LaTeX and MiKTeX
Introduction Part 1

http://www.win.tue.nl/~jknopper/latex/intro/

TU/e Technische Universiteit **Eindhoven** University of Technology

November 27, 2012 Jan Willem Knopper (jknopper@win.tue.nl)

Contents

Introduction

LaTeX is a document preparation system. It is widely used in the fields of mathematics and natural sciences, but also spreading to many other disciplines.

- LaTeX is a set of markup commands used with the powerful typesetting program TeX.
- totally open software system, free of charge.
- platform independent.

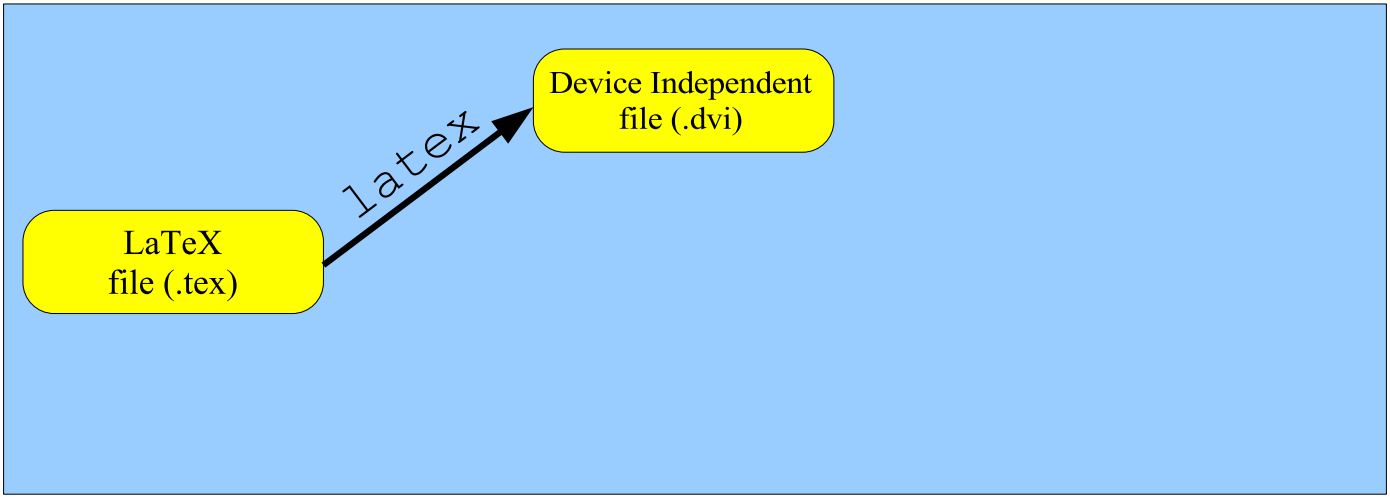
LaTeX is no word processor! LaTeX stimulates placing emphasis on **content** (logical markup) instead of **appearance** (typographical markup).

Introduction

Introduction

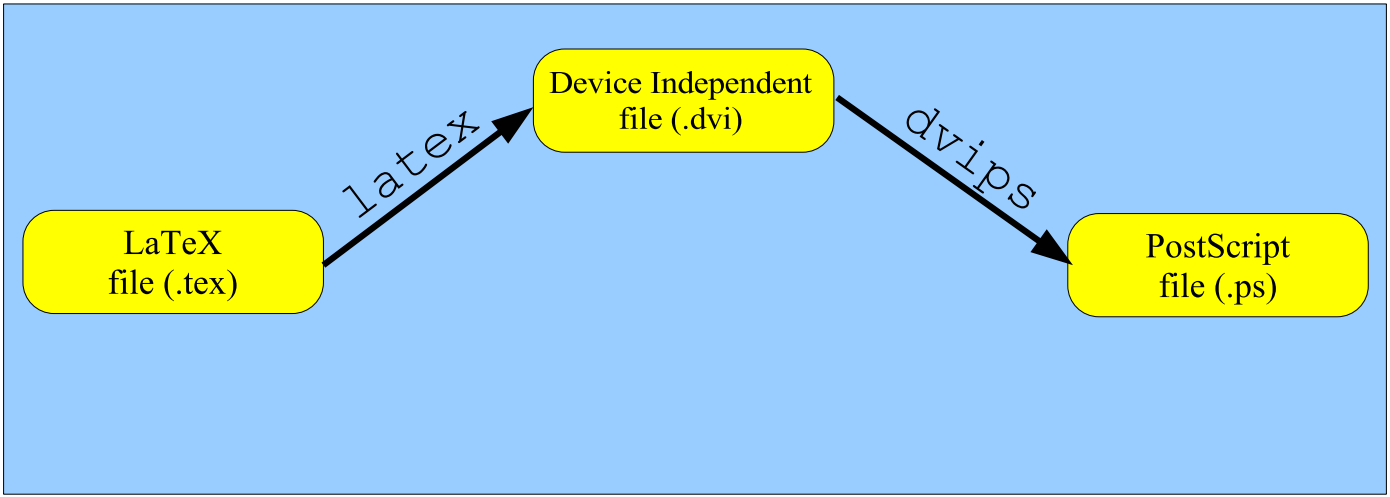
LaTeX editor: WinEdt

Introduction



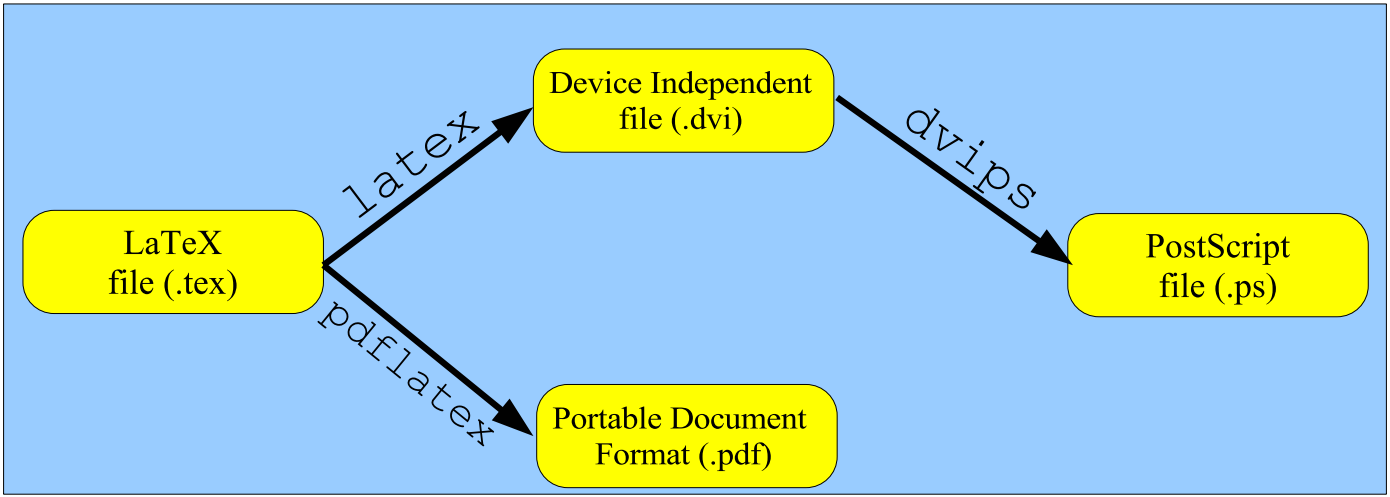
DVI previewer: Yap

Introduction



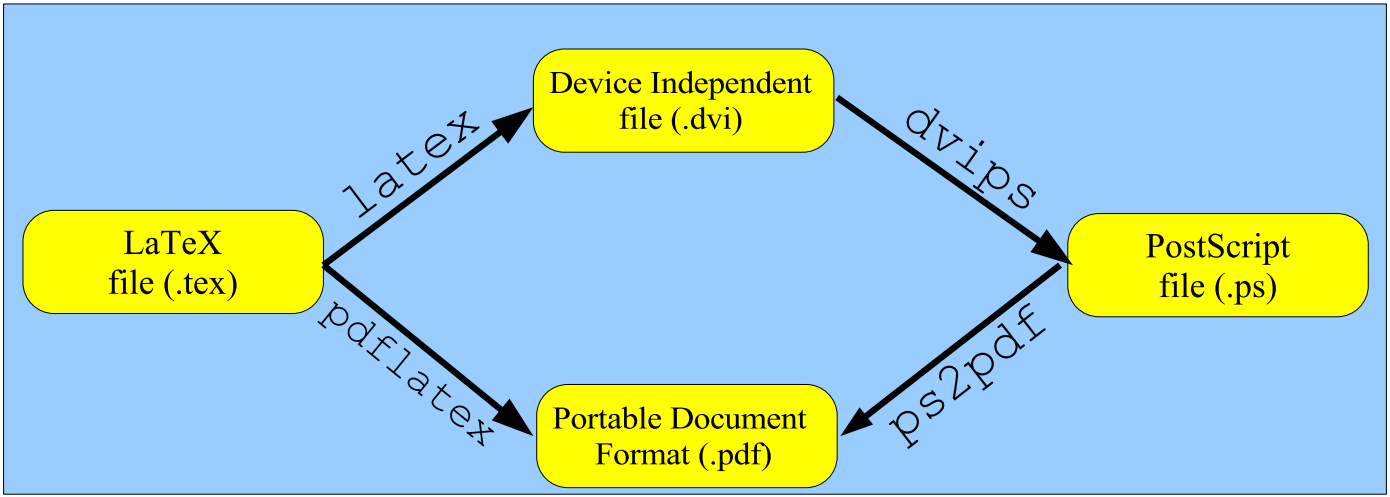
PostScript viewer: GSView

Introduction



PDF viewer: Adobe (Acrobat) Reader

Introduction



The LaTeX language

- LaTeX commands always start with a backslash: \

The LaTeX language

- LaTeX commands always start with a backslash: \
- required command arguments are placed between curly brackets: { }

The LaTeX language

- LaTeX commands always start with a backslash: \
- required command arguments are placed between curly brackets: { }
- optional command arguments are placed between brackets: [ ]

The LaTeX language

- LaTeX commands always start with a backslash: \
- required command arguments are placed between curly brackets: { }
- optional command arguments are placed between brackets: [ ]
- comments start with a percentage symbol: %

Introduction

The LaTeX language

- LaTeX commands always start with a backslash: \
- required command arguments are placed between curly brackets: { }
- optional command arguments are placed between brackets: [ ]
- comments start with a percentage symbol: %
- LaTeX takes care of the spacing between words and paragraphs (just like HTML).

