

Sample L^AT_EX Article

Glenn Murray

<http://www.mines.edu/~gmurray>

January 11, 2007

Abstract

A template file for latex files.

Contents

1	Introduction	1
1.1	Colors And Such	1
2	Images	2
3	Source Code Listings	3

1 Introduction

I recommend Tobi Oetiker's *The Not So Short Introduction to L^AT_EX 2 ϵ* at <http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf>. In a PDF file, the title, the Contents section, the last sentence, and the next three sentences contain clickable links.

Note the default font is Times New Roman. Some people prefer this to the default L^AT_EX font¹.

Here are examples of references: for information about using colors, see Section 1.1, for information about using colors, see Section 3.

Here is an example of a citation: Read about agile[1].

Of course, L^AT_EX is the best way to typeset mathematics, $\mathbb{R}: \sum_{i=0}^{\infty} \cos nx$.

1.1 Colors And Such

Here is some colored text. Text in a framebox, colorbox, fcolorbox, Text, Text

¹An unnecessary footnote.

2 Images

Unfortunately, \LaTeX needs its images in different formats depending on whether you are creating Postscript or PDF. For Postscript, the image should be Postscript (.ps) or Encapsulated Postscript (.eps), and, unsurprisingly, the format for PDF is PDF (.pdf). Images can be converted from .jpg, etc., to .eps and .pdf formats with [ImageMagick](#)'s `convert` command. For example,

```
$ convert WomanWithMargarita.jpg WomanWithMargarita.eps
```

The `convert` command can be used to do batch conversions, too.



Figure 1: Woman with a margarita.

Note that it is not necessary to use the “.eps” extension in your \LaTeX source file, \LaTeX will find the file from the basename, so there really is no issue with trying to keep track of two kinds of files; e.g., `\includegraphics[width=1.5in]{WomanWithMargarita}` works with both `WomanWithMargarita.eps` and `WomanWithMargarita.pdf`.

Images are scaled by setting the image width. In Figure 1 an image with a caption is centered on the page. Figure 2 is the same image file, but the image is scaled to a different width.



Floating figures are somewhat problematic. I have used two packages, `floatflt` and `wrapfig`, neither of which is perfect. I have had a little more success with the latter. In Figure 2 the image is wrapped by the surrounding text using `wrapfig`. The optional caption has been omitted. Getting an acceptable look often means trial and error, as the floating images seem to collide with other elements. Another package worth investigating is `picins`.

To arrange more than one image in a figure there are several solutions which can be found on the web; including the `subfigure` package.

3 Source Code Listings

Code listings can be pasted into the .tex file or imported with `\lstinputlisting{Test.java}`. Here is a margin note about the class declaration (“public class Test”) in Listing 1, with a reference to the correct line number. This is good for annotating text, but requires a `label{}` in the code listing; the input method lets the code be updated outside the .tex file. *line 4*

Plain listing:

```
<html>
  <body bgcolor="#dddddd">
    <applet code="DataCollector.class" archive="DataCollector.jar" width="150"
      height="50">
      <param name="DataZipTargetDir" value="H:\\SCC-FTIR-database" />
    </applet>
  </body>
</html>
```

Fancy listing: this listing is set wide enough to show 80 columns of this size font without wrapping.

```
1 /**
   * A short class for testing.
   */
3 public class Test {
5     public static void main(String[] args) throws Exception {
        System.out.println("user.name, "+System.getProperty("user.name")); // 80
7     }
}
```

Listing 1: Here is a Test class

The listing package provides many customizations. Another possibility is shown in the next example, Listing 2. This one demonstrates line breaks and a caption on top.

Listing 2: Here is another listing.

```
1 <html>
2   <body bgcolor="#dddddd">
3     <applet code="DataCollector.class" archive="DataCollector.jar" width="150"
4       height="50">
5       <param name="DataZipTargetDir" value="H:\\SCC-FTIR-database" />
6     </applet>
7   </body>
</html>
```

That’s all folks!

