

# Introduction to Scientific Typesetting

## Lesson 2: Typing Text

Ryan Higginbottom

January 11, 2011

An Overview

Packages

---

Formatting and  
Positioning Text

---

Font Sizes

---

Units of the Document

---

Spaces

---

Boxes Around Text

---

**Packages**

**Formatting and Positioning Text**

**Font Sizes**

**Units of the Document**

**Spaces**

**Boxes Around Text**

An Overview

## Packages

Loading Packages

Packages in the Lab

Acquiring Packages

Elsewhere

The geometry

Package

A geometry example

The setspace

Package

Practice

Formatting and

Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

# Packages

# Loading Packages

An Overview

Packages

Loading Packages

Packages in the Lab

Acquiring Packages  
Elsewhere

The geometry  
Package

A geometry example

The setspace  
Package

Practice

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

“Packages” are add-ons to  $\text{\LaTeX}$  that can greatly extend its flexibility and your options. They are called in the preamble of your document with the command

$$\backslash\text{usepackage} [\textit{options}] \{\textit{packagename}\}$$

Most packages come with documentation, which tells you how to use the package properly. Find the relevant folder here:

```
C:\Program Files (x86)\MiKTeX 2.9\doc\latex
```

Let’s open the documentation for geometry (geometry.pdf) as an example.

# Packages in the Lab

An Overview

Packages

Loading Packages

**Packages in the Lab**

Acquiring Packages

Elsewhere

The geometry

Package

A geometry example

The setspace

Package

Practice

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

Believe it or not, *every single*  $\text{\LaTeX}$  package has been loaded onto your computer in this lab!

So, accessing  $\text{\LaTeX}$  packages in the lab should be easy.

It will be different on your own machine.

# Acquiring Packages Elsewhere

An Overview

Packages

---

Loading Packages

Packages in the Lab

Acquiring Packages  
Elsewhere

The geometry  
Package

A geometry example

The setspace  
Package

Practice

Formatting and  
Positioning Text

---

Font Sizes

---

Units of the Document

---

Spaces

---

Boxes Around Text

---

If you run MiKTeX, the easiest way to get a package onto your computer is to use the “Package Manager” program. (Go to the Start Menu, find the MiKTeX 2.9 folder, then Maintenance, then Package Manager.)

Once you find something there, MiKTeX will grab it and “refresh” your database.

Another way to get a package is to poke around on <http://www.ctan.org>. This is a central repository for  $\text{\LaTeX}$  packages.

Finally, if you call a package in your preamble and it isn’t on your computer,  $\text{\LaTeX}$  will either download and install it automatically, throw an error, or ask you if you’d like to download and install it. (It all depends upon the options you chose when installing MikTeX on your machine.)

# The geometry Package

An Overview

Packages

Loading Packages

Packages in the Lab

Acquiring Packages  
Elsewhere

The geometry  
Package

A geometry example

The setspace  
Package

Practice

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

The geometry package controls the margins and page layout for your documents. It is *very* useful.

A lot of the commands for the geometry package are straightforward. See all of them in the documentation.

These are all examples of options for the geometry package.

Example	Description
<code>margin=1in</code>	1-inch margins
<code>top=1.5in</code>	1.5-inch top margin
<code>height=8in</code>	8 vertical inches of text
<code>width=4in</code>	4 horizontal inches of text

The geometry package is good at arithmetic, so you won't have to specify everything.

# A geometry example

An Overview

Packages

Loading Packages

Packages in the Lab

Acquiring Packages  
Elsewhere

The geometry  
Package

A geometry example

The setspace  
Package

Practice

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

Let's create a quick file to test out this package.

```
-----  
\documentclass{article}  
\usepackage[margin=1in]{geometry}  
\begin{document}  
Hello, world!  
\end{document}  
-----
```

Now play around with the options above. Use `left`, `right`, `top`, `bottom`, or `margin`. Remember to include a unit of measurement!

# The setspace Package

An Overview

Packages

Loading Packages

Packages in the Lab

Acquiring Packages

Elsewhere

The geometry

Package

A geometry example

The setspace

Package

Practice

Formatting and

Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

You've probably noticed that  $\text{\LaTeX}$  typesets everything single-spaced by default. This is actually quite challenging to change without the aid of the `set space` package.

Put `\usepackage{set space}` in your preamble. Then use the `spacing` environment to change the spacing.

```
\begin{spacing}{2}
...
\end{spacing}
```

The mandatory argument is the spacing you desire. Most numerical values are fair game.

**Note on** `set space`: This package is *really* easy to use, so there isn't much documentation to speak of. It's commented out in the `.sty` file.

