Arno Trautmann
arno.trautmann@gmx.de

lernkarten

Abstract

This is the documentation of the class `lernkarten`. I am writing this class to have an easy way to typeset my cards for learning for exams, especially diploma-exams. As I always loose and forget things, I wanted to have all important stuff on my computer so I will not forget it. – Just print and learn ☺

If you have any suggestions or comments, just drop me a mail, I'll be happy to get any response!

Contents

Usage 1
Language 1
Class Options 1
Typesetting Cards 2
Structurising 2
Open Questions 2
Implementation 2
Known Bugs 7
To Dos 7

Usage

Language

So far, the whole user interface is english. Support for german is being implemented. Use `german` as class-option to active it.

Class Options

The usage of this cass is quite easy. Just load it with

```
\documentclass{lernkarten}
```

Class options are: (english/german, if available)

number/anzahl The number of cards per page. This directly affects the size of each card. You must take care that none of your cards does exceed the given size! Else there will be huge damage and the sun will explode!
A reasonable number, and therefore preset, is 10.

noheader If you use sections and/or parts to structure your cards, they will be in the header. `noheader` will turn this off.

sectionsoncards Prints the section on every card. So you always know what theme this cards belongs to. Default is off. Take care that it is *sections**on**cards*, but not *sections-cards*.

enumerate/nummerieren By default, all cards are enumerated. `enumerate=false` turns enumeration off. With `enumerate=section`, the cards will be enumerated section-wise.

Typesetting Cards

There are approximately three ways to typeset a card: Write

```
\card[ion optics]{How many electrodes are needed for a single lens}{3}
```

for a short question–answer-pair. For longer sentences or formulae, write:

```
\question
question text …
\answer
3

```

␣

It is extremely important that you leave an *empty line* after the answer, as this empty line is part of the command!

For the lazy guys:

```
\shortquestion
questiontext

answertext
```

Again, the empty lines *must* be given! Else your document will most probably not compile! The `questiontext` or `answertext` can be anything that has no empty lines. For such, use `\card` or `\question`.

Structurising

You can structure your document using the normal commands `\section` and `\part`. Depending on class options, the regarding part and section will be printed on the top of the pages and/or on the cards itself.

Open Questions

Whenever anything is not clear when writing the cards, you can put an `\openquestion{}` at that position. At the end of the document, there will be a table of open questions (`tooq`) where the arguments of `\openquestion{}` are listet whith pagenumbers and links to the cards. `\oq{}` and german `\unklar{}` are aliases for `\openquestion{}`.

If there is an open question, a small ☹ (`\frownie`) will be set. If you want to change that symbol or want none at all, say `\openquestionmark{your␣mark}`.

Implementation

We begin with loading of options and setting of constants that are needed even before defining the keys:

```
\cards@per@page          1 \def\cards@per@page{10}
\cards@enumerate@value   2 \def\cards@enumerate@value{}
\ifcards@header          3 \newif\ifcards@header
```

```
\ifcards@sections
\ifcards@sectioncards
\ifcards@enumerate
\ifcards@enumerate@section
```

```
4  \newif\ifcards@sections
5  \newif\ifcards@sectioncards
6  \newif\ifcards@enumerate
7  \newif\ifcards@enumerate@section
8  \cards@headertrue
9  \cards@sectioncardsfalse
10 \cards@enumeratetrue
11 \cards@enumerate@sectionfalse
```

```
anzahl
\cards@per@page
number
\cards@per@page
noheader
nummerieren
\cards@enumerate@value
enumerate
\cards@enumerate@value
sectionsoncards
german
\cards@language
language
\cards@language
```

```
12 \RequirePackage{xkeyval}
13 \DeclareOptionX{anzahl}{\def\cards@per@page{#1}}
14 \DeclareOptionX{number}{\def\cards@per@page{#1}}
15 \DeclareOptionX{noheader}{\cards@headerfalse}
16 \DeclareOptionX{nummerieren}{\def\cards@enumerate@value{#1}}
17 \DeclareOptionX{enumerate}{\def\cards@enumerate@value{#1}}
18 \DeclareOptionX{sectionsoncards}{\cards@sectioncardstrue}
19 \DeclareOptionX{german}{\def\cards@language{german}}
20 \DeclareOptionX{language}{\def\cards@language{#1}}
21 \ProcessOptionsX
```