References

[1] The Agile Alliance, at this [link](#).

```

% Here is the source code.
%
% $Id$ (An RCS/CVS version control tag).
\RequirePackage{ifpdf}          % Different settings for PDF's.
\ifpdf
  \documentclass[12pt,pdftex]{article}
  \usepackage{hyperref}
  \hypersetup{colorlinks, citecolor=MyDarkRed, filecolor=MyDarkBlue,
    linkcolor=MyDarkRed, urlcolor=MyDarkBlue}
\else
  \documentclass[12pt,dvips]{article}
  \usepackage{hyperref}
  \hypersetup{colorlinks, citecolor=MyDarkRed, filecolor=MyDarkBlue,
    linkcolor=MyDarkRed, urlcolor=MyDarkBlue}
  \usepackage{breakurl}        % Break url's in ps output, has to be
                                % after hyperref.
\fi

% Packages
\usepackage[in]{fullpage}      % Sets up 1 in margins all around.
\usepackage{amsmath,amssymb} % Some math symbols, like \mathbb

% About fonts: default is Computer Modern and looks ``TeX''y to some.
% times          Times, Helvetica, Courier
% pslatex same as Times, but uses a specially narrowed Courier. This is
%              preferred over Times because of the way it handles Courier.
% newcent New Century Schoolbook, Avant Garde, Courier
% palatino Palatino, Helevetica, Courier
% palatcm changes the Roman to Palatino only, but uses CM mathematics
\usepackage{pslatex} % Change font
% For dvipsnam.def Color names, ps only. For PDF, lower case color
% names work, but there aren't many. The named dvips colors must be
% defined by hand.
\usepackage[usenames]{color}
\definecolor{MyDarkBlue}{rgb}{0, 0.0, 0.45}
\definecolor{MyDarkRed}{rgb}{0.45, 0.0, 0}
\definecolor{MyDarkGreen}{rgb}{0, 0.45, 0}
\definecolor{MyLightGray}{gray}{.90}
\definecolor{MyLightGreen}{rgb}{0.5, 0.99, 0.5}
% From dvipsnam.def
\DefineNamedColor{named}{Peach}          {cmyk}{0,0.50,0.70,0}
\DefineNamedColor{named}{Cyan}          {cmyk}{1,0,0,0}

```

```

% Images
\usepackage{graphicx} % For including images
% For floating (text-wrapped) figures
\usepackage{floatflt}
\usepackage{wrapfig}
% \begin{wrapfigure}[height of figure in lines]{l|r,...}[overhang]{width}
%   graphic, caption, etc.
% \end{wrapfigure}
%
% Height can be omitted, in which case it will be calculated by the
% package; the package will use the greater of the specified and the
% actual width. The {l,r,etc.} parameter can also be specified as
% i(nside) or o(utside) for two-sided documents, and uppercase can be
% used to indicate that the picture should float. The overhang allows
% the figure to be moved into the margin.

\usepackage{listings}          % For source code fragments, many options
\lstset{
  language=Java,                % choose the language of the code
  basicstyle=\footnotesize,    % code font size
  numbers=left,                 % where to put the line-numbers
  numberstyle=\footnotesize,   % line number font size
  stepnumber=0,                % the step between two line-numbers.
  backgroundcolor=\color{white}, % choose the background color
  frame=single,
  framerule=0pt,
  captionpos=b,                % t or b
  showstringspaces=false,      % underline spaces within strings
  showspaces=false,            % show spaces within strings with underscores
  showtabs=false,              % show tabs within strings with underscores
  breaklines=true              % Break long lines of code
}
\lstset{escapeinside={(*@}{@*)}} % To add labels in listing, needs own line

% Misc.
\pagecolor{white} % For latex2html image backgrounds
%\setlength{\parskip}      {\baselineskip}
%\setlength{\parindent}    {0in}
% Some bullet possibilities: \bullet (default), \circ, \diamond, \ast, \prec
\renewcommand{\labelitemi}{\circ}

```

```

% Abbreviations
\newcommand{\my}{MySQL}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\begin{document}

%\tableofcontents           % This can go after the abstract, too.
%\begin{titlepage}         % Uncomment for a separate title page
\title{Sample \LaTeX Article} % type title between braces
\author{Glenn Murray \\\   % type author(s) between braces
        \url{http://www.mines.edu/~gmurray}}
\date{\today}              % type specific date between braces
\maketitle
%\end{titlepage}

\begin{abstract}
  A template file for latex files.
\end{abstract}

\tableofcontents

\section{Introduction} % section 1; use ``\section*'' to omit numeration

I recommend Tobi Oetiker's \emph{The Not So Short Introduction to
  \LaTeX2 $\epsilon$} at
\url{http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf}.
In a PDF file, the title, the Contents section, the last sentence, and
the next three sentences contain clickable links.

Note the default font is Times New Roman. Some people prefer this to
the default \LaTeX\ font\footnote{An unnecessary footnote.}.

Here are examples of references: for information about using colors,
see Section~\ref{sec:colors}, for information about using colors, see
Section~\ref{sec:listings}.

Here is an example of a citation: Read about agile\cite{agile}.

Of course, \LaTeX\ is the best way to typeset mathematics,
 $\mathbb{R}$ :  $\sum_{i=0}^{\infty} \cos n x$ .

