

# ESA 2009 L<sup>A</sup>T<sub>E</sub>Xworkshop

Y. Lucero

## Contents

<b>1</b>	<b>Basic Typesetting</b>	<b>2</b>
<b>2</b>	<b>Mathematics</b>	<b>4</b>
<b>3</b>	<b>Figures</b>	<b>9</b>
<b>4</b>	<b>Tables</b>	<b>11</b>
<b>5</b>	<b>Advanced Typesetting</b>	<b>16</b>
5.1	Working with counters, references and labels . . . . .	16
5.2	Titles, Line Numbers, Headers and Footers . . . . .	17
5.3	Working with the EA style file . . . . .	19
5.4	Manipulating White Space . . . . .	20
<b>6</b>	<b>Bibliographies</b>	<b>22</b>
6.1	Bibliographies without BibTeX . . . . .	22
6.2	BibTeX . . . . .	22

# 1 Basic Typesetting

**Exercise 1.1:** Set up a basic input file modeled on the file shown in the slide. If you like, you can copy filler text from the file `LatexExercises.tex`.

**Exercise 1.2:** Use the options in `\documentclass[options]{class}` to change the font size of your document. Note that there are only templates types for three font sizes: 10pt, 11pt and 12 pt, where 10pt is the default.

**Exercise 1.3:** Add section headings using `\section{text}`

**Exercise 1.4:** Add subsection headings using `\subsection{text}`

**Exercise 1.5:** Make an unnumbered section heading, `\section*{text}`

**Exercise 1.6:** Make an unnumbered subsection heading, `\subsection*{text}`

**Exercise 1.7:** Latex has a built in run in heading. These are called paragraph headings. Add paragraph headings using `\paragraph{text} text text`

**Exercise 1.8:** Make a sub-paragraph headings `\subparagraph{text} text text`

**Exercise 1.9:** Change some text to bold `\textbf{text}`

**Exercise 1.10:** Emphasize some text (with italics) using `\emph{text}`

**Exercise 1.11:** Underline some text using `\underline{text}`

**Environments:** Many important things are handled with environments. For example, offset equations are created using the equation environment: typing

```
\begin{equation}
E=mc^2
\end{equation}
```

creates:

$$E = mc^2 \tag{1}$$

Environments are a powerful concept that are used in a variety of ways within latex. We will return to this concept many times. For now, try using the itemize environment in the next exercise.

**Exercise 1.12:** Create a list:

```
\begin{itemize}
\item green
\item blue
\item red
\end{itemize}
```

This should produce:

- green
- blue
- red

**Exercise 1.13:** Reproduce the list above as a numbered list using the enumerate environment

**Exercise 1.14:** Special characters. Some character in tex have special uses, like % is used for commenting lines out and \$ is used to offset mathematics within a paragraph. In order to actually use these characters in the text, you have to place an escape in front of them: \%. Add these characters to your document: %, \$, &, #, , , {.

**Exercise 1.15:** A quirk of latex is the lack of smart quotes. When you type quotes, you always get closed quotes. In order to get an open quote, you have to use the upper left key with the ~ on it: ‘‘`quoted`’’ yields “quoted”

**Exercise 1.16:** Foreign language: Use `\~n` to get ñ. Also try `\^a` and `\.x`

## 2 Mathematics

Make sure that you have the following packages in your preamble:

```
\usepackage{amssymb}
\usepackage{amsbsy}
\usepackage{amsmath}
```

**Exercise 2.1:** You can create math notation within a paragraph (called inline math) by surrounding the text with dollar symbols: `$c^2$`. The \$ symbols open and close math mode. Reproduce the following inline math text:

The pythagorean equation is sometimes also written as  $c^2 = a^2 + b^2$ .

Reproduce the equations in this section using some of the commands given.