Now loading of the class and usefull packages. `amsmath` is surely needed for any mathematic typesetting, `boxedminipage` is needed for the cards, etc. Then setup of the hyperlinks and page layout.

```
22 \LoadClass[fleqn]{scrartcl}
23 \RequirePackage{
24   amsmath,
25   boxedminipage,
26   calc,
27   geometry,
28   hyperref,
29   ifthen,
30   polyglossia,
31   scrpage2,
32   xltxtra
33 }
34 \hypersetup{%
35   pdfborder=false,%
36   colorlinks=true,%
37   linkcolor=blue%
38 }
39 \geometry{
40   bindingoffset=0cm,
41   margin=1cm,
42   headsep=0.2cm
43 }
44 \setlength{\parindent}{0em}
```

Language specific settings. Only german and english are available so far! (So a simple `\ifthenelse` does the job.)

```
45 \setmainlanguage{\cards@language}
46 \ifthenelse{\equal{\languagename}{german}}{
```

```
\cards@answertext
\cards@openquestiontext
    \def\cards@answertext{Antwort}
    \def\cards@openquestiontext{Liste␣offener␣Fragen}
}{
\cards@answertext
\cards@openquestiontext
    \def\cards@answertext{answer}
    \def\cards@openquestiontext{List␣of␣Open␣Questions}
}
```

Now, the pagelayout. `scrheadings` is used and `part:section` is written on top of every site. If option `noheader` is given, there will be no header. This might be useful for printing, while headers are useful for view on screen.

```
\cards@part
\cards@sect
\def\cards@part{}
\def\cards@sect{}
\pagestyle{scrheadings}
\setkomafont{pagehead}{\normalfont\bfseries}
\cfoot{}
\ifcards@header
  \chead{\cards@part\cards@sect}
\else
\fi
```

The definition of the sections is a bit tricky. For the table of contents, we need a valid `\section*` command, so we save it by the `\@ifstar` construct. The rest is for setting the sections as wishey by the user: resetting the counters, writing the section on the cards and/or in the headings. Definition of `\part` is way more straight-forward. `\subsection` and `\subsubsection` have no effect at the moment, but can be used.

```
\let\oldsection\section
\renewcommand\section{\@ifstar{\oldsection*}{\cards@section}}
\cards@section
\newcommand\cards@section[1]{
  \ifnum␣\thequestion@page␣>␣1␣␣% needed here for conistent page numbering
    \set@answers
  \fi
\cards@sect
  \def\cards@sect{#1}
  \ifcards@enumerate@section
    \setcounter{total@question}{0}
    \setcounter{total@answer}{0}
  \fi
  \refstepcounter{section}%
  \addcontentsline{toc}{section}{%
      \protect\numberline{\thesection}#1}%
}
\renewcommand\part[1]{
  \ifnum␣\thequestion@page␣>␣1
    \set@answers
  \fi
\cards@sect
  \def\cards@sect{#1}
\cards@part
  \def\cards@part{#1:~}
  \refstepcounter{part}%
  \addcontentsline{toc}{part}{%
  \protect\numberline{#1}}%
}
```

```
\renewcommand\subsection[1]{}
\renewcommand\subsubsection[1]{}
```

Need `\TeXXeTstate=1` for the right-to-left typesetting of the answers

```
\TeXXeTstate=1
```

For evalutation of the enumeration we need some booleans. First one: enumeration at all, second one: section-wise enumeration.

```
\ifthenelse{\equal{\cards@enumerate@value}{false}}{%
      \cards@enumeratefalse}{}
\ifthenelse{\equal{\cards@enumerate@value}{section}}{%
      \cards@enumerate@sectiontrue}{}
```

The calculation of the height of the cards is straight-forward: Textheight divided by the number of cards. Unfortunately, this does *not* work as expected! There is a correction factor needed – even at user level :(See page 7.

ds@height@correct@factor
```
\def\cards@height@correct@factor{\real{1.5}}␣␣% FIXME why is this factor
      needed to fill the page?
```

\setcorrectionfactor ds@height@correct@factor
```
\def\setcorrectionfactor#1{\def\cards@height@correct@factor{\real{%
      #1}}}
```

\height@of@boxes
```
\def\height@of@boxes{\textheight/\cards@per@page\expandafter*%
      \cards@height@correct@factor}
```

\lernkarte
```
\newcommand\lernkarte[1]{%
  \boxedminipage{.5\textwidth}\textbf{#1}\\[2ex]%
  \minipage[t]{\textwidth}%
}
```

\endlernkarte
```
\def\endlernkarte{
  \endminipage\vphantom{\rule[-\height@of@boxes]{0pt}{%
       \height@of@boxes}}
  \endboxedminipage\kern-1em
}
```