An Overview

Packages

Loading Packages

Packages in the Lab

Acquiring Packages

Elsewhere

The geometry

Package

A geometry example

The setspace

Package

**Practice**

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

## Let's practice!

Open up the first example PDF file from Sakai, and reproduce it.

An Overview

Packages

---

Formatting and  
Positioning Text

---

Text Decorations

Commands and  
Declarations

Text Decorations with  
`soul`

Changing colors for  
`soul`

Text Justification

Footnotes

Accents and Ellipses

Practice

Font Sizes

---

Units of the Document

---

Spaces

---

Boxes Around Text

---

# Formatting and Positioning Text

# Text Decorations

Short changes to the way your text looks (italics, bold, etc.) require *commands* in  $\text{\LaTeX}$ .

Command	Effect
<code>\textbf{This is bold}.</code>	<b>This is bold.</b>
<code>\textit{This is italic}.</code>	<i>This is italic.</i>
<code>\emph{This is emphasized}.</code>	<i>This is emphasized.</i>
<code>\underline{Underlined}.</code>	<u>Underlined.</u>
<code>\textsc{Small caps}.</code>	SMALL CAPS.
<code>\textrm{Roman}.</code>	Roman.
<code>\textsf{Sans Serif}.</code>	Sans Serif.
<code>\textsl{Slanted shape}.</code>	<i>Slanted shape.</i>
<code>\texttt{Typewriter type}.</code>	Typewriter type.

Commands like this can be nested.

- An Overview
- Packages
- Formatting and Positioning Text
- Text Decorations
- Commands and Declarations
- Text Decorations with soul
- Changing colors for soul
- Text Justification
- Footnotes
- Accents and Ellipses
- Practice
- Font Sizes
- Units of the Document
- Spaces
- Boxes Around Text

# Commands and Declarations

An Overview

Packages

Formatting and  
Positioning Text

Text Decorations

Commands and  
Declarations

Text Decorations with  
`soul`

Changing colors for  
`soul`

Text Justification

Footnotes

Accents and Ellipses

Practice

Font Sizes

Units of the Document

Spaces

Boxes Around Text

For text changes that stretch on for longer periods of time, use *declarations* instead of commands.

Instead of `\textbf{Bold text}` use  
`{\bfseries Bold text}`

or as an environment: `\begin{bfseries}`.

Here is the correspondence:

<b>Command</b>	<b>Declaration</b>	<b>Command</b>	<b>Declaration</b>
<code>\textrm</code>	<code>\rmfamily</code>	<code>\textsf</code>	<code>\sffamily</code>
<code>\texttt</code>	<code>\ttfamily</code>	<code>\textmd</code>	<code>\mdseries</code>
<code>\textbf</code>	<code>\bfseries</code>	<code>\textup</code>	<code>\upshape</code>
<code>\textit</code>	<code>\itshape</code>	<code>\textsl</code>	<code>\slshape</code>
<code>\textsc</code>	<code>\scshape</code>	<code>\emph</code>	<code>\em</code>

# Text Decorations with `soul`

An Overview

Packages

Formatting and  
Positioning Text

Text Decorations

Commands and  
Declarations

Text Decorations with  
`soul`

Changing colors for  
`soul`

Text Justification

Footnotes

Accents and Ellipses

Practice

Font Sizes

Units of the Document

Spaces

Boxes Around Text

You can do other cool things to text by loading the `soul` package.  
(You will need to load the `color` package too.)

■ strikethrough      `\st{...}`

Example: I went to the ~~doctor's~~ dentist's office this morning.

■ highlighting      `\hl{...}`

Example: Color can be used for **emphasis** instead of italics.

■ underlining      `\ul{...}`

Example: Underlining is not used very frequently for emphasis.

■ letterspacing      `\so{...}`

Example: Letterspacing makes some text look f u n n y.

# Changing colors for soul

An Overview

Packages

Formatting and  
Positioning Text

Text Decorations

Commands and  
Declarations

Text Decorations with  
soul

Changing colors for  
soul

Text Justification

Footnotes

Accents and Ellipses

Practice

Font Sizes

Units of the Document

Spaces

Boxes Around Text

You can change the strikethrough, highlighting, and underline colors.

```
\setstcolor{red}
```

Makes the ~~strikethrough~~ red.

```
\sethlcolor{green}
```

Makes the highlight color **green**.

```
\setulcolor{orange}
```

Makes the underline color orange.