Introduction

The LaTeX language

- LaTeX commands always start with a backslash: \
- required command arguments are placed between curly brackets: { }
- optional command arguments are placed between brackets: [ ]
- comments start with a percentage symbol: %
- LaTeX takes care of the spacing between words and paragraphs (just like HTML).
- the commands \begin{ } and \end{ } create environments.

A .tex file

```
\documentclass[options]{document_class}

% preamble

\begin{document}

% document

\end{document}
```

A .tex file: intro.tex

```
\documentclass[12pt]{article}
\usepackage[english]{babel}

\begin{document}
\section{Introduction}

LaTeX is a document preparation system used to
create documents of high quality typography.

It is mostly used in the fields of mathematics
and natural sciences, but can in fact be used
for any type of publication.
\end{document}
```

MiKTeX

MiKTeX is an up-to-date TeX implementation for the Windows operating system.

- can be downloaded from http://www.miktex.org
- contains all LaTeX related binaries, like
  `latex.exe, pdflatex.exe, yap.exe, bibtex.exe, dvips.exe, ps2pdf.exe`
- contains all standard packages (will be discussed later)
- TU/e version with additional packages and fonts can be installed from (free) DVD or network drive.

LaTeX Related Programs

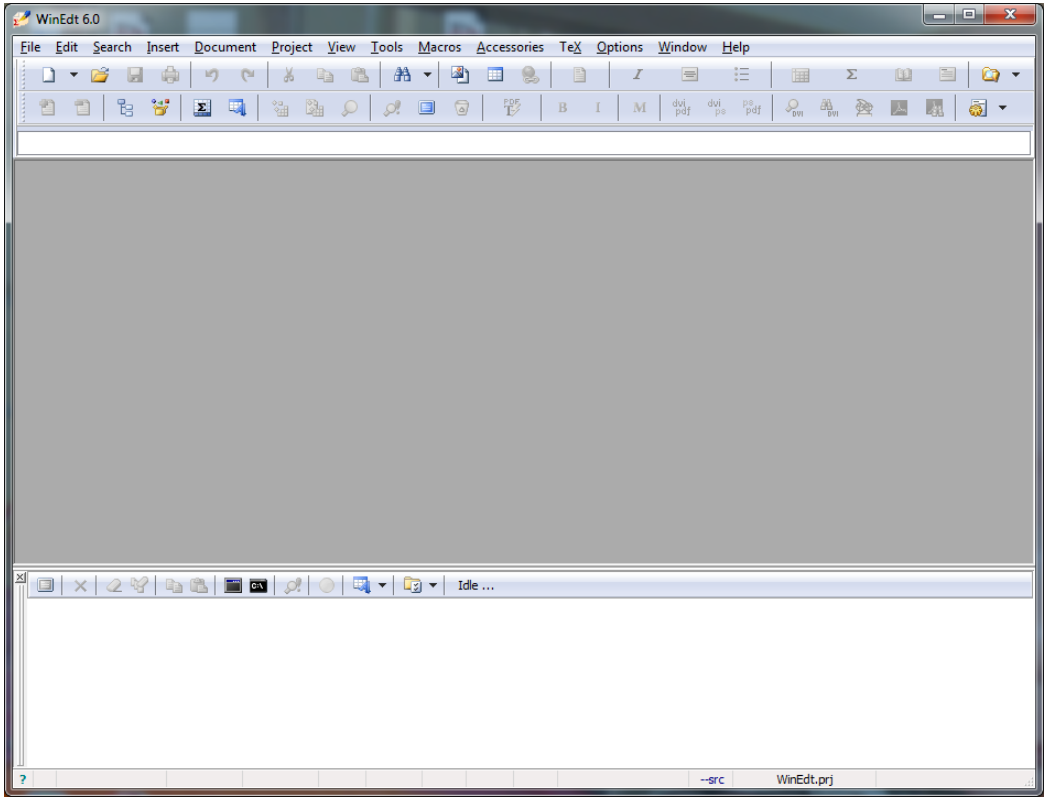
- WinEdt – editor
- Yap – DVI previewer
- GSView – PS previewer

Other useful programs:

- Adobe Acrobat or Adobe Reader – view/edit PDF files
- LibreOffice Draw – Create and export EPS Images