Now set some counters. They will be used to number the cards. So you might sort your cards and you can see how far you come on one day … or you can exchange with your partners (”Having trouble on question 1357 – can you explain that equation?“)

total@question
```
\newcounter{total@question}␣␣␣␣␣␣␣␣␣␣␣% total number of questions
\setcounter{total@question}{0}
```

total@answer
```
\newcounter{total@answer}␣␣␣␣␣␣␣␣␣␣␣% total number of answers
\setcounter{total@answer}{0}
```

question@page
```
\newcounter{question@page}␣␣␣␣␣␣% the n-th question on one page
\setcounter{question@page}{1}
```

answer@page
```
\newcounter{answer@page}␣␣␣␣␣␣␣␣␣␣␣␣␣% the n-th answer@page on one page
```

\card
```
\def\card#1#2#3{␣% Titlequestionanswer typesetting the question: use the environ-
      ment lernkarte for the layout
  \begin{lernkarte}{%
    \stepcounter{total@question}%
    \ifcards@sectioncards
      \cards@sect~%
    \fi
    \ifcards@enumerate%
```

```
      \thetotal@question:␣%
    \fi%
    #1%
  }
  #2
  \end{lernkarte}
  \expandafter\def\csname␣answer@\thequestion@page␣\endcsname{#3}
  \stepcounter{question@page}
```

typeset the answers when the page is full of questions.

```
  \ifnum\thequestion@page␣>␣\cards@per@page
    \set@answers
  \fi\unskip
}
```

Now we try a nicer way to typeset the answers:

\set@answers
```
\def\set@answers{
  \clearpage
  \setcounter{answer@page}{1}
```

and typeset them L to R, so every answer is on the back of the correct question

```
  \beginR\unskip
```

we loop through all saved answers

```
  \whiledo{\theanswer@page␣<␣\thequestion@page}{%
    \begin{lernkarte}{\cards@answertext:␣\stepcounter{total@answer}%
          \thetotal@answer}
      \csname␣answer@\theanswer@page\endcsname
    \end{lernkarte}
    \stepcounter{answer@page}
  }\kern2em␣% FIXME why is this kern needed here?
  \newpage%
  \setcounter{question@page}{1}%
  \setcounter{answer@page}{0}%
}
```

Now the definition and setting of the *table of open questions* (tooq):

```
\addtotoclist[tooq]␣{tooq}
```
\listoftooqname
```
\newcommand{\listoftooqname}{\cards@openquestiontext}
```
openquestion
```
\newcounter{openquestion}
```
\frownie
```
\def\frownie{\color{blue}\raisebox{2ex}{\bfseries\rotatebox{-90}{%
      \small␣:-(}}}
```
\cards@mark@oq
```
\def\cards@mark@oq{\frownie}
```
\openquestion
```
\def\openquestion#1{
  \refstepcounter{openquestion}%
```
\l@tooq
```
  \newcommand*{\l@tooq}{\l@figure}%
  \addcontentsline{tooq}{figure}{%
  \protect\numberline{\textbf{\theopenquestion}}\textbf{#1}}%
  \cards@mark@oq
}
```

typeset the missing answers and the table of open questions.

```
\AtEndDocument{
  \part{\cards@openquestiontext}
  \listoftoc{tooq}
}
\author{no␣author␣given}
\title{no␣title␣given}
\date{}
\AtBeginDocument{
  \maketitle
  \newpage
  \tableofcontents
  \newpage
}
```

And finally the user-interface. German names and shorthands for above `\defd` functions:

```
\karte
\question
\frage
\shortquestion
\openquestionmark
\cards@mark@oq
```

```
\newcommand\karte[3][]{\card{#1}{#2}{#3}}
\long\def\question#1\answer#2\par{\karte{#1}{#2}}
\long\def\frage#1\antwort#2\par{\karte{#1}{#2}}
\def\shortquestion#1\par#2\par{\frage␣#1␣\antwort␣#2␣\par}
\let\unklar\openquestion
\let\oq\openquestion
\def\openquestionmark#1{\def\cards@mark@oq{#1}}
```

⟨/class⟩

Known Bugs

There are several bugs, some only regarding the code, some heavily destroying the layout:

references The references are broken! This is surely *not* section 626!

display formula Under certain circumstances (which are not clear to me), there will be a problem if you write only one display formula as answer. This can enlarge the regarding card, which destroys the whole layout Just *add* an empty line or whatever – this will in fact *decrease* the size of the answer-card to the correct size.

tabular The same things for tabulars. But so far I have no idea how to fix this.

size of cards With the default implementation, 10 cards fill one page. If you use another number of cards, you maybe have to correct the size, as the calculation of the height does something strange. Use the command `\setcorrectionfactor{1.3}` to get the height that fits. Default is `\setcorrectionfactor{1.5}` for 10 cards.

To Dos

Some things that should be implemented but aren't so far or are very poor at the moment:
`\section` should have an optional argument for short titles.

□