# Text Justification

An Overview

Packages

Formatting and  
Positioning Text

Text Decorations

Commands and  
Declarations

Text Decorations with  
`soul`

Changing colors for  
`soul`

Text Justification

Footnotes

Accents and Ellipses

Practice

Font Sizes

Units of the Document

Spaces

Boxes Around Text

*Environments* are necessary to change the standard full justification that  $\text{\LaTeX}$  uses.

Environment	Effect
<code>center</code>	Centered text
<code>flushright</code>	Right justification
<code>flushleft</code>	Left justification

Example:

```
-----  
\documentclass{article}  
\begin{document}  
\begin{center}  
Hello, world!  
\end{center}  
\end{document}  
-----
```

An Overview

Packages

Formatting and  
Positioning Text

Text Decorations

Commands and  
Declarations

Text Decorations with  
`soul`

Changing colors for  
`soul`

Text Justification

Footnotes

Accents and Ellipses

Practice

Font Sizes

Units of the Document

Spaces

Boxes Around Text

$\text{\LaTeX}$  handles footnotes beautifully.

```
-----  
\documentclass{article}  
\begin{document}  
\underline{War and Peace} is a great  
book.\footnote{Actually, I haven't read it.}  
\end{document}  
-----
```

An Overview

Packages

Formatting and  
Positioning Text

Text Decorations

Commands and  
Declarations

Text Decorations with  
`soul`

Changing colors for  
`soul`

Text Justification

Footnotes

Accents and Ellipses

Practice

Font Sizes

Units of the Document

Spaces

Boxes Around Text

## Accents

If you need accented characters, like from a different language, you'll need some special commands.

Command	Effect
<code>\" {a}</code>	ä
<code>\' {e}</code>	é
<code>\^ {o}</code>	ô

Consult the guide posted on Sakai for a full list.

## Ellipses

type: It was the best of times, `\dots`  
see: It was the best of times, ...

An Overview

Packages

---

Formatting and  
Positioning Text

---

Text Decorations

Commands and  
Declarations

Text Decorations with  
`soul`

Changing colors for  
`soul`

Text Justification

Footnotes

Accents and Ellipses

Practice

Font Sizes

---

Units of the Document

---

Spaces

---

Boxes Around Text

---

## Let's practice!

Open up the second PDF file posted on Sakai and reproduce it.

An Overview

Packages

Formatting and  
Positioning Text

**Font Sizes**

The Size Option

Changing Sizes

Two Examples

Units of the Document

Spaces

Boxes Around Text

# Font Sizes

# The Size Option

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

The Size Option

Changing Sizes

Two Examples

Units of the Document

Spaces

Boxes Around Text

All of  $\text{\LaTeX}$ 's font size adjustments are set based upon the *normal* size for the document.

1. The default text size for a document is 10 point.
2. To make a change, use an optional argument on the `\documentclass` command.

```
\documentclass [12pt] {article}
```

3. Permitted font sizes for the `article` class: 9, 10, 11, 12.

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

The Size Option

Changing Sizes

Two Examples

Units of the Document

Spaces

Boxes Around Text

From smallest to largest:

Command
<code>\tiny</code>
<code>\scriptsize</code>
<code>\footnotesize</code>
<code>\small</code>
<code>\normalsize</code>
<code>\large</code>
<code>\Large</code>
<code>\LARGE</code>
<code>\huge</code>
<code>\Huge</code>

# Two Examples

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

The Size Option

Changing Sizes

Two Examples

Units of the Document

Spaces

Boxes Around Text

An example of changing font sizes within a document.

type: These `\large` words `\Large` get `\LARGE` big.  
see: These words **get big**.

type: Lots `\huge` of different  
`\small` sizes `\tiny` of `\normalsize` text.  
see: Lots **of different** sizes of text.

An Overview

Packages

---

Formatting and  
Positioning Text

---

Font Sizes

---

**Units of the Document**

---

Breaking Lines

An Example

Breaking Paragraphs

Breaking Pages

Practice

Spaces

---

Boxes Around Text

---

# Units of the Document

There are three ways to get a new line of text.

1. Insert a blank line in the `.tex` file. This produces a new *paragraph*.
2. Use the `\\` command. Text after the double backslash will appear on the next line, without any indentation. (Make sure you have a space in the `.tex` file after the `\\` to achieve this.)
3. Use `\newline`. This functions the same way as `\\`.

Most prefer `\\` because of this added functionality:

Command	Effect
<code>\\[length]</code>	creates interline space of <code>length</code>
<code>\\*</code>	prevents page break after line
<code>\\*[length]</code>	combination

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Breaking Lines

An Example

Breaking Paragraphs

Breaking Pages

Practice

Spaces

Boxes Around Text

Type:

I think we should continue  
this conversation `\\[.3in]` down here.