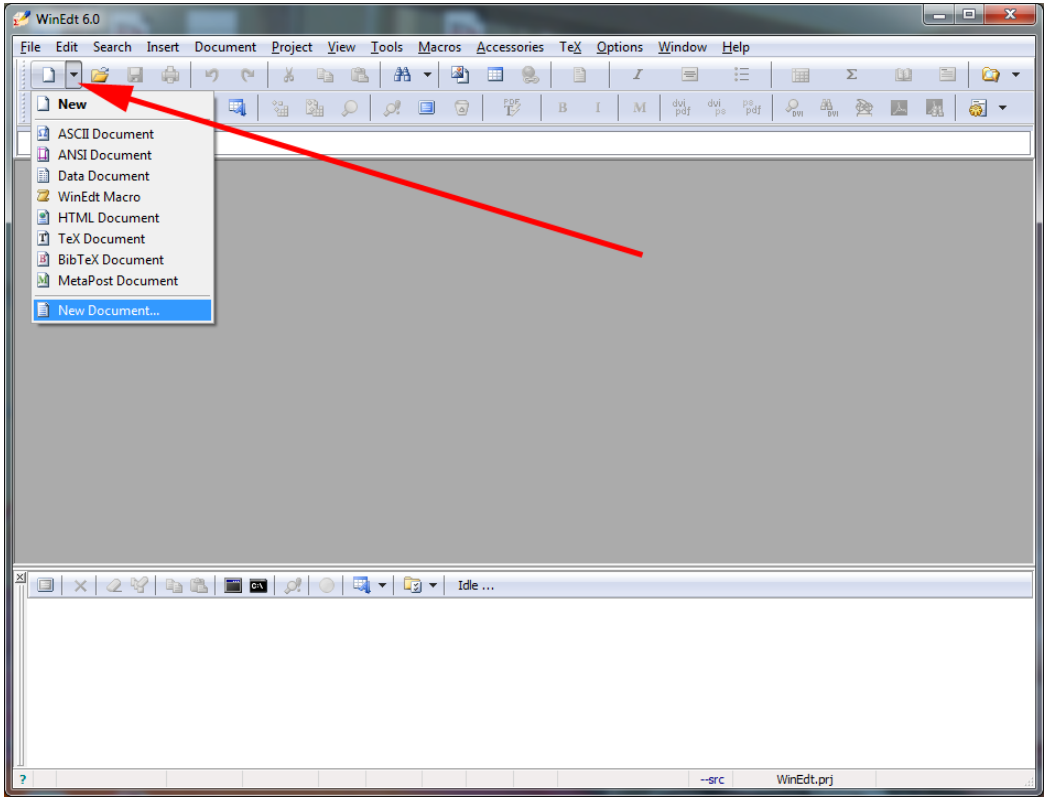
LaTeX Related Programs

WinEdt



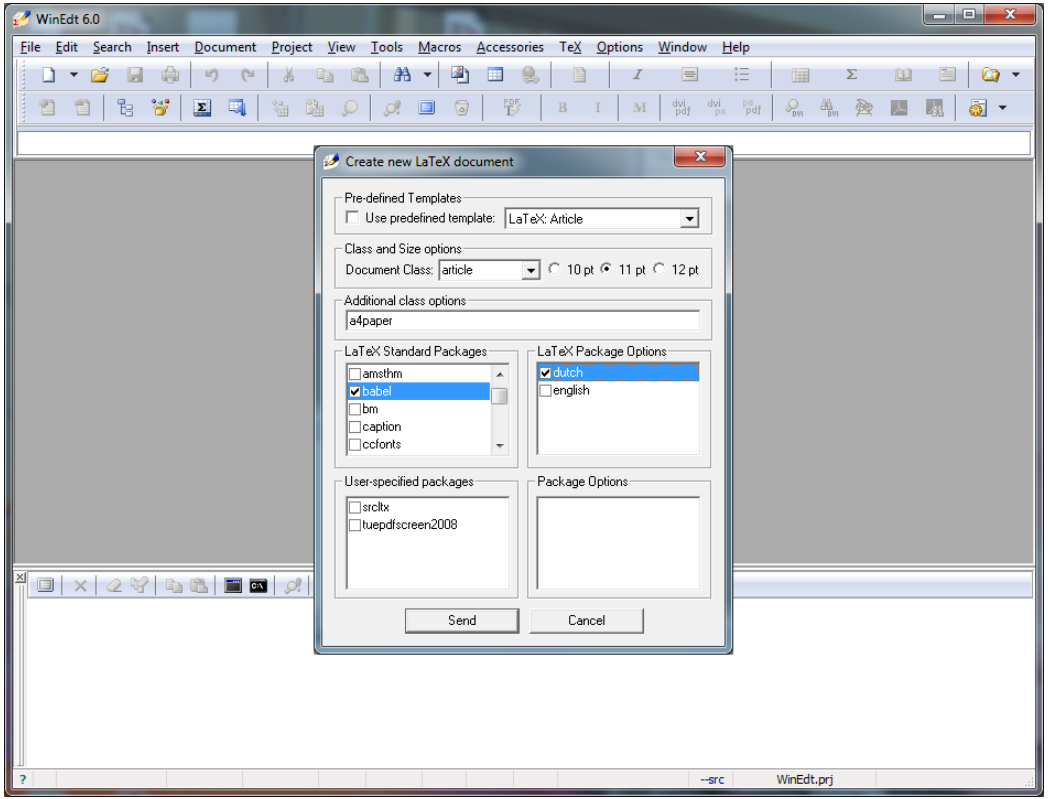
LaTeX Related Programs

WinEdt



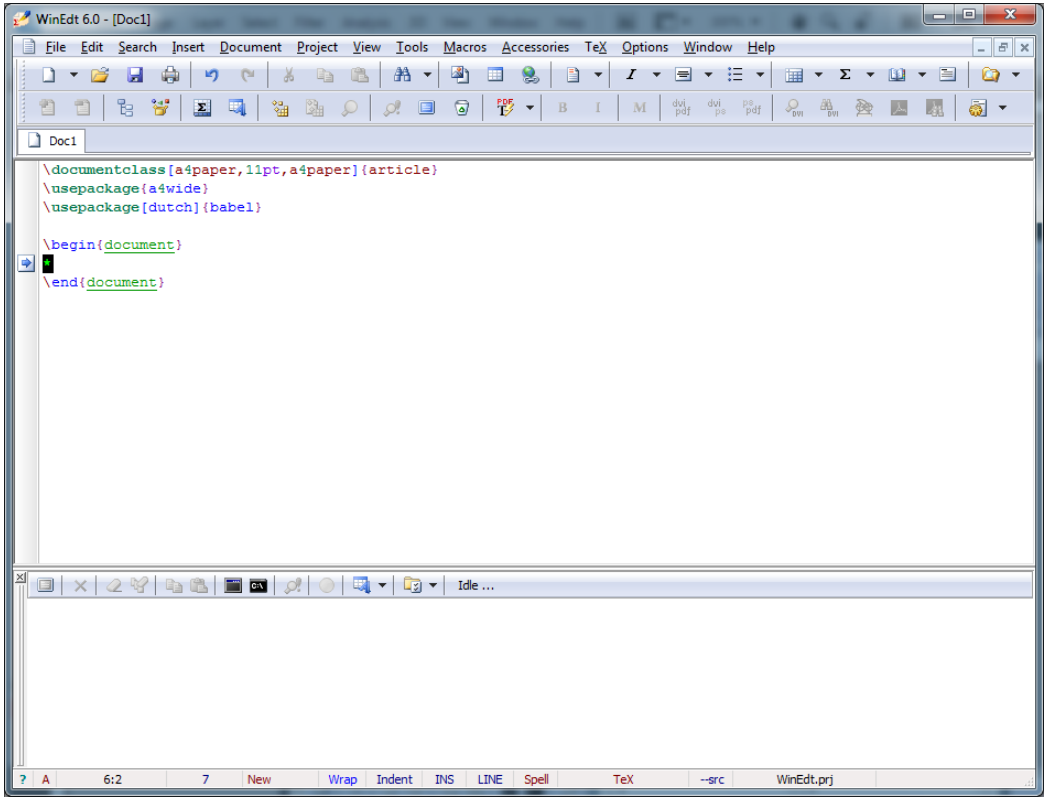
LaTeX Related Programs

WinEdt

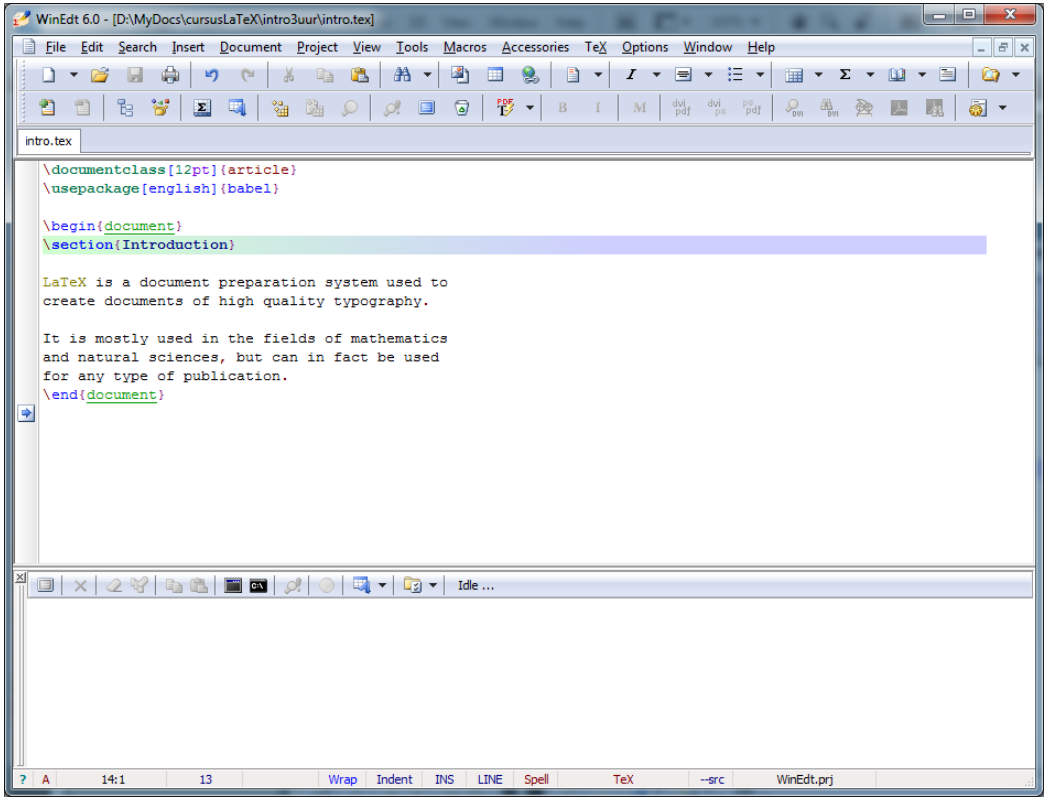


LaTeX Related Programs

WinEdt



WinEdt


WinEdt 6.0 - [D:\MyDocs\cursusLaTeX\intro3uur\intro.tex]
File Edit Search Insert Document Project View Tools Macros Accessories TeX Options Window Help
intro.tex
\documentclass[12pt]{article}
\usepackage[english]{babel}
\begin{document}
\section{Introduction}
LaTeX is a document preparation system used to
create documents of high quality typography.
It is mostly used in the fields of mathematics
and natural sciences, but can in fact be used
for any type of publication.
\end{document}
Idle ...
14:1
13
Wrap
Indent
INS
LINE
Spell
TeX
--src
WinEdt.prj

LaTeX Related Programs

runs LaTeX on the current document. If no errors are found, the resulting DVI file will be opened in Yap.

runs Yap on the generated DVI file.

converts DVI to PostScript.

opens the PostScript file in GSView.

runs PDFLaTeX on the current document.

opens the PDF document in Adobe Reader.

opens the document in Yap and jumpt to the current location in the document.

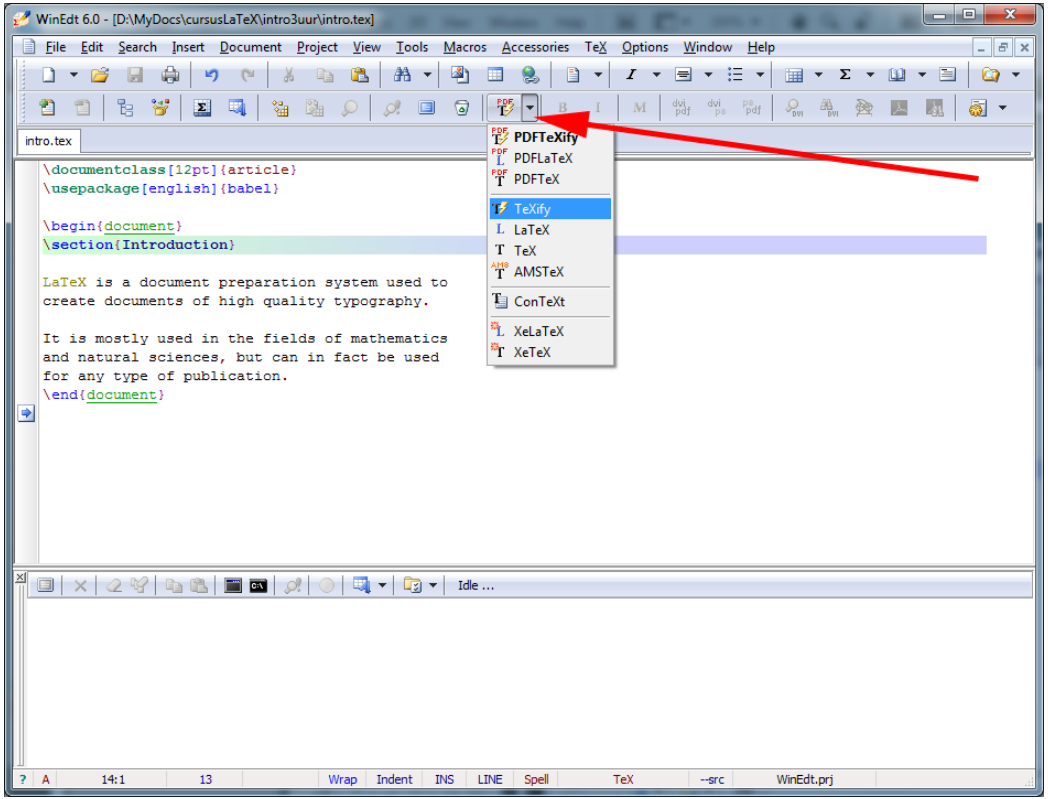
starts BibTeX(for bibliographies).

generates a master index.

