

## Writing a LaTeX Class File to Produce a Form

★★★★★ (22 votes, average: 4.73 out of 5)

LaTeX - Document Classes

WRITTEN BY NICOLA TALBOT

FRIDAY, 20 NOVEMBER 2009 11:55



0



Share

Share

This article describes how to create a class file to generate a form. There are three types of forms: a form to be printed and filled in with a pen, a form filled in on the computer and then printed or sent as a PDF file, and a form to be filled in and sent electronically via a CGI (or similar) script. This article is about the second type. The third type is beyond the scope of this article and the first type can be viewed as a subset of the second.

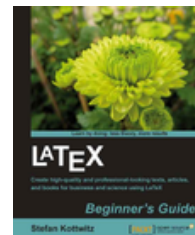
### A Simple Form

Suppose I need to create a simple one page form that must meet the following requirements:

- ▶ Use A4 paper and 12pt normal font size.
- ▶ Use a sans serif font for the pre-defined form text and a serif font for the user supplied text.
- ▶ The form should have the title "A Simple Form" centred in a large bold font.
- ▶ The form should have single-lined spaces for the user's surname and forenames.
- ▶ The form should have multi-lined spaces for the user's address and comments that can expand or contract to fit. The area for comments should be framed. Both boxes should be 1in in height when empty so that there is room in case the user wants to fill the form in by hand.
- ▶ The form should have "male" and "female" checkboxes for users to indicate their gender.
- ▶ The form should have "<18", "18-30", "31-40", "41-60" and ">60" checkboxes for users to indicate their age range.
- ▶ The form shouldn't have a header or footer (it's a single page form so doesn't require a page number).

The blank (unfilled in) form should look like:

#### Featured Book



#### Supporters and Partners

Sponsored by DANTE e.V.: The German speaking TeX Users Group

TEXNIQUE.fr

TeX <sub>W</sub><sub>LT</sub> Fragen & Antworten

TeXblog

TeXample.net  
& the TikZ Gallery

### A Simple Form

Forenames \_\_\_\_\_  
Surname \_\_\_\_\_  
Gender    Male ☐    Female ☐  
Age range:    <18 ☐    18-30 ☐    31-40 ☐    41-60 ☐    >60 ☐  
Address:

Comments:

The aim is to create a class file called, say, "simpleform" that provides commands to supply the relevant information and display the form. A sample document might look like:

```
\documentclass{simpleform}

\begin{document}
\surname{Talbot}
\forenames{Nicola Louise Cecilia}
\gender{female}
\agerange{31-40}
\address{School of Computing Sciences\\
University of East Anglia\\
Norwich. Norfolk. NR4 7TJ}
\comments{I like using \LaTeX!}
\makeform
\end{document}
```

This is similar to the way title pages work where you supply the information using commands like `\title` and display the information using `\maketitle`. However in this example, `\makeform` should display a blank form that can be filled in by hand if the information is missing.

## Identifying the Class File and Loading Required Packages and Underlying Class

Since the class is going to be called "simpleform", I need to create a file called "simpleform.cls". The first line of this file must indicate that we are using LaTeX2e:

```
\NeedsTeXFormat{LaTeX2e}
```

In the next line I need to identify this class file and specify the version information:

```
\ProvidesClass{simpleform}[2009/11/20 v1.0 A Simple Form]
```

This class is going to be based on the "article" class but I need to specify the normal font size and paper size according to the above requirements:

```
\LoadClass[12pt,a4paper]{article}
```

I'm also going to need the [wasysym](#) package because I want to use the commands `\Square` and `\XBox` to display unchecked and checked boxes:

```
\RequirePackage{wasysym}
```

 Note the use of `\RequirePackage` rather than `\usepackage` as this is in a class file not in a document.

The form should have no header or footer:


```
\pagestyle{empty}
```

I'm going to define commands that can be used to switch the font for the user supplied information and the predefined text:

```
\newcommand*{\@fillinfont}{\rmfamily}
\newcommand*{\@formfont}{\sffamily}
```

and I'm also going to define commands that display an unchecked and a checked box:

```
\newcommand*{\@unchecked}{\Square}
\newcommand*{\@checked}{\XBox}
```

 Note that I could just dispense with `\@fillinfont`, `\@formfont`, `\@unchecked` and `\@checked` by directly using `\rmfamily`, `\sffamily`, `\Square` and `\XBox`, but by defining the new commands I have made the class easier to maintain. If, for example, I later want to change the symbols or fonts used, I only need to change those four lines.

## Defining User Commands

The commands like `\surname` and `\forenames` work in much the same way as commands like `\title`. This is done by defining an *internal* command that stores the information and a *user* command that sets the information. The internal command is initially defined to do nothing to ensure that the form can be produced if no information is set.

```
\newcommand*{\@forenames}{}
\newcommand*{\forenames}[1]{\renewcommand*{\@forenames}{#1}}

\newcommand*{\@surname}{}
\newcommand*{\surname}[1]{\renewcommand*{\@surname}{#1}}
```

 The starred version of `\newcommand` and `\renewcommand` has been used because the arguments shouldn't contain paragraph breaks. This helps TeX to pick up missing braces at that point in the document.