```

```

\subsection{Colors And Such}          \label{sec:colors} % subsection 1.1
Here is some \textcolor{MyDarkGreen}{colored} \textcolor{cyan}{text}.
\framebox[1.1\width]{Text in a framebox}, \colorbox{yellow}{colorbox},
\newlength{\savedfboxrule}
\setlength{\savedfboxrule}{\fboxrule} \setlength{\fboxrule}{2pt}
\fcolorbox{red}{MyLightGreen}{fcolorbox},
\framebox[2\width]{Text},
\setlength{\fboxrule}{\savedfboxrule}
\framebox[1.1\width]{Text}

```

```

\section{Images}                      % section 2
Unfortunately, \LaTeX\ needs its images in different formats depending
on whether you are creating Postscript or PDF. For Postscript, the
image should be Postscript (.ps) or Encapsulated Postscript (.eps),
and, unsurprisingly, the format for PDF is PDF (.pdf). Images can be
converted from .jpg, etc., to .eps and .pdf formats with
\href{http://www.imagemagick.org/}{ImageMagick}'s \path{convert}
command. For example,
\begin{quotation}
  \texttt{\$ convert WomanWithMargarita.jpg WomanWithMargarita.eps}
\end{quotation}
The \path{convert} command can be used to do batch conversions, too.

```

```

\begin{figure}[th!]
  \begin{center}
    \includegraphics[width=2in]{WomanWithMargarita}
    \caption{Woman with a margarita.}
    \label{fig:womancenter}
  \end{center}
\end{figure}

```

Note that it is not necessary to use the ``.eps'' extension in your \LaTeX\ source file, \LaTeX\ will find the file from the basename,

```

\begin{wrapfigure}[10]{L}[0mm]{1.6in}
  \includegraphics[width=1.5in]{WomanWithMargarita}
  %\caption{Woman with a margarita.}
  \label{fig:floatwoman}
\end{wrapfigure}

```

so there really is no issue with trying to keep track of two kinds of files; e.g., \path{\includegraphics[width=1.5in]{WomanWithMargarita}}

works with both `\path{WomanWithMargarita.eps}` and `\path{WomanWithMargarita.pdf}`.

Images are scaled by setting the image width. In Figure~\ref{fig:womancenter} an image with a caption is centered on the page. Figure~\ref{fig:floatwoman} is the same image file, but the image is scaled to a different width.

Floating figures are somewhat problematic. I have used two packages, `\path{floatflt}` and `\path{wrapfig}`, neither of which is perfect. I have had a little more success with the latter. In Figure~\ref{fig:floatwoman} the image is wrapped by the surrounding text using `\path{wrapfig}`. The optional caption has been omitted. Getting an acceptable look often means trial and error, as the floating images seem to collide with other elements. Another package worth investigating is `picins`.

To arrange more than one image in a figure there are several solutions which can be found on the web; including the `subfigure` package.

```
\section{Source Code Listings}          \label{sec:listings}    % section 3
Code listings can be pasted into the .tex file or imported with
\texttt{\$\backslash$lstinputlisting\{Test.java\}}. Here is a margin
note about the class declaration
\marginpar{\emph{line~\ref{lis:clasdec}}} (``public class Test'') in
Listing~\ref{lis:TestClass}, with a reference to the correct line
number. This is good for annotating text, but requires a
\texttt{label\{\}} in the code listing; the input method lets the code
be updated outside the .tex file.
```

Plain listing:

```
\begin{lstlisting}[language=HTML]
<html>
  <body bgcolor="#dddddd">
    <applet code="DataCollector.class" archive="DataCollector.jar" width="150" height="50">
      <param name="DataZipTargetDir" value="H:\SCC-FTIR-database" />
    </applet>
  </body>
</html>
\end{lstlisting}
```

Fancy listing: this listing is set wide enough to show 80 columns of this size font without wrapping.

```
\begin{lstlisting}[caption={Here is a Test class}, label={lis:TestClass},
  xleftmargin=0pt, xrightmargin=-12pt, % For 80 column lines
  stepnumber=2, backgroundcolor=\color{MyLightGray}
]
/**
 * A short class for testing.
 */
public class Test {          (*@ \label{lis:clasdec} @*)
  public static void main(String[] args) throws Exception {
    System.out.println("user.name, "+System.getProperty("user.name")); // 80
  }
}
\end{lstlisting}
```

The listing package provides many customizations. Another possibility is shown in the next example, Listing~\ref{lis:html}. This one demonstrates line breaks and a caption on top.

```
\begin{lstlisting}[language=HTML, framerule=0.7pt, frameround=tttt,
  rulecolor=\color{blue}, stepnumber=1,
  caption={Here is another listing.}, captionpos=t, label=lis:html]
<html>
  <body bgcolor="#dddddd">
    <applet code="DataCollector.class" archive="DataCollector.jar" width="150" height="50"
      <param name="DataZipTargetDir" value="H:\SCC-FTIR-database" />
    </applet>
  </body>
</html>
\end{lstlisting}
```

That's all folks!

```
\begin{thebibliography}{9}
  % type bibliography here
  \bibitem{agile} The Agile Alliance, at this
  \href{http://www.agilealliance.com}{link}.
\end{thebibliography}
```

```
\end{document}
```

%%%