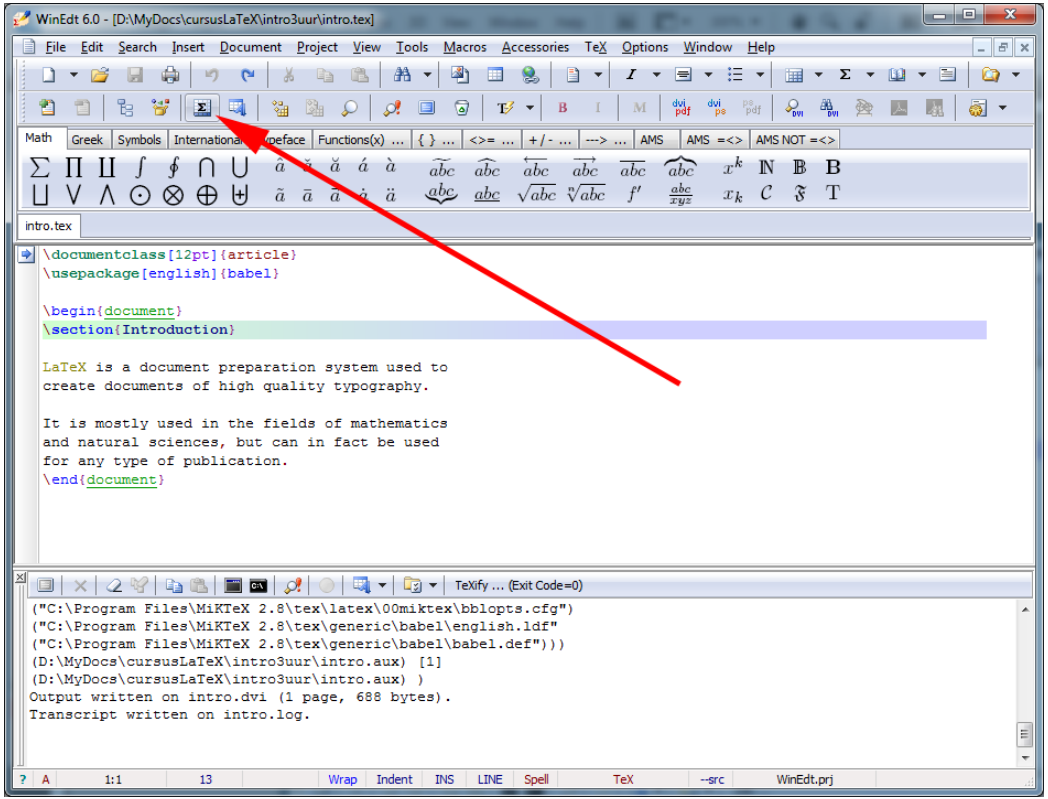
removes all generated auxiliary files (DVI, LOG, PostScript, BIB, . . .). Only the PDF file will not be deleted.

LaTeX Related Programs

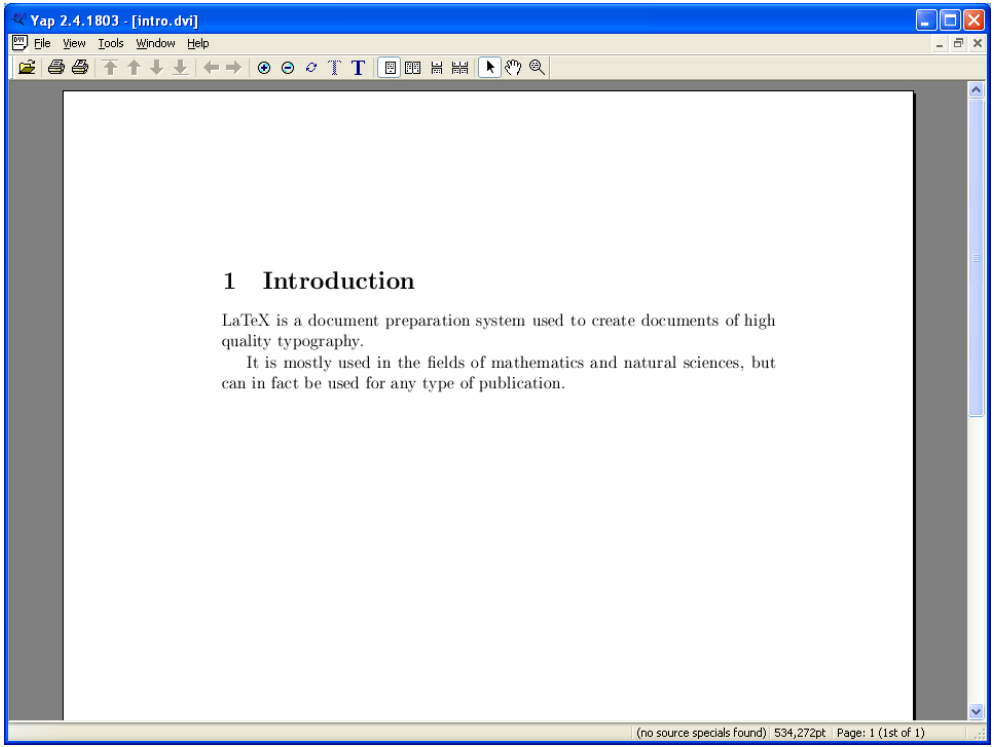
WinEdt



WinEdt

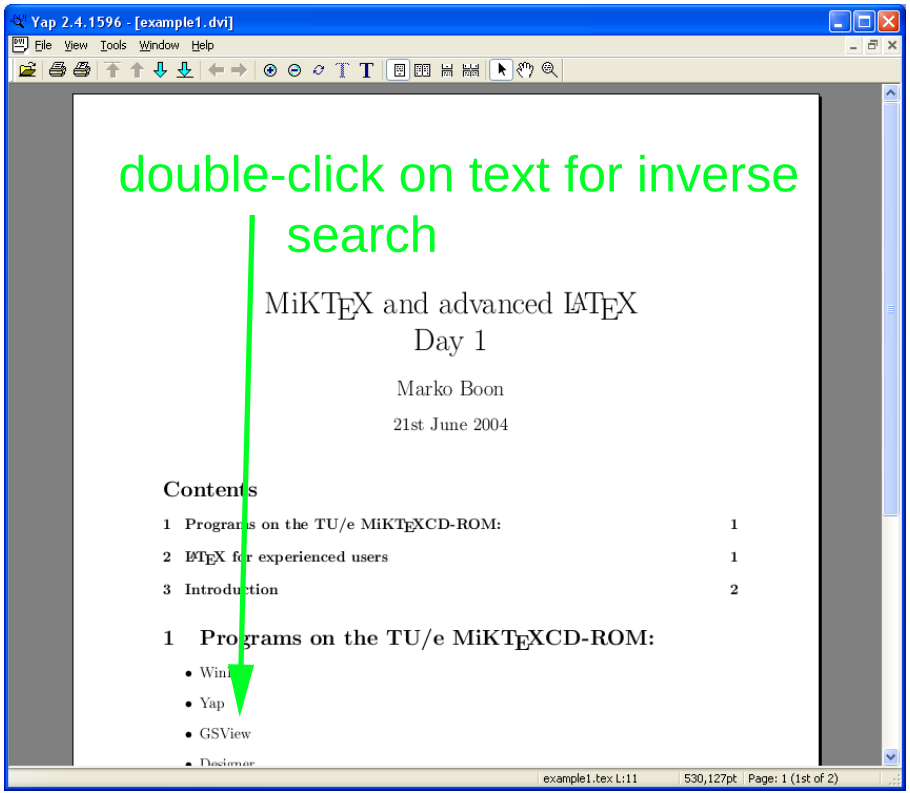

WinEdt 6.0 - [D:\MyDocs\cursusLaTeX\intro3uur\intro.tex]
File Edit Search Insert Document Project View Tools Macros Accessories TeX Options Window Help
Math Greek Symbols International ...peface Functions(x) ... { } ... <>= ... +/- ... ---> ... AMS AMS =<> AMS NOT =<>
intro.tex
\documentclass[12pt]{article}
\usepackage[english]{babel}
\begin{document}
\section{Introduction}
LaTeX is a document preparation system used to
create documents of high quality typography.
It is mostly used in the fields of mathematics
and natural sciences, but can in fact be used
for any type of publication.
\end{document}
TeXify ... (Exit Code=0)
("C:\Program Files\MiKTeX 2.8\tex\latex\00miktex\bblopts.cfg")
("C:\Program Files\MiKTeX 2.8\tex\generic\babel\english.ldf"
("C:\Program Files\MiKTeX 2.8\tex\generic\babel\babel.def")))
(D:\MyDocs\cursusLaTeX\intro3uur\intro.aux) [1]
(D:\MyDocs\cursusLaTeX\intro3uur\intro.aux))
Output written on intro.dvi (1 page, 688 bytes).
Transcript written on intro.log.
? A 1:1 13 Wrap Indent INS LINE Spell TeX --src WinEdt.prj