**Exercise 2.2:** Follow the example given in Equation 1 to set up a basic equation using the equation environment: `\begin{equation}`...

$$S = cA^z \tag{2}$$

**Exercise 2.3:** use the function `\sqrt{}` to write this formula

$$c = \sqrt{a^2 + b^2} \tag{3}$$

**Exercise 2.4:** Create a ratio with the function `\frac{numerator}{denominator}`

$$I = \frac{aP}{1 + abP} \quad (4)$$

**Exercise 2.5:** use the function `\hat{}` to add a modifier to  $S$

$$\hat{S} = \frac{IP}{I + E} \quad (5)$$

**Exercise 2.6:** Create delimiters that can adjust their size: `\left( content \right)`

$$\frac{dN}{dt} = rN \left( 1 - \frac{N}{K} \right) \quad (6)$$

**Exercise 2.7:** Special characters: create the greek letters with `\epsilon`, `\mu` and `\sigma` and use `\sim` to create the tilde symbol

$$\epsilon \sim Norm(\mu, \sigma^2) \quad (7)$$

**Exercise 2.8:** Create superscripts with more than one character: `e^{-rx}` and create a sum with subscripts `\sum_x`

$$1 = \sum_x e^{-rx} l(x) b(x) \quad (8)$$

**Exercise 2.9:** Create a sum with multi-character subscripts and a superscript `\sum_{i=0}^k`

$$RSS = \sum_{i=1}^N (y_{pred} - y_{obs})^2 \quad (9)$$

**Exercise 2.10:** Special characters and spaces, use `\cdots` to create a ellipses and `\quad` to create a space. Also, use the greek `\beta`

$$y \sim \beta_0 + \beta_1 x_1 + \cdots \beta_i x_i + \epsilon \quad \epsilon \sim Norm(\mu, \sigma^2) \quad (10)$$

**Exercise 2.11:** Special characters: `\rightarrow` and `\infty` and the function `\lim_{x}`

$$e^x = \lim_{n \rightarrow \infty} \left(1 + \frac{x}{n}\right)^n \quad (11)$$

**Exercise 2.12:** Use `\mid` to create a vertical line and and escape character to get curly braces without an error: `\{ text \}`

$$Pr\{A \mid B\} = \frac{Pr\{A, B\}}{Pr\{B\}} \quad (12)$$

**Exercise 2.13:** Use `\textup{}` to get the plain text “good.”

$$Pr\{J_i(T) = A\} = Pr\{H(A) = \text{good}\} Pr\{S(T) > dist(A, B) \mid J_i(0) = B, t = T\} \quad (13)$$

**Exercise 2.14:** Make one equation number label go away by using `\nonumber` before you close the equation environment. Notice that the equation numbering skips over that equation.

**Exercise 2.15:** There are a few alternatives to the equation environment for creating equations in display mode. You can use the `displaymath` environment or you can use the square bracket notation:

$$\left[ E=mc^2 \right]$$

These options do not number the equations, only the equation environment does that.

**Exercise 2.16:** Change the equation numbering to be on the left side by adding the document class option “leqno” in the preamble:

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

**Exercise 2.17:** Also, try the document class option “fleqn” to have the equations flush left.

**Exercise 2.18:** The align environment is used for creating multi-line equations. The the \\ is used to break the lines and the & is used to indicate where to align the different lines.

```
%~~~~~
\begin{align}
f(x) = x^4 &+ 7x^3 + 2x^2 \nonumber \\
&+ 10x + 12 \\
\end{align}
%~~~~~
```

$$f(x) = x^4 + 7x^3 + 2x^2 + 10x + 12 \quad (14)$$

Here, I have used the command `\nonumber` on the first line, so the equation produced only numbers the second line. We could also use the environments `align*` to produce an unnumbered equation.

