbatim}
\end{problemset}
\end{verbatim}

2.2 The \entry command

Inside the problemset environment, use the \entry command to typeset the example.\footnote{The package also has the \entr command, which is identical to \entry.}

\begin{verbatim}
\entry{data}{gloss}
\end{verbatim}

Note: You do not need to format the data or the gloss. This is done automatically. Glosses are given scare quotes by the problemset package. If you want to change the format, see section 3.
2.3 Example

This set is from Swahili (Bantu, Eastern Africa)
\begin{problemset}
\entry{atanipenda}{he will like me} a. atanipenda ‘he will like me’
\entry{utanipenda}{you will like me} b. utanipenda ‘you will like me’
\entry{alinipiga}{he beat me} c. alinipiga ‘he beat me’
\end{problemset}

3 Formatting problem sets

3.1 Global formatting via options

Formatting can be done at a global level with the following package options, which can be used together if \LaTeX generally allows.

<table>
<thead>
<tr>
<th>[option]</th>
<th>format effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>bare</td>
<td>removes problem prompts</td>
</tr>
<tr>
<td>heavy</td>
<td>sets problem prompts in boldface</td>
</tr>
<tr>
<td>roman</td>
<td>uses Roman numerals for prompts</td>
</tr>
<tr>
<td>arabic</td>
<td>uses Arabic numerals for prompts</td>
</tr>
<tr>
<td>IPA</td>
<td>sets data in IPA (requires \texttt{tipa})</td>
</tr>
<tr>
<td>bold</td>
<td>sets data in boldface</td>
</tr>
<tr>
<td>italic</td>
<td>sets data in italics</td>
</tr>
<tr>
<td>slant</td>
<td>sets data in slanted</td>
</tr>
<tr>
<td>noquote</td>
<td>removes scare quotes</td>
</tr>
<tr>
<td>italicgloss</td>
<td>puts glosses in italics</td>
</tr>
<tr>
<td>boldgloss</td>
<td>puts glosses in boldface</td>
</tr>
</tbody>
</table>

For example: \texttt{\usepackage[IPA,bold,bare]{problemset}}
3.2 Local formatting via \texttt{fproblemset}

To format locally, use the \texttt{fproblemset} (formatted problemset) environment. It takes an argument, in which you can put any of the declarations in 3.3. The available capabilities of the local options are the same as those of the global formatting options.

\begin{fproblemset}\italicdata
\entry{\textit{voiture}}{car} \quad \text{a. voire} \quad \text{‘car’}
\entry{\textit{chaise}}{chair} \quad \text{b. \textit{chaise}} \quad \text{‘chair’}
\end{fproblemset}

3.3 Declarations for local formatting

<table>
<thead>
<tr>
<th>declaration</th>
<th>format effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>\bareprompts</td>
<td>removes problem prompts</td>
</tr>
<tr>
<td>\heavyprompts</td>
<td>sets prompts in boldface</td>
</tr>
<tr>
<td>\romanprompts</td>
<td>sets prompts in Roman numerals</td>
</tr>
<tr>
<td>\arabicprompts</td>
<td>sets prompts in Arabic numerals</td>
</tr>
<tr>
<td>\alphprompts</td>
<td>sets prompts in a-z</td>
</tr>
<tr>
<td>\IPAdata</td>
<td>sets data in IPA (requires \texttt{tipa})</td>
</tr>
<tr>
<td>\bolddata</td>
<td>sets data in boldface</td>
</tr>
<tr>
<td>\italicdata</td>
<td>sets data in italics</td>
</tr>
<tr>
<td>\slantdata</td>
<td>sets data in slanted</td>
</tr>
<tr>
<td>\noquote</td>
<td>removes scare quotes</td>
</tr>
<tr>
<td>\italicgloss</td>
<td>puts glosses in italics</td>
</tr>
<tr>
<td>\boldgloss</td>
<td>puts glosses in boldface</td>
</tr>
</tbody>
</table>

Note: The declarations do not have any effect if you place them inside a \texttt{problemset} environment.

Overriding global options: Placing these declarations in your document formats the problemsets from that point onward, overriding the global options.

Formatting individual examples: There is no way provided by the package to format individual examples. It can be done by putting the ordinary \LaTeX formatting declarations or commands inside the arguments of \texttt{entry}. The specially formatted text must be delimited by curly braces.

Example: \texttt{\entry\{\bfseries atanipenda\}\{he will like me\}}

3.4 Columns

Many problem sets are split over columns. The \texttt{problemset} package does not provide a way to split problem sets into columns, but it is compatible with \texttt{multicol} package.