Yap


Yap 2.4.1803 - [intro.dvi]
File View Tools Window Help
1 Introduction
LaTeX is a document preparation system used to create documents of high quality typography.
It is mostly used in the fields of mathematics and natural sciences, but can in fact be used for any type of publication.
(no source specials found) 534,272pt Page: 1 (1st of 1)

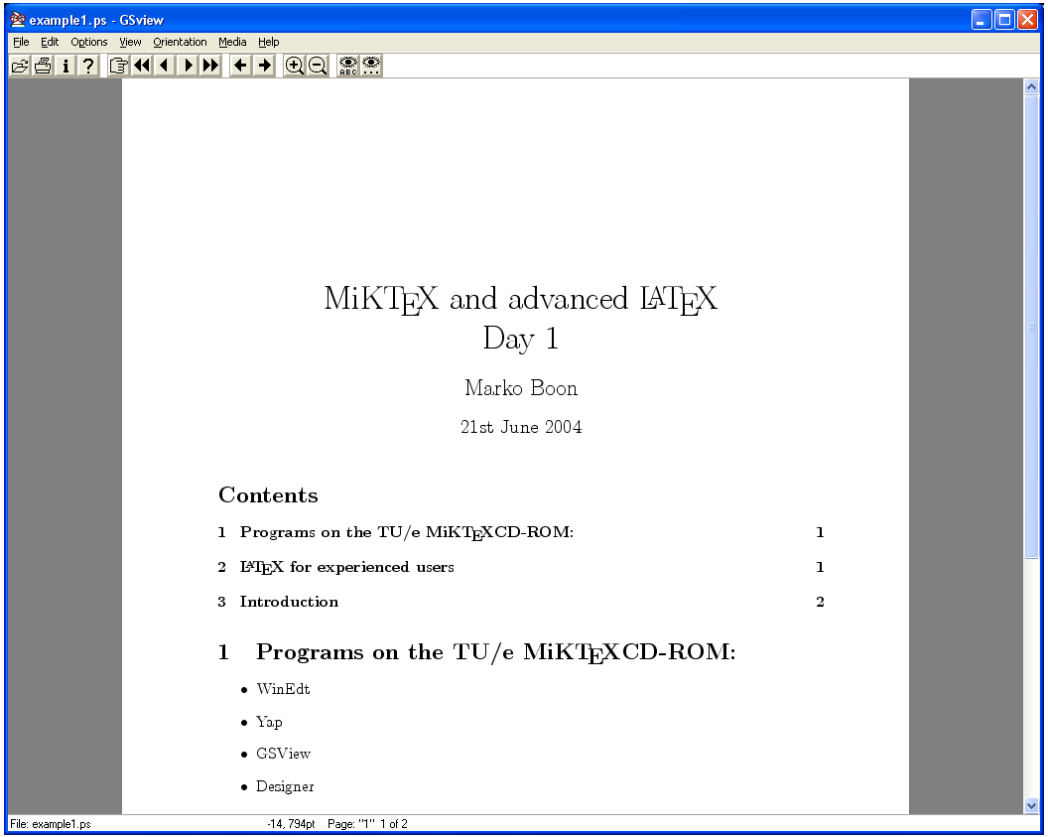
LaTeX Related Programs

Yap



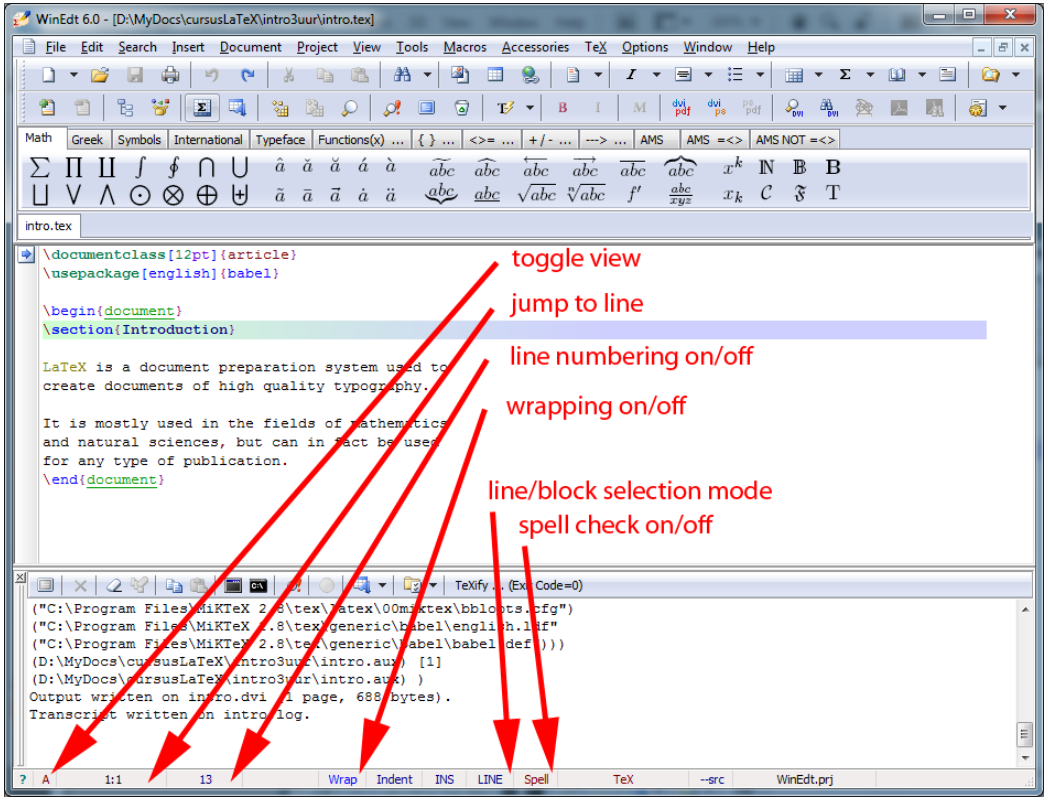
LaTeX Related Programs

GSView



LaTeX Related Programs

WinEdt



Text, Symbols and Commands

A *command* is an instruction to LaTeX to do something special. Three types of command names:

- the single characters `# $ & ~ _ ^ % { }` all have special meaning
- to print one of these characters, precede it with a backslash: `\$ \# \%`
- the backslash character `\` plus a sequence of letters, ending with the first non-letter: `\large \Large \bfseries`

Text, Symbols and Commands

A *command* is an instruction to LaTeX to do something special. Three types of command names:

- the single characters `# $ & ~ _ ^ % { }` all have special meaning
- to print one of these characters, precede it with a backslash: `\$ \# \%`
- the backslash character `\` plus a sequence of letters, ending with the first non-letter: `\large \Large \bfseries`

Some commands have a so-called *-form to modify their functionality somehow. Example:

```
\section*{Introduction}
```

Text, Symbols and Commands

Many commands operate on some piece of text, which then appears as an *argument* in curly braces following the command name. Examples:

```
\section{Introduction}

\textbf{bold text}

\begin{document}
```

Text, Symbols and Commands

Many commands operate on some piece of text, which then appears as an *argument* in curly braces following the command name. Examples:

```
\section{Introduction}

\textbf{bold text}

\begin{document}
```

Optional arguments are put into square brackets and mandatory arguments into curly brackets:

```
\documentclass[11pt]{article}

\usepackage[dutch]{babel}
```

Environments

An *environment* affects the text within it treating it differently according to the environment parameters.

```
This text will not appear centered.
\begin{center}
This text will appear centered.
This text will appear centered.
This text will appear centered.
\end{center}
This text will not appear centered.
```

Declarations

A *declaration* is a command that changes the values or meanings of certain parameters or commands without printing any text. The effect ends when another declaration of the same type is encountered.

```
This text appears normal while \bfseries this text
appears boldface.
```

Declarations

A *declaration* is a command that changes the values or meanings of certain parameters or commands without printing any text. The effect ends when another declaration of the same type is encountered.

```
This text appears normal while \bfseries this text
appears boldface.
```

When the declaration occurs within an environment or within a { } block, its scope extends only to until the end of this environment or block.

```
This text appears normal while {\bfseries this text
appears boldface}. This text is normal again.
\begin{center}
\bfseries
This text appears bold.
\end{center}
This text is normal again.
```

Loading Packages

A *package* is a set of LaTeX commands (or symbols, environments, declarations) stored in a file with the extension .sty. To invoke a package, simply call

```
\usepackage{package_name}
```

in the preamble!

Example: LaTeX does not have a command to include graphics, so if we want to include graphics in our document, we should load the package graphicx which defines a new command `\includegraphics`.