The address and comments section are initially going to be set to a zero-width vertical rule of height 1in to ensure the required default space when the form hasn't been filled in. This will be overridden when the information is set which allows the areas to expand or shrink to fit the text.

```
\newcommand*{\@address}{\rule{0pt}{1in}}
\newcommand*{\address}[1]{\renewcommand*{\@address}{#1}}


\newcommand*{\@comments}{\rule{0pt}{1in}}
\newcommand*{\comments}[1]{\renewcommand*{\@comments}{#1}}
```

Next comes the checked boxes. Let's first deal with the gender boxes. For this, I want to have two internal commands, `\gender@male` and `\gender@female`, that default to an unchecked box. I also want a user command, `\gender` that takes one argument that can either be "male" or "female". If "male", it must redefine `\gender@male` to be a checked box, and, if "female", it must redefine `\gender@female` to be a checked box. If the argument is anything else an error message needs to be generated. Also, in case the user is tempted to try doing

```
\gender{male}\gender{female}
```

The commands should override each other so that no more than one box is ever checked. This can be done by resetting `\gender@male` and `\gender@female` to `\@unchecked` before updating the required command.

```
\newcommand*{\reset@gender}{%
  \def\gender@male{\@unchecked}%
  \def\gender@female{\@unchecked}%
}
```

 Note that I have used TeX's `\def` rather than LaTeX's `\newcommand` or `\renewcommand` because I don't want to worry about whether or not the command has already been defined.


Once the `\reset@gender` command has been defined, I need to implement it in case a blank form is required:

```
\reset@gender
```

I now need to define the user command `\gender`:

```
\newcommand*{\gender}[1]{%
  \@ifundefined{gender@#1}%
  {%
    \ClassError{simpleform}{Unknown gender `#1'}%
    {Gender can either be `male' or `female'}%
  }%
  {%
    \reset@gender
    \@namedef{gender@#1}{\@checked}%
  }%
}
```

Note that I haven't needed to use any nested `\ifthenelse` commands to check the argument.

 I've used `\@namedef` to define the command as the command name needs to be constructed from the argument supplied to `\gender`. (In the code below, I've also used `\@namedef`, in this case because the command name consists of non-alphabetical characters.) The percent signs at the end of each line discard the end-of-line character to prevent unwanted spaces appearing in the document.

I could use a similar technique for the age ranges and define:

```
\newcommand*{\reset@ageranges}{%
  \@namedef{agerange@<18}{\@unchecked}%
  \@namedef{agerange@18-30}{\@unchecked}%
  \@namedef{agerange@31-40}{\@unchecked}%
  \@namedef{agerange@41-60}{\@unchecked}%
  \@namedef{agerange@>60}{\@unchecked}%
}
```

However, this is getting a bit tiring. In a form I once had to create, there was a section that had over 50 check boxes. Using the above method is time consuming and liable to errors. Instead, I'm going to define a list of available options and the text that should go next to the relevant box. This is going to be done by defining a comma-separated list where each element contains the `{label}{text}` pair:

```
\newcommand*{\@validageranges}{%
  {<18}{\textless18},%
  {18-30}{18--30},%
  {31-40}{31--40},%
  {41-60}{41--60},%
  {>60}{\textgreater60}%
}
```

I also need a way to extract the information when I iterate through the list:

```
\newcommand*{\get@agerange}[2]{%
  \def\@agerange@label{#1}%
  \def\@agerange@text{#2}%
}
```

This will store the label in `\@agerange@label` and the text in `\@agerange@text`. I can now define `\reset@ageranges` as follows:

```

\newcommand*\reset@ageranges{%
  \@for\@agerange:=\@validageranges\do{%
    \expandafter\get@agerange\@agerange
    \@namedef{agerange@\@agerange@label}{\@unchecked}%
  }%
}

```

This iterates through each element in \@validageranges and sets the current element to \@agerange. Remember that this is in the form `{label}{text}` so \@expandafter\get@agerange\@agerange is needed. (This expands \@agerange to `{label}{text}` before applying \@get@agerange. This ensures that \@get@agerange picks up its arguments correctly.)