I think we should continue this conversation

down here.

Acceptable units of measurement: `in`, `cm`, `pt`

# Breaking Paragraphs

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Breaking Lines

An Example

Breaking Paragraphs

Breaking Pages

Practice

Spaces

Boxes Around Text

We've seen that a blank line in your file produces a new paragraph. Sometimes you might not want this paragraph indented. (It is indented by default.)

Use the command `\noindent` before the first line of the next paragraph.

```
-----  
\documentclass{article}  
\begin{document}  
This is indented.
```

```
\noindent This is not.  
\end{document}  
-----
```

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Breaking Lines

An Example

Breaking Paragraphs

**Breaking Pages**

Practice

Spaces

Boxes Around Text

This command is exactly what you might expect now:

```
\newpage.
```

Everything after this command will appear on the next page. That new paragraph will be indented.

An Overview

Packages

---

Formatting and  
Positioning Text

---

Font Sizes

---

Units of the Document

---

Breaking Lines

An Example

Breaking Paragraphs

Breaking Pages

**Practice**

Spaces

---

Boxes Around Text

---

## Let's practice!

Open up the third PDF file posted on Sakai and reproduce it.

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

**Spaces**

Horizontal Space

Vertical Space

Filling Space

Boxes Around Text

# Spaces

# Horizontal Space

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Horizontal Space

Vertical Space

Filling Space

Boxes Around Text

We've learned about `\_` and `\,`, as ways to add horizontal space.

We'll mainly use the `\hspace` command:

Command	Result
<code>\quad</code>	
<code>\qquad</code>	
<code>\hspace{8pt}</code>	
<code>\hspace{.6in}</code>	
<code>\hspace{1.2cm}</code>	

Horizontal space can be negative if needed.

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Horizontal Space

Vertical Space

Filling Space

Boxes Around Text

The analog to `\hspace` is `\vspace`.

$\text{\LaTeX}$  adds vertical space after the typeset line in which that command appears.

This means that `\vspace` commands should usually be on their own line in the `.tex` file.

- `\hfill` fills available space in the line with blank space
- `\vfill` fills available space on the page with blank space

```
-----  
\documentclass{article}  
\begin{document}  
\noindent The top. \\[.1in]  
The left \hfill and the right.  
\vfill  
The bottom.  
\end{document}  
-----
```

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Horizontal Space

Vertical Space

Filling Space

Boxes Around Text

An Overview

Packages

---

Formatting and  
Positioning Text

---

Font Sizes

---

Units of the Document

---

Spaces

---

**Boxes Around Text**

---

Framed Boxes

Paragraph Boxes

A Combination

Background Colors

Practice

# Boxes Around Text

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

Framed Boxes

Paragraph Boxes

A Combination

Background Colors

Practice

An added text effect can be achieved by framing text.

Like this!

```
\fbox{Like this!}
```

**Restriction:** You cannot break a line within `\fbox`.

# Paragraph Boxes

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

Framed Boxes

Paragraph Boxes

A Combination

Background Colors

Practice

The text within a paragraph box works just like a paragraph, except that the author sets the width.

```
\parbox{2in}{The text within...sets the width.}
```

You can accomplish the same thing with the `minipage` environment.

## Syntax

```
\parbox{width}{text}
```

```
\begin{minipage}{width}  
text
```

```
\end{minipage}
```

# A Combination

As you can see, we can combine these techniques together to form a boxed paragraph of any width we choose.

```
\fbox{\parbox{2in}{As you can...we choose.}}
```

As you can see, we can combine these techniques together to form a boxed paragraph of any width we choose.

```
\fbox{\begin{minipage}{2in}  
As you can ... we choose.  
\end{minipage}}
```

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

Framed Boxes

Paragraph Boxes

A Combination

Background Colors

Practice

# Background Colors

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

Framed Boxes

Paragraph Boxes

A Combination

Background Colors

Practice

We can also change the background color for a paragraph box:

Look at this yellow box. Look at  
this yellow box. Look at this yellow  
box. Look at this yellow box.

```
\colorbox{yellow}{\parbox{2in}{...}}
```

If you want this to happen within a document, you'll need to use a width of `\textwidth` within the `parbox`. (Alternatively, the `minipage` environment can be used in place of the `parbox`.)

An Overview

Packages

Formatting and  
Positioning Text

Font Sizes

Units of the Document

Spaces

Boxes Around Text

Framed Boxes

Paragraph Boxes

A Combination

Background Colors

Practice

## Let's practice!

Open up the fourth PDF file posted on Sakai and reproduce it.