Text, Symbols and Commands

Special Characters – Spaces

LaTeX takes care of spacing in your document. The following two texts appear exactly the same in the DVI file:

```
\section{Introduction}

LaTeX is a document preparation system.
It is widely used in the fields of mathematics
and natural sciences, but also spreading to
many other disciplines.
```

```
\section{Introduction}LaTeX is                a
document preparation system. It        is
widely       used in the   fields
of
     mathematics  and natural
sciences,
but also spreading to many other disciplines.
```

Special Characters – Spaces

Some rules:

- one blank is the same as a thousand, only the first one counts.
- blanks at the beginning of an input line are ignored.
- blanks terminating a command name are removed.
- the end of a line is treated as a blank.

Special Characters – Spaces

Some rules:

- one blank is the same as a thousand, only the first one counts.
- blanks at the beginning of an input line are ignored.
- blanks terminating a command name are removed.
- the end of a line is treated as a blank.

To force a space to appear where it would otherwise be ignored: `\ `.
To force a new line: `\newline` or `\\`

Special Characters – Spaces

Spacing of any desired size may be inserted into the text with the commands

```
\hspace{10cm}
\hspace*{-3mm}
```

\hspace has no effect if it should come at the beginning of a line. The *-form will insert the spacing no matter where it occurs.

Special Characters – Spaces

Spacing of any desired size may be inserted into the text with the commands

```
\hspace{10cm}
\hspace*{-3mm}
```

\hspace has no effect if it should come at the beginning of a line. The *-form will insert the spacing no matter where it occurs.

Vertical spacing is created using the \vspace command:

```
\vspace{10cm}
\vspace*{-3mm}
```

Special Characters – Spaces

Spacing of any desired size may be inserted into the text with the commands

```
\hspace{10cm}
\hspace*{-3mm}
```

\hspace has no effect if it should come at the beginning of a line. The *-form will insert the spacing no matter where it occurs.

Vertical spacing is created using the \vspace command:

```
\vspace{10cm}
\vspace*{-3mm}
```

Further commands for increasing the spacing between paragraphs are:

\smallskip	\medskip	\bigskip

Lengths and units

Lengths consist of a decimal number, followed by a dimensional unit. Some units:

cm	centimeter
mm	millimeter
in	inch ($= 2.54$cm)
pt	point (1 in $= 72.27$ pt)
em	font-specific size: the width of the capital M
ex	font-specific size: the height of the letter x

Special Characters – Command Characters

As mentioned before, the characters # $ ~ _ ^ { } % are interpreted as commands.
To print them as text, give a command consisting of \ plus that character:

```
\#   \$   \~   \_   \^   \{   \}   \%
```

$ ~ _ ^ { } %

Special Characters – Command Characters

As mentioned before, the characters # $ ~ _ ^ { } % are interpreted as commands.
To print them as text, give a command consisting of \ plus that character:

```
\#   \$   \~   \_   \^   \{   \}   \%
```

$ ~ _ ^ { } %

To print a backslash, use the command \textbackslash: \

Special Characters – Accents

Diacritical marks or accents can be created with LaTeX:

```
\`e \'e \^o \"o \~o \=o \v{s} \c{c}
be\"invloeden
het re\"ele deel
Cura\c{c}ao
```

è é ô ö õ ō š ç
beïnvloeden
het reële deel
Curaçao

Special Characters

Special symbols can be entered directly, but only if the right input encoding is specified. The input encoding depends on the type and language of the operating system. We have to load the package inputenc to specify the correct encoding:

```
\usepackage[ansinew]{inputenc}
```

```
beïnvloeden, reëel, Curaçao
€ ƒ © ¥ §
```

beïnvloeden, reëel, Curaçao
€ ƒ © ¥ §

Special Characters

Special symbols can be entered directly, but only if the right input encoding is specified. The input encoding depends on the type and language of the operating system. We have to load the package inputenc to specify the correct encoding:

```
\usepackage[ansinew]{inputenc}
```

```
beïnvloeden, reëel, Curaçao
€ ƒ © ¥ §
```

beïnvloeden, reëel, Curaçao
€ ƒ © ¥ §

Please note that some of these characters also require the textcomp package.

The Euro Symbol: €

Adobe created a font containing euro symbols which also contains bold, italic and serif versions. To use these symbols, load the package europs. Now we can use the following commands:

The Euro Symbol: €

Adobe created a font containing euro symbols which also contains bold, italic and serif versions. To use these symbols, load the package europs. Now we can use the following commands:

\EURofc – creates the official Euro symbol: €

The Euro Symbol: €

Adobe created a font containing euro symbols which also contains bold, italic and serif versions. To use these symbols, load the package europs. Now we can use the following commands:

\EURofc – creates the official Euro symbol: €

\EUR – creates a Euro symbol depending on the current text style

The Euro Symbol: €

Adobe created a font containing euro symbols which also contains bold, italic and serif versions. To use these symbols, load the package europs. Now we can use the following commands:

\EURofc – creates the official Euro symbol: €

\EUR – creates a Euro symbol depending on the current text style

Bold: €
Italic: €
Sans-serif: €

The date

LaTeX contains a macro to print the current date: `\today`. The format of the date depends on the language set with the package babel.

`\today`
November 27, 2012

The date

LaTeX contains a macro to print the current date: `\today`. The format of the date depends on the language set with the package babel.

```
\today
```
November 27, 2012

```
\selectlanguage{dutch}\today
```
27 november 2012

Exercise

1. create a new LaTeX article. Font size: 11pt. Load the package a4wide to adjust the margins.
2. copy-paste the text from snowwhite.txt in the document body and run LaTeX. Explain what the error message means and fix the error.
3. right before the error, a Euro symbol occurs. Verify that the symbol is not printed in the DVI file. Make sure that LaTeX also prints the Euro symbol.
4. the last line (The End) should be large and centered.
5. create sections: Introduction, The evil stepmother, The great forest, The seven dwarfs, The murder of Snow White, The funeral, The prince, and The marriage.
6. create subsections: The cottage, The dwarfs, The encounter, First attempt, Second attempt, and Third attempt

Document Class

The first command in a .tex file determines the global processing format for the entire document:

```
\documentclass[options]{class}
```

Supported classes are book, report, article, letter or slides.

Supported options:

Document Class

The first command in a .tex file determines the global processing format for the entire document:

```
\documentclass[options]{class}
```

Supported classes are book, report, article, letter or slides.

Supported options:

- font sizes: `10pt` `11pt` `12pt`

Document Class

The first command in a .tex file determines the global processing format for the entire document:

```
\documentclass[options]{class}
```

Supported classes are book, report, article, letter or slides.

Supported options:

- font sizes: `10pt 11pt 12pt`
- paper size: `a4paper letterpaper`

Document Class

The first command in a .tex file determines the global processing format for the entire document:

```
\documentclass[options]{class}
```

Supported classes are book, report, article, letter or slides.

Supported options:

- font sizes: `10pt 11pt 12pt`
- paper size: `a4paper letterpaper`
- number of columns: `onecolumn twocolumn`

Document Class

The first command in a .tex file determines the global processing format for the entire document:

```
\documentclass[options]{class}
```

Supported classes are book, report, article, letter or slides.

Supported options:

- font sizes: `10pt 11pt 12pt`
- paper size: `a4paper letterpaper`
- number of columns: `onecolumn twocolumn`
- print style: `oneside twoside`

Loading packages

Packages are loaded in the preamble. A *package* is a set of LaTeX commands (or symbols, environments, declarations) stored in a file with the extension .sty. Important packages:

a4wide	uses smaller page margins, which means that more text fits on one page.
amsmath	contains advanced mathematical symbols.
babel	loads hyphenation rules for foreign languages.
europs	loads the Euro symbol: €.
fancyhdr	is used to customise headers and footers.
graphicx	defines a command to load external graphics.
hyperref	adds interactivity (hyperlinks, bookmarks) to your document.
listings	inserting source code with syntax highlighting

Paragraph Formatting

The following parameters affect the appearance of a paragraph:

\parskip the distance between paragraphs, usually in units of ex

\parindent the amount of indentation for the first line of a paragraph

Use the \setlength command to change the values of these parameters.

```
\setlength{\parskip}{1ex}
\setlength{\parindent}{0mm}
```

Paragraph Formatting

The following parameters affect the appearance of a paragraph:

\parskip the distance between paragraphs, usually in units of ex

\parindent the amount of indentation for the first line of a paragraph

Use the \setlength command to change the values of these parameters.

```
\setlength{\parskip}{1ex}
\setlength{\parindent}{0mm}
```

To suppress the indentation for one paragraph, or to force it:

```
\noindent
\indent
```

Parts of the Document – Title Page

```
\title{Title text}
\author{Author names and addresses}
\date{Date text}
\maketitle
```

Parts of the Document – Title Page

```
\title{Title text}
\author{Author names and addresses}
\date{Date text}
\maketitle
```

Use the \and command to define multiple authors:

```
\author{Jan Willem Knopper\\ jknopper@win.tue.nl \and
        Marko Boon\\ marko@win.tue.nl}
```

Document Layout

Parts of the Document – Title Page

```
\title{Title text}
\author{Author names and addresses}
\date{Date text}
\maketitle
```

Use the `\and` command to define multiple authors:

```
\author{Jan Willem Knopper\\ jknopper@win.tue.nl \and
        Marko Boon\\ marko@win.tue.nl}
```

Use `\date{}` to omit the date.

Parts of the Document – Abstract

The abstract is produced with the abstract environment:

```
\begin{abstract}
Text for the abstract.
\end{abstract}
```

In document class report the abstract appears on a separate page (without page number).
In document class article the abstract appears below the title.