Now that I've defined \@reset@ageranges, I need to use it to ensure the form is correctly initialised:

```

\reset@ageranges

```

The command \@agerange is defined in a similar manner to \@gender:

```

\newcommand*\@agerange{[1]{%
  \@ifundefined{agerange@#1}%
  {%
    \ClassError{simpleform}{Unknown age range `#1'}{ }%
  }%
  {%
    \reset@ageranges
    \@namedef{agerange@#1}{\@checked}%
  }%
}

```

Note that in this case I haven't bothered to supply a helpful message listing possible options if an error is produced. If I want to be really helpful I can change the above to:

```

\newcommand*\@agerange{[1]{%
  \@ifundefined{agerange@#1}%
  {%
    \def\@ageoptions{%
      \def\@ageoptionssep{%
        \@for\@agerange:=\@validageranges\do{%
          \expandafter\get@agerange\@agerange
          \edef\@ageoptions{\@ageoptions\@ageoptionssep\@agerange@label}%
        }%
        \def\@ageoptionssep{, }%
      }%
      \ClassError{simpleform}{Unknown age range `#1'}%
      {Available options: \@ageoptions}%
    }%
  }%
  {%
    \reset@ageranges
    \@namedef{agerange@#1}{\@checked}%
  }%
}

```

Alternatively, I could be less helpful by directing the user to the manual.

## Displaying the Form

Once the user has specified the required information, the form needs to be displayed using \@makeform, but before I define it, I first need to define some new lengths that I'll need:

```

\newlength\boxwidth
\newlength\underlinewidth

```

Now I can define \@makeform:

```

\newcommand{\makeform}{%
  \begin{center}\large\bfseries
  A Simple Form
  \end{center}
  \noindent
  {\renewcommand{\arraystretch}{1.5}%
   \settowidth{\underlinewidth}{Forenames}%
   \addtolength{\underlinewidth}{-\linewidth}%

```

```

\addtolength{\underlinewidth}{2\abcolsep}%
\setlength{\underlinewidth}{-\underlinewidth}%
\begin{tabular}{@{}ll@{}}
Forenames & \@fillinfont
\quad \underline{\makebox[\underlinewidth][l]{\@forenames}}\\
Surname & \@fillinfont
\quad \underline{\makebox[\underlinewidth][l]{\@surname}}\\
Gender & Male\ \gender@male\qquad Female\ \gender@female\\
\end{tabular}%
}%
\par
\vspace{\baselineskip}%
\noindent
Age range:\qquad \@for\@agerange:=\@validageranges\do{%
\expandafter\get@agerange\@agerange
\@agerange@textt\ \@nameuse{agerange@\@agerange@label}\qquad
}%
\par
\vspace{\baselineskip}%
\noindent
Address:\par
\vspace{\baselineskip}%
\noindent
{\@fillinfont
\begin{tabular}{@{}l@{}}
\@address
\end{tabular}%
}%
\par
\vspace{\baselineskip}%
\noindent
Comments:\par
\vspace{\baselineskip}%
\noindent
\setlength{\boxwidth}{\textwidth}%
\addtolength{\boxwidth}{-2\fbboxsep}%
\addtolength{\boxwidth}{-2\fbboxrule}%
\fbbox{\parbox{\boxwidth}{\@fillinfont\@comments}}
}

```

Finally, at the end of the class file I need the line:

```
\endinput
```

## The Complete Class File

The complete code for "simpleform.cls" is given below:

```

\NeedsTeXFormat{LaTeX2e}
\ProvidesClass{simpleform}[2009/11/20 v1.0 A Simple Form]

\LoadClass[12pt,a4paper]{article}

\RequirePackage{wasysym}

\pagestyle{empty}

\newcommand*{\@fillinfont}{\rmfamily}
\newcommand*{\@formfont}{\sfamily}
\newcommand*{\@unchecked}{\Square}
\newcommand*{\@checked}{\XBox}

\newcommand*{\@forenames}{}
\newcommand*{\@forenames}[1]{\renewcommand*{\@forenames}{#1}}

\newcommand*{\@surname}{}
\newcommand*{\@surname}[1]{\renewcommand*{\@surname}{#1}}

\newcommand*{\@address}{\rule{0pt}{1in}}
\newcommand*{\@address}[1]{\renewcommand*{\@address}{#1}}

\newcommand*{\@comments}{\rule{0pt}{1in}}
\newcommand*{\@comments}[1]{\renewcommand*{\@comments}{#1}}