Parts of the Document – Sections and chapters

The following commands produce automatic, sequential sectioning:

`\chapter{ }`	`\chapter*{ }`
`\section{ }`	`\section*{ }`
`\subsection{ }`	`\subsection*{ }`
`\subsubsection{ }`	`\subsubsection*{ }`

Parts of the Document – Sections and chapters

The following commands produce automatic, sequential sectioning:

`\chapter{ }`	`\chapter*{ }`
`\section{ }`	`\section*{ }`
`\subsection{ }`	`\subsection*{ }`
`\subsubsection{ }`	`\subsubsection*{ }`

Remarks:

- The command `\chapter` exists in document classes book and report only.
- A * behind the command results in the unnumbered version which will not be included in the table of contents.

Parts of the Document – Appendix

An appendix is introduced with the declaration `\appendix`

- Resets the section/chapter counter
- Changes the numbering form from numerals to capital letters (A, B, ...)
- Replaces the word “Chapter” by “Appendix”.

Please note that the actual word “Appendix” is **not** added to the table of contents!

Table of Contents

The table of contents is generated and printed with the command `\tableofcontents` (normally after title page and abstract).
All entries are created automatically, based on the sectioning commands. You have to run `latex` twice to get all references right!

Table of Contents

The table of contents is generated and printed with the command `\tableofcontents` (normally after title page and abstract).
All entries are created automatically, based on the sectioning commands. You have to run `latex` twice to get all references right!

TeXify and PDFTeXify

If you use the button for (PDF)TeXify instead of (PDF)LaTeX, WinEdt will run LaTeX, BibTeX, makeindex as many times as necessary.

 TeXify

 PDFTeXify

Labels and References

The command `\label{marker}` stores the current value of the relevant counter (section, chapter, equation, figure, table etc.) at that point in the text. To refer to a label, use:

\ref	to print the section, chapter, equation, figure or table number.
\pageref	to print the page number on which the `\label` command was issued.

Labels and References

The command \label{marker} stores the current value of the relevant counter (section, chapter, equation, figure, table etc.) at that point in the text. To refer to a label, use:

\ref to print the section, chapter, equation, figure or table number.

\pageref to print the page number on which the \label command was issued.

```
\section{Labels and References\label{labels}}

In section \ref{labels} you will find information
on how to create labels and references in \LaTeX.
The sections starts on page \pageref{labels}.
```

Changing Font Style

The following commands and declarations change the current font style:

Command	Declaration	Result
\emph	\em	*emphasised*
\textrm	\rmfamily	Roman font family
\texttt	\ttfamily	`Typewriter font family`
\textsf	\sffamily	Sans serif font family
\textup	\upshape	Normal, upright font shape
\textit	\itshape	*Italic font shape*
\textsl	\slshape	*Slanted font shape*
\textsc	\scshape	SMALL CAPS FONT SHAPE
\textbf	\bfseries	**Boldface font weight**
\textmd	\mdseries	normal (medium) font weight

Changing Font Style – Example

```
This is normal text with one \textit{italic} word.
{\sffamily This whole \itshape line is \textbf{sans}
serif.}

\textbf{\textit{Bold and italic}}

Do you see the difference?
\emph{emphasised}, \textit{italic}, \textsl{slanted}
```

Changing Font Style – Example

```
This is normal text with one \textit{italic} word.
{\sffamily This whole \itshape line is \textbf{sans}
serif.}

\textbf{\textit{Bold and italic}}

Do you see the difference?
\emph{emphasised}, \textit{italic}, \textsl{slanted}
```

This is normal text with one *italic* word. This whole *line is sans serif.*
Bold and italic
Do you see the difference? *emphasised*, *italic*, *slanted*

Changing Font Style – Example

```
This is normal text with one \textit{italic} word.
{\sffamily This whole \itshape line is \textbf{sans}
serif.}

\textbf{\textit{Bold and italic}}

Do you see the difference?
\emph{emphasised}, \textit{italic}, \textsl{slanted}
```

This is normal text with one *italic* word. This whole *line is sans serif.*
Bold and italic
Do you see the difference? *emphasised*, *italic*, *slanted*

This is an italic sentence containing an emphasised *word.*

Font Size

The font size can be changed using one of the following declarations:

Declaration	Result
\tiny	smallest
\scriptsize	very small
\footnotesize	smaller
\small	small
\normalsize	normal
\large	large
\Large	larger
\LARGE	even larger
\huge	still larger
\Huge	largest

Displaying Text

Lists

There are three environments available for producing formatted lists: itemize, enumerate and description.

```
\begin{itemize}
\item This is the first item
\item This is the second item
\item This is an item with a nested list:
\begin{itemize}
\item This list has different labels.
\item Another item.
\end{itemize}
\item the final item?
\item[+] it is even possible to change the label
\end{itemize}
```

Lists – Itemize

- This is the first item
- This is the second item
- This is an item with a nested list:
  - This list has different labels.
  - Another item.
- the final item?

+ it is even possible to change the label

Lists – Enumerate

```
\begin{enumerate}
\item This is the first item
\item This is another item \label{lab}
\item This is an item with a nested list:
\begin{enumerate}
\item This list has different labels.
\item In this item we refer to item \ref{lab}.
\end{enumerate}
\item the final item
\end{enumerate}
```

Lists – Enumerate

1. This is the first item
2. This is another item
3. This is an item with a nested list:
   (a) This list has different labels.
   (b) In this item we refer to item 2.
4. the final item

Exercise

1. add the a4paper option to the document class.
2. use the paragraph formatting commands to set the default paragraph indent to `0` and the default paragraph skip to `1ex`.
3. use the appropriate author and title commands to create the title.
4. create an abstract environment
5. add a table of contents after the abstract.
6. change the font size of the last line (The End) to Huge.
7. add a numbered list in section 4.2 (The Dwarfs)

Graphics Inclusion

To include an external graphics file:

Graphics Inclusion

To include an external graphics file:

- Load the package graphicx in the preamble:

```
\usepackage{graphicx}
```

Graphics Inclusion

To include an external graphics file:

- Load the package graphicx in the preamble:

```
\usepackage{graphicx}
```

- Include the graphics using this command:

```
\includegraphics[width=0.7\linewidth]{filename}
```

Graphics

Supported File Formats:

	EPS	PDF	JPG	GIF	PNG
LaTeX	yes	no	*yes**	no	*yes**
PDFLaTeX	no	yes	yes	no	yes

Please notice: only EPS and PDF are scalable. Use JPG and PNG just for photographs!

Many programs can generate EPS images. Use Corel Designer to export images created in other programs. Copy/Paste the objects in Corel Designer and export to EPS.

Use EPS2PDF (on your desktop) to convert EPS to PDF.

---

* does not work automatically when working with LaTeX. You should enter the coordinates of the bounding box manually.

Graphics


```
\includegraphics[options]{filename}
```

Graphics


```
\includegraphics[options]{filename}
```

When including EPS or PDF files, use the file name **without** extension!
LaTeX will take the EPS, PDFLaTeX will take the PDF.

Graphics


```
\includegraphics[options]{filename}
```

When including EPS or PDF files, use the file name **without** extension!
LaTeX will take the EPS, PDFLaTeX will take the PDF.

Supported options are:

scale=number	magnifies the figure by *number* over its natural size.
width=length	specifies the width to which the figure should be scaled
height=length	specifies the height to which the figure should be scaled
angle=number	rotates the figure counterclockwise over the specified angle (in degrees)
bb=llx lly urx ury	enters the coordinates of the bounding box manually.

Graphics

You can create a figure environment to create “floating” figures. LaTeX will put the image at the location that you specify, or on the top of the next page if the figure does not fit at the current page. In a figure environment you can add a caption and a label to refer to the figure.

```
\begin{figure}[ht]
\begin{center}
\includegraphics{normal}
\end{center}
\caption{Two dimensional normal distribution}
\label{fig:normal}
\end{figure}
```

Graphics

You can create a figure environment to create “floating” figures. LaTeX will put the image at the location that you specify, or on the top of the next page if the figure does not fit at the current page. In a figure environment you can add a caption and a label to refer to the figure.

```
\begin{figure}[ht]
\begin{center}
\includegraphics{normal}
\end{center}
\caption{Two dimensional normal distribution}
\label{fig:normal}
\end{figure}
```