```

```

\newcommand*{\reset@gender}{%
  \def\gender@male{\@unchecked}%
  \def\gender@female{\@unchecked}%
}

\reset@gender

\newcommand*{\gender}[1]{%
  \ifundefined{gender@#1}%
  {%
    \ClassError{simpleform}{Unknown gender `#1'}%
    {Gender can either be `male' or `female'}%
  }%
  {%
    \reset@gender
    \@namedef{gender@#1}{\@checked}%
  }%
}

\newcommand*{\@validageranges}{%
  {<18}{\textless18},%
  {18-30}{18--30},%
  {31-40}{31--40},%
  {41-60}{41--60},%
  {>60}{\textgreater60}%
}

\newcommand*{\get@agerange}[2]{%
  \def\@agerange@label{#1}%
  \def\@agerange@textt{#2}%
}

\newcommand*{\reset@ageranges}{%
  \@for\@agerange:=\@validageranges\do{%
    \expandafter\get@agerange\@agerange
    \@namedef{agerange@\@agerange@label}{\@unchecked}%
  }%
}

\reset@ageranges

\newcommand*{\agerange}[1]{%
  \ifundefined{agerange@#1}%
  {%
    \def\@ageoptions{%
      \def\@ageoptionssep{}%
      \@for\@agerange:=\@validageranges\do{%
        \expandafter\get@agerange\@agerange
        \edef\@ageoptions{\@ageoptions\@ageoptionssep\@agerange@label}%
      }%
      \ClassError{simpleform}{Unknown age range `#1'}%
      {Available options: \@ageoptions}%
    }%
    {%
      \reset@ageranges
      \@namedef{agerange@#1}{\@checked}%
    }%
  }
}

\newlength\boxwidth
\newlength\underlinewidth

\newcommand{\makeform}{%
  \@formfont
  \begin{center}\large\bfseries
  A Simple Form
  \end{center}
  \noindent
  {\renewcommand{\arraystretch}{1.5}%
   \settowidth{\underlinewidth}{Forenames}%
   \addtolength{\underlinewidth}{-\linewidth}%
   \addtolength{\underlinewidth}{2\tabcolsep}%
   \setlength{\underlinewidth}{-\underlinewidth}%
   \begin{tabular}{@{}ll@{}}
   Forenames & \@fillinfont
   \underline{\makebox[\underlinewidth][l]{\@forenames}}\backslash

```

```

Surname & \@fillinfont
\underline{\makebox[\underlinewidth][l]{\@surname}}\
Gender & Male\ \gender@male\qquad Female \gender@female\
\end{tabular}%
}%
\par
\vspace{\baselineskip}%
\noindent
Age range:\qquad \@for\@agerange:=\@validageranges\do{%
\expandafter\get@agerange\@agerange
\@agerange@textt\ \@nameuse{agerange@\@agerange@label}\qquad
}%
\par
\vspace{\baselineskip}%
\noindent
Address:\par
\vspace{\baselineskip}%
\noindent
{\@fillinfont
\begin{tabular}{@{}l@{}}
\@address
\end{tabular}%
}%
\par
\vspace{\baselineskip}%
\noindent
Comments:\par
\vspace{\baselineskip}%
\noindent
\setlength{\boxwidth}{\textwidth}%
\addtolength{\boxwidth}{-2\fbboxsep}%
\addtolength{\boxwidth}{-2\fbboxrule}%
\fbbox{\parbox{\boxwidth}{\@fillinfont\@comments}}
}
\endinput

```

## A More Complicated Form

This example is more complicated. It is three pages long and has a box on page one that's continued on page two. All the boxes are of fixed height and shouldn't shrink or expand to fit the text. A sample document might be created as follows:

```

\documentclass{complicatedform}

\begin{document}
\collaborator{Dr A Person}{CMP}{1234}
\collaborator{Prof A.N. Other}{ENV}{5678}
\otherfunding{This proposal has also been submitted to
the intergalactic federation of Nog last year.}
\begin{projectdescription}
This is a really interesting project!
\end{projectdescription}
\training{The student will be trained in intergalactic space
exploration.}
\monitored{Telepathically}
\end{document}

```

Note that in this example, a blank form to be filled in by hand can be obtained using:

```

\documentclass{complicatedform}
\begin{document}
\begin{projectdescription}
\end{projectdescription}
\end{document}

```

The first page should look like:



### A Complicated Form

Collaborators

Name(s)

Institute/School Ext No

Has this proposal been submitted for funding elsewhere? YES/NO. If "YES" please provide details.

*Do not exceed box*

Description of the project

*Continue overleaf*

The second page should look like:

Continued from previous page

The third page should look like:

Give details of the training the student will receive.

Do not exceed box

How will the student be monitored?

Do not exceed box

## Identifying the Class File and Loading Required Packages and Underlying Class

I'm going to call this class file "complicatedform", so I need to create a file called "complicatedform.cls". The first two lines specify the TeX format and identify this class:

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesClass{complicatedform}[2009/11/20 v1.0 A Complicated Form]
```

As with the previous example, this class file is based on the article class, but this time the normal font size needs to be 10pt:

```
\LoadClass[10pt]{article}
```

I'm going to need the [flowfram](#) and [pgf](#) packages:

```
\RequirePackage{flowfram}
\RequirePackage{pgf}
```

I'm also going to need the [ulem](#) package for its strike-through command `\sout`:

```
\RequirePackage[normalem]{ulem}
```

## Page Layout

The paper size and margins can be set using the geometry package:

```
\RequirePackage[a4paper,portrait]{geometry}
\geometry{lmargin=72.0bp,rmargin=72.0bp,tmargin=72.0bp,bmargin=72.0bp}
```

This specifies A4 portrait paper with 1in margins.

On the first page, I want a frame with the title, the list of collaborators, the funding box and the first part of the description box. The first three boxes can go in "dynamic" frames, but the description needs to flow from the box at the end of the first page into the box on the second page. These two boxes should therefore be "flow" frames. The two boxes on the third page (for training and monitored details) can also be dynamic frames. Determining the parameters for these frames can be a bit tricky, so I cheated and used my graphics application [jpgfdraw](#) to create them. The code it produced to define the frames is as follows:

Create a frame with label "title" with its contents centred vertically:

```
\newdynamicframe[1]{451.0bp}{73.0bp}{-0.5bp}{625.391068bp}[title]
\setdynamicframe*{title}{valign=c}
```

The contents of the title frame must be set:

```
\setdynamiccontents*{title}{\centering\Large\bfseries
A Complicated Form\par}
```

Create a frame with label "collaborators" with a border:

```
\newdynamicframe[1]{431.0bp}{179.0bp}{9.5bp}{419.391068bp}[collaborators]

%Border command for frame 'collaborators'
\expandafter\def\csname @flf@border@collaborators\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{199.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{198.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{198.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{198.5bp}}
\pgfclosepath
\color{rgb}{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setdynamicframe*
{collaborators}{offset=0pt,border={@flf@border@collaborators}}

\setdynamicframe*{collaborators}{valign=t}
```

Create a frame with label "fundingelsewhere" with a border:

```
\newdynamicframe[1]{431.0bp}{143.0bp}{9.5bp}{239.391068bp}[fundingelsewhere]

%Border command for frame 'fundingelsewhere'
\expandafter\def\csname @flf@border@fundingelsewhere\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{163.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{162.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{162.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{162.5bp}}
\pgfclosepath
\color{rgb}{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setdynamicframe*
{fundingelsewhere}{offset=0pt,border={@flf@border@fundingelsewhere}}

\setdynamicframe*{fundingelsewhere}{valign=t}
```

Create a frame labelled "description1" with a border that contains the text "Continue overleaf" at the bottom:

```
\newflowframe[1]{431.0bp}{187.0bp}{9.5bp}{15.391068bp}[description1]

%Border command for frame 'description1'
\expandafter\def\csname @flf@border@description1\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{217.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-20.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{216.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{216.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{216.5bp}}
\pgfclosepath
\color{rgb}{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
\begin{pgfscope}
\pgftransformcm{1.0}{-0.0}{0.0}{1.0}{\pgfpoint{448.529542bp}{4.703406bp}}
\pgftext[right,base]{\@formfont\em\normalsize
\color{rgb}{0.0,0.0,0.0}Continue overleaf}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setflowframe*
{description1}{offset=0pt,border={@flf@border@description1}}
```

Create a frame labelled "description2" with a border that contains the text "Continued from previous page" at the top:

```
\newflowframe[2]{431.0bp}{673.0bp}{9.5bp}{5.391068bp}[description2]

%Border command for frame 'description2'
\expandafter\def\csname @flf@border@description2\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{703.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{702.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{702.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{702.5bp}}
\pgfclosepath
\color{rgb}{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
\begin{pgfscope}
\pgftransformcm{1.0}{-0.0}{0.0}{1.0}{\pgfpoint{2.648204bp}{695.857377bp}}
\pgftext[left,top]{\@formfont\em\normalsize
\color{rgb}{0.0,0.0,0.0}Continued from previous page}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setflowframe*
{description2}{offset=0pt,border={@flf@border@description2}}
```

Create a frame labelled "training" with a border:

```
\newdynamicframe[3]{431.0bp}{269.0bp}{9.5bp}{419.391068bp}[training]

%Border command for frame 'training'
\expandafter\def\csname @flf@border@training\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{289.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
```

```

\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{288.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{288.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{288.5bp}}
\pgfclosepath
\color{rgb}{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setdynamicframe*
{training}{offset=0pt,border={@flf@border@training}}

\setdynamicframe*{training}{valign=t}

```

Create a frame labelled "monitored" with a border:

```

\newdynamicframe[3]{431.0bp}{377.0bp}{9.5bp}{5.391068bp}[monitored]

%Border command for frame 'monitored'
\expandafter\def\csname @flf@border@monitored\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{397.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{396.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{396.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{396.5bp}}
\pgfclosepath
\color{rgb}{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setdynamicframe*
{monitored}{offset=0pt,border={@flf@border@monitored}}

\setdynamicframe*{monitored}{valign=t}

```

At the end of each page, the flowfram package looks ahead at the next page for the next frame. To avoid problems, I've defined a flow frame to be displayed after page two even though it won't be needed:

```

\onecolumn[>2]

```

The form doesn't have any headers or footers:

```

\pagestyle{empty}

```

As with the previous example, I'm going to define commands that set the fonts used by the form:

```

\newcommand*{\@fillinfont}{\rmfamily}
\newcommand*{\@formfont}{\sffamily}

```

## Defining User Commands and Environments

Each collaborator can be specified using `\collaborator`. The first argument specifies the collaborator's name, the second argument specifies the school and the third argument the extension number. The entire list of collaborators will be stored in the internal command `\@collaborators`, and that in turn needs to go in the "collaborators" frame. First of all, initialise `\@collaborators` to do nothing:


```

\newcommand*{\@collaborators}{}

```

This will eventually go inside a tabular environment where each collaborator is on a row. The user command `\collaborator` must add its information to `\@collaborators` remembering to use `&` and `\\`

```
\newcommand*{\collaborator}[3]{%
  \protected@edef\@collaborators{\@collaborators
    #1 & #2 & #3\\}%
}
```

 If I'd used `\renewcommand` instead of `\protected@edef`, it would cause an infinite loop where `\@collaborators` calls itself recursively. Using `\protected@edef` expands the definition before applying it.

Now set the contents of the "collaborators" frame:

```
\setdynamiccontents*{collaborators}{%
  \@formfont Collaborators\par
  \vspace{\baselineskip}%
  \noindent\@fillinfont
  \begin{tabular}{@{}p{0.6\linewidth}ll@{}}
  \@formfont Name(s) & \@formfont Institute/School & \@formfont Ext
  No\\
  \@collaborators
  \end{tabular}
}
```

Next the user needs to specify if the proposal has been submitted elsewhere for funding. Two commands are going to be available for this: `\nootherfunding` (hasn't been submitted) or `\otherfunding{details}` (has been submitted and details supplied in the argument). If it hasn't been submitted elsewhere, the word "YES" must be crossed out and if it has the word "NO" must be crossed out and the information put in the "fundingelsewhere" frame. First two internal commands are needed to display "YES" and "NO", possibly crossed out:

```
\newcommand*{\@otherfunding@yes}{YES}
\newcommand*{\@otherfunding@no}{NO}
```

and an internal command that contains the details:

```
\newcommand*{\@otherfunding@details}{}
```

The command `\nootherfunding` should redefine the internals as appropriate:

```
\newcommand*{\nootherfunding}{%
  \renewcommand*{\@otherfunding@yes}{\sout{YES}}%
  \renewcommand*{\@otherfunding@no}{NO}%
  \renewcommand*{\@otherfunding@details}{}%
}
```

Similarly for `\otherfunding` which needs an argument:

```
\newcommand*{\otherfunding}[1]{%
  \renewcommand*{\@otherfunding@yes}{YES}%
  \renewcommand*{\@otherfunding@no}{\sout{NO}}%
  \renewcommand*{\@otherfunding@details}{#1}%
}
```

The information now needs to go in the "fundingelsewhere" frame:

```
\setdynamiccontents*{fundingelsewhere}{%
  \@formfont Has this proposal been submitted for funding elsewhere?
  \@otherfunding@yes/\@otherfunding@no. If ``YES'' please provide
  details.\par\noindent
  \@fillinfont\@otherfunding@details
  \par\vfill\noindent\@formfont\normalsize\em Do not exceed box}
```

The frames labelled "description1" and "description2" are flow frames and therefore don't have their contents set because that's where the main document text goes. If insufficient text is given in the project description, the second page may not get processed because from TeX's point of view the document has finished. The "projectdescription" environment can check if the description has gone over into the second page. If not, it will force a new page. A new page is then forced after the description to ensure that the third page is shipped out:

```
\newenvironment{projectdescription}%
{\noindent\@formfont Description of the project\par\@fillinfont}%
{\par\null\ifnum\c@page=1\relax\null\newpage\null\fi\newpage\null}
```

The training information is stored in the internal command \@training:

```
\newcommand*{\@training}{}
```

The user command \training sets the information:

```
\newcommand*{\training}[1]{\renewcommand*{\@training}{#1}}
```

The contents of the "training" frame needs to be set:

```
\setdynamiccontents*{training}{\@formfont\noindent
Give details of the training the student will receive.
\par\noindent\@fillinfont\@training\par
\vfill\noindent\@formfont\normalsize\em Do not exceed box}
```

Similarly for the monitored information:

```
\newcommand*{\@monitored}{}
\newcommand*{\@monitored}[1]{\renewcommand*{\@monitored}{#1}}

\setdynamiccontents*{monitored}{\@formfont\noindent
How will the student be monitored?
\par\noindent\@fillinfont\@monitored\par
\vfill\noindent\@formfont\normalsize\em Do not exceed box}
```

Finally, the class file needs the line:

```
\endinput
```

## The Complete Class File

The complete code for "complicatedform.cls" is given below:

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesClass{complicatedform}[2009/11/20 v1.0 A Complicated Form]
\LoadClass[10pt]{article}

\RequirePackage{flowfram}
\RequirePackage{pgf}
\RequirePackage[normalem]{ulem}
\RequirePackage[a4paper,portrait]{geometry}

\geometry{lmargin=72.0bp,rmargin=72.0bp,tmargin=72.0bp,bmargin=72.0bp}
\newdynamicframe[1]{451.0bp}{73.0bp}{-0.5bp}{625.391068bp}[title]

\setdynamicframe*{title}{valign=c}
\newdynamicframe[1]{431.0bp}{179.0bp}{9.5bp}{419.391068bp}[collaborators]

%Border command for frame 'collaborators'
\expandafter\def\csname @flf@border@collaborators\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{199.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{198.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{198.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{198.5bp}}
\pgfclosepath
\color{rgb}{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
}
```



```

\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setdynamicframe*
{collaborators}{offset=0pt,border={@flf@border@collaborators}}

\setdynamicframe*{collaborators}{valign=t}
\newdynamicframe[1]{431.0bp}{143.0bp}{9.5bp}{239.391068bp}[fundingelsewhere]

%Border command for frame 'fundingelsewhere'
\expandafter\def\csname @flf@border@fundingelsewhere\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{163.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{162.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{162.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{162.5bp}}
\pgfclosepath
\color[rgb]{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setdynamicframe*
{fundingelsewhere}{offset=0pt,border={@flf@border@fundingelsewhere}}

\setdynamicframe*{fundingelsewhere}{valign=t}
\newflowframe[1]{431.0bp}{187.0bp}{9.5bp}{15.391068bp}[description1]

%Border command for frame 'description1'
\expandafter\def\csname @flf@border@description1\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{217.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-20.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{216.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{216.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{216.5bp}}
\pgfclosepath
\color[rgb]{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
\begin{pgfscope}
\pgftransformcm{1.0}{-0.0}{0.0}{1.0}{\pgfpoint{448.529542bp}{4.703406bp}}
\pgftext[right,base]{\@formfont\em\normalsize
\color[rgb]{0.0,0.0,0.0}Continue overleaf}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setflowframe*
{description1}{offset=0pt,border={@flf@border@description1}}

\newflowframe[2]{431.0bp}{673.0bp}{9.5bp}{5.391068bp}[description2]

%Border command for frame 'description2'
\expandafter\def\csname @flf@border@description2\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{703.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{702.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{702.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{702.5bp}}

```

```

\pgfclosepath
\color[rgb]{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
\begin{pgfscope}
\pgftransformcm{1.0}{-0.0}{0.0}{1.0}{\pgfpoint{2.648204bp}{695.857377bp}}
\pgftext[left,top]{\@formfont\em\normalsize
\color[rgb]{0.0,0.0,0.0}Continued from previous page}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setflowframe*
{description2}{offset=0pt,border={@flf@border@description2}}

\newdynamicframe[3]{431.0bp}{269.0bp}{9.5bp}{419.391068bp}[training]

%Border command for frame 'training'
\expandafter\def\csname @flf@border@training\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{289.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{288.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{288.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{288.5bp}}
\pgfclosepath
\color[rgb]{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setdynamicframe*
{training}{offset=0pt,border={@flf@border@training}}

\setdynamicframe*{training}{valign=t}
\newdynamicframe[3]{431.0bp}{377.0bp}{9.5bp}{5.391068bp}[monitored]

%Border command for frame 'monitored'
\expandafter\def\csname @flf@border@monitored\endcsname#1{%
\begin{pgfpicture}{0bp}{0bp}{451.0bp}{397.0bp}
\pgfputat{\pgfpoint{-10.0bp}{-10.0bp}}{%
\begin{pgfscope}
\pgfsetlinewidth{1.0bp}
\pgfsetbuttcap
\pgfsetroundjoin
\pgfpathmoveto{\pgfpoint{0.5bp}{396.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{0.5bp}}
\pgfpathlineto{\pgfpoint{450.5bp}{396.5bp}}
\pgfpathlineto{\pgfpoint{0.5bp}{396.5bp}}
\pgfclosepath
\color[rgb]{0.0,0.0,0.0}
\pgfusepath{stroke}
\end{pgfscope}
}
\pgfputat{\pgfpoint{0bp}{0bp}}{\pgftext[left,bottom]{#1}}
\end{pgfpicture}}
\setdynamicframe*
{monitored}{offset=0pt,border={@flf@border@monitored}}

\setdynamicframe*{monitored}{valign=t}

\onecolumn[>2]

\pagestyle{empty}

\newcommand*{\@fillinfont}{\rmfamily}
\newcommand*{\@formfont}{\sfamily}

\setdynamiccontents*{title}{\centering\Large\bfseries
A Complicated Form\par}