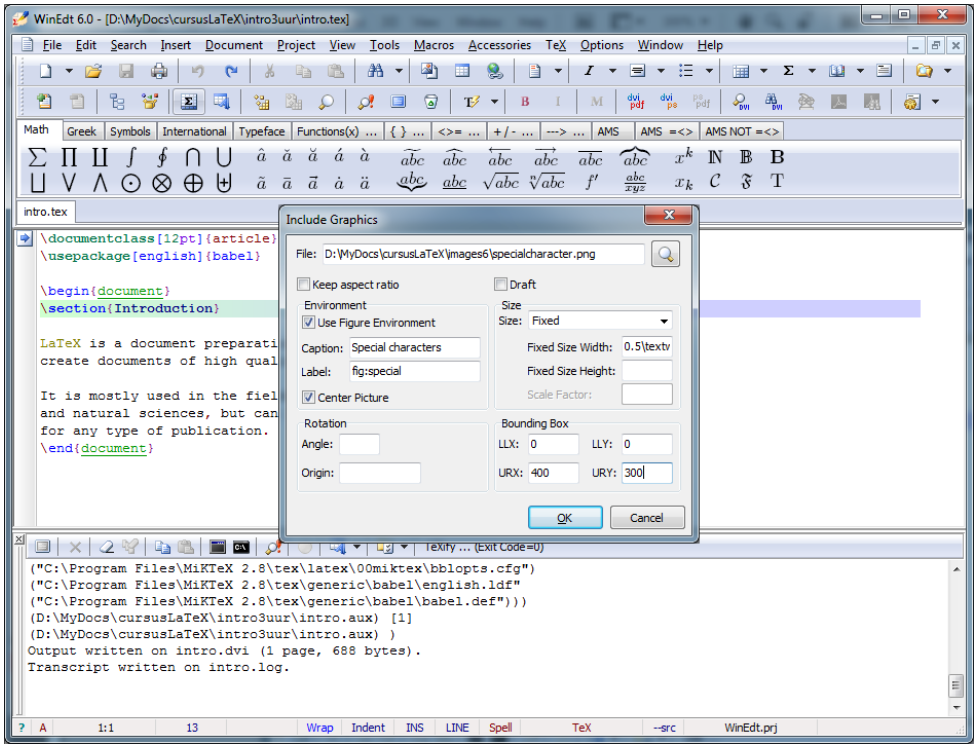
Now we can refer to the image:

```
See figure \ref{fig:normal}.
```

Graphics

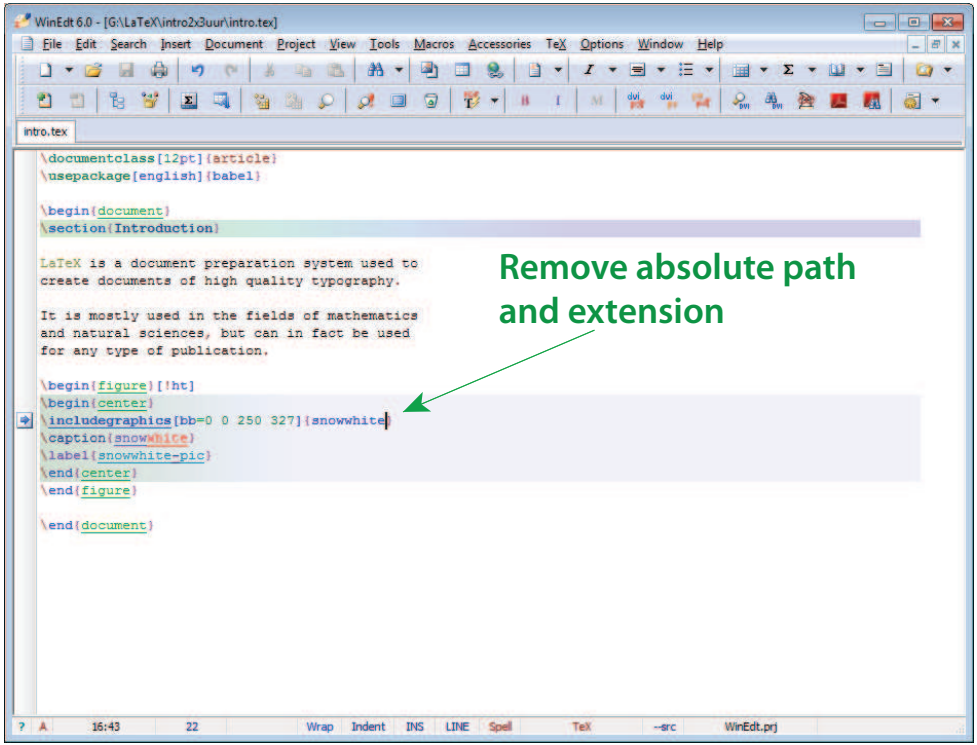
WinEdt has a useful plug-in to insert pictures.

It can be started by pressing the  -button in the top toolbar.

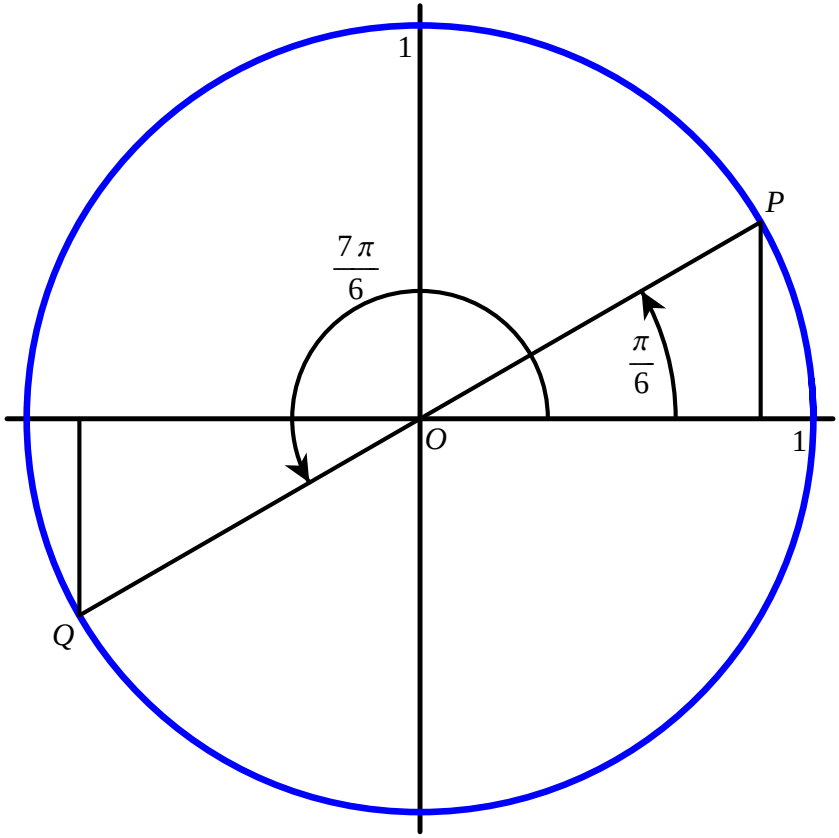


Graphics

WinEdt has a useful plug-in to insert pictures:

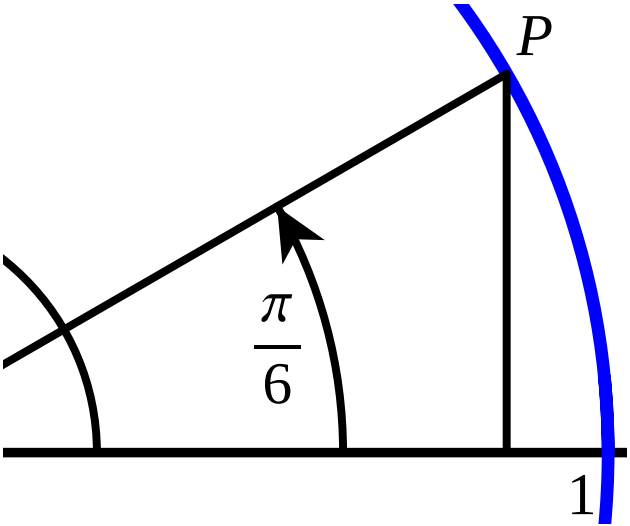


Scalable and non-scalable graphics

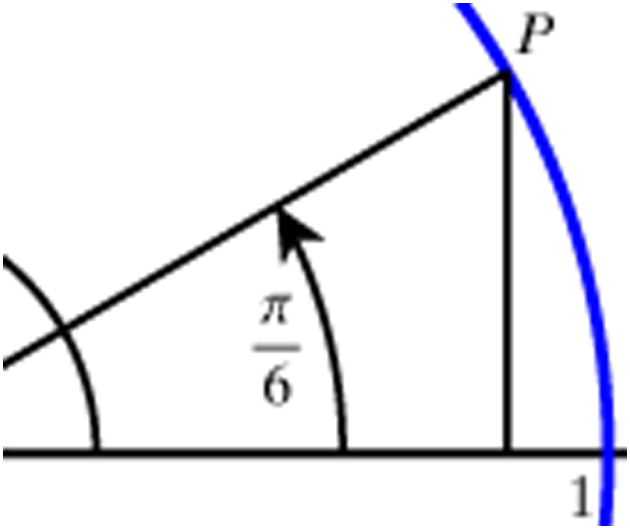


Graphics

Scalable and non-scalable graphics



Scalable graphics formats:
EPS, PDF, WMF, EMF, SVG.



Non-scalable graphics formats:
JPG, GIF, BMP, PNG.
But also: all scalable formats!

Graphics

Scalable Graphics Software

- Corel Designer,
- CorelDraw,
- Adobe Illustrator,
- Microsoft Visio,
- Microsoft Office Drawing,
- OpenOffice.org Draw,
- all computer algebra software (*Mathematica*, Matlab, Maple)

Non-scalable Graphics Software

- Adobe PhotoShop,
- Paint Shop Pro,
- MS Paint,
- all digital photo editing software!

Including JPG/PNG Images

LaTeX (unlike PDFLaTeX) cannot determine the bounding box automatically.

JPEG Image, 2304 × 1728 pixels, 72dpi, taken with 4.0 megapixel digital camera

Including JPG/PNG Images

LaTeX (unlike PDFLaTeX) cannot determine the bounding box automatically.

JPEG Image, 2304 × 1728 pixels, 72dpi, taken with 4.0 megapixel digital camera

```
\includegraphics[width=8cm,bb=0 0 2304 1728]
  {holiday.jpg}
```

Exercise

1. insert the picture snowwhite.jpg between the title and the abstract.