```

```

\newcommand*{\@collaborators}{}

\newcommand*{\collaborator}[3]{%
  \protected@edef\@collaborators{\@collaborators
    #1 & #2 & #3\\}%
}

\setdynamiccontents*{collaborators}{%
  \@formfont Collaborators\par
  \vspace{\baselineskip}%
  \noindent\@fillinfont
  \begin{tabular}{@{}p{0.6\linewidth}ll@{}}
  \@formfont Name(s) & \@formfont Institute/School &
  \@formfont Ext No\\
  \@collaborators
  \end{tabular}
}

\newcommand*{\@otherfunding@yes}{YES}
\newcommand*{\@otherfunding@no}{NO}
\newcommand*{\@otherfunding@details}{}

\newcommand*{\nootherfunding}{%
  \renewcommand*{\@otherfunding@yes}{\sout{YES}}%
  \renewcommand*{\@otherfunding@no}{NO}%
  \renewcommand*{\@otherfunding@details}{}%
}

\newcommand*{\otherfunding}[1]{%
  \renewcommand*{\@otherfunding@yes}{YES}%
  \renewcommand*{\@otherfunding@no}{\sout{NO}}%
  \renewcommand*{\@otherfunding@details}{#1}%
}

\setdynamiccontents*{fundingelsewhere}{%
  \@formfont Has this proposal been submitted for funding elsewhere?
  \@otherfunding@yes/\@otherfunding@no. If ``YES'' please provide
  details.\par\noindent
  \@fillinfont\@otherfunding@details
  \par\vfill\noindent\@formfont\normalsize\em Do not exceed box}

\newenvironment{projectdescription}%
{\noindent\@formfont Description of the project\par\@fillinfont}%
{\par\null\ifnum\c@page=1\relax\null\newpage\null\fi\newpage\null}

\newcommand*{\@training}{}

\newcommand*{\training}[1]{\renewcommand*{\@training}{#1}}

\setdynamiccontents*{training}{\@formfont\noindent
  Give details of the training the student will receive.
  \par\noindent\@fillinfont\@training\par
  \vfill\noindent\@formfont\normalsize\em Do not exceed box}

\newcommand*{\@monitored}{}
\newcommand*{\monitored}[1]{\renewcommand*{\@monitored}{#1}}

\setdynamiccontents*{monitored}{\@formfont\noindent
  How will the student be monitored?
  \par\noindent\@fillinfont\@monitored\par
  \vfill\noindent\@formfont\normalsize\em Do not exceed box}

\endinput

```

## Comments

Og DeSouza

+3  

Excellent! Thank you. This was clear and straightforward. I always wanted to learn how to build a LaTeX class but never found instructions so easy to follow as the ones you provided here.

**Tuesday 12 June 2012, 13:07**

Please login to post comments or replies.

## Latest Forum Posts

### Re: Metafile to EPS converter: should it work from within LY

01/11/2016 15:21, scottkosty

Ah OK, I see. This is a bit a pitty, since I strongly believe that what LyX does with LaTeX is exactly what "Windows"-people want/need: something "easy"/"user-friendly"/"working-out-of-the-box"/"easy-GUI"/WYSIWYM. Yes it is too bad we do not have more developers on W...

### Re: Use TikZ with LyX

01/11/2016 14:34, jonjoexx

Thank you so much for posting this I am new to Lyx and spent two days trying to get tikz to work....

### Re: Metafile to EPS converter: should it work from within LY

01/11/2016 09:49, Stef Pillaert

- There is only one LyX developer who uses Windows. Most use Linux. The developer on Windows does what he can and dedicates a lot of time, but it is hard when he is the only one. Ah OK, I see. This is a bit a pitty, since I strongly believe that what LyX does with LaTeX is exactly what "Windows"-people wan...

### Re: Metafile to EPS converter: should it work from within LY

01/11/2016 00:07, scottkosty

OK, I did that for both emf and wmf to eps, now it works (though it takes quite a while the first time; also seems slow the first time I generate a pdf). But OK, it seems fine now. Thanks a lot for your time!!! Great! Sorry it took so long. Thanks for your patience. (Just out of curiosity: did I do something wrong...

### Re: Android

31/10/2016 19:16, Stefan Kottwitz

It is good to see Lyx in the cloud. Yes! lyx-cloud.jpg (Click image to expand) On my Blackberry phone. Stefan...

### Re: Metafile to EPS converter: should it work from within LY

31/10/2016 17:24, Stef Pillaert

OK, I did that for both emf and wmf to eps, now it works (though it takes quite a while the first time; also seems slow the first time I generate a pdf). But OK, it seems fine now. Thanks a lot for your time!!! (Just out of curiosity: did I do something wrong while installing? Or does this happen on every Windows-machi...

### Re: Metafile to EPS converter: should it work from within LY

31/10/2016 14:51, scottkosty

"magick

*i*

o" OK, copy that command to the command for the converter that is not working. Then restart LyX. Does it work now?...

### Re: customizing screen font size

31/10/2016 14:50, scottkosty

In Tools > Preferences > Screen Fonts you can change the Zoom %....

### Re: hyperlinks in tikz

31/10/2016 06:55, omerangel

Looks great. Thanks!...

### customizing screen font size

30/10/2016 20:22, jalea148

My vision is poor and I would like everything larger - especally the fonts....