**Exercise 2.19:** Notice that in the equation environment the characters in the denominator are sometimes very small:

$$x = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + a_4}}} \quad (15)$$

Let’s subvert this by reformatting the characters. Use the command `\displaystyle{denominator}` in the denominators to produce:

$$x = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + a_4}}} \quad (16)$$

**Exercise 2.20:** The `amsmath` package gives us five environments for creating matrices: `pmatrix`, `bmatrix`, `Bmatrix`, `vmatrix` and `Vmatrix`. We use these environment within the `equation` environment. We use the command `\\` to break lines and `&` to separate columns. Use the code below to create a matrix, and then try out all five environments.

```
%~~~~~
\begin{equation}
A = \begin{pmatrix}
a_{11} & a_{12} & a_{13} \\
a_{21} & a_{22} & a_{23} \\
a_{31} & a_{32} & a_{33}
\end{pmatrix}
\end{equation}
%~~~~~
```

**Exercise 2.21:** The `cases` environment is used to create piecemeal equations. As before, we use the command `\\` to break lines and `&` to separate columns. For example:

```
%~~~~~
\begin{equation}
f(x)=\begin{cases}
0 & \text{for } x \leq 0 \\
1 & \text{for } x > 0
\end{cases}
\end{equation}
%~~~~~
```

The command `\text{text}` formats the characters as words rather than math. Within any math environment, the command `\;` creates a single blank space. Use the `cases` environment to reproduce Equation 17. You will require the symbol `\leq` for the lesser-than-or-equal-to symbol.

$$f(x, y) = \begin{cases} 0 & \text{for } x \leq 0, \\ \sin x + \phi & \text{for } x > 0. \end{cases} \quad (17)$$



### 3 Figures

In both Word and in Latex, figures and tables are usually created as floats. Floats are objects that cannot be broken up into parts and do not need to appear in the exact location where they are placed within the text. Each time we use the figure environment to create a figure, we create a float objects.

Make sure that you have `\usepackage{graphicx}` in your preamble.

**Exercise 3.1:** To include figures, we will use the command `\includegraphics[options]{filename}`. Make sure that you have a file named `fig1.pdf` in the same directory as your tex file, then use the following code to create a basic figure.

```
%~~~~~
\begin{figure}
\begin{center}
\includegraphics[scale=0.4]{fig1}
\caption{A very basic figure.}
\end{center}
\end{figure}
%~~~~~
```

By default, `\includegraphics{filename}` assumes that the file given is a pdf file. You can use files in many formats, including jpg, gif, png and ps. If you are using a file format other than pdf, then you must include the file suffix when you specify the file name.

**Exercise 3.2:** You may specify a longer pathname in the command `\includegraphics{pathname}`. Create subdirectory named `figs` in your current folder, and move `fig1.pdf` to that subdirectory. Then specify `\includegraphics{figs/fig1}`. You should also be able to move up a directory using the pathname `../fig1`

**Exercise 3.3:** In the above example, we used the option “scale” for `includegraphics`. Try these other options:

```
\includegraphics[width=5in, height=4in, angle=90]{filename}
```

**Exercise 3.4:** The figure environment has options for figure placement: h for here, t for top of page, b for bottom of page and p for place it on a floats only page. For example, try `\begin{figure}[t]`.

LaTeX tries to follow your placement directions, but it will only do so if the figure “fits.” I.e., if you specify that you want to place a five inch figure “here,” but the page is already mostly filled with text, LaTeX will override your specification and find another place to fit the figure. Therefore, you can specify several placement options at once, and LaTeX will follow them in the order that you specify. For example, the default figure option is often written: `[htbp]`. Note, LaTeX will try to find a place where the figure fits in both width and height. This means that if your figure is too wide for the page, it will not fit anywhere and LaTeX will place it at the end of your document. Furthermore because floats are placed in order, this will cause all of your remaining figures will be placed at the end of the document as well.

**Exercise 3.5:** Of course, in manuscript form you *want* to place the figures at the end of the document and on its own page. To do this, place your figures at the end of your document and use the command `\newpage` before each figure.

**Exercise 3.6:** Make sure you have the subfigure package loaded and recreate Figure ?? with the following text. Use the command `\\` to break between rows of figures.

```
%~~~~~
\begin{figure}[htbp]
  \begin{center}
    \subfigure[cap 1]{\label{fig.1a}\includegraphics[scale=0.20]{figs/fig1}}
    \subfigure[cap 2]{\label{fig.1b}\includegraphics[scale=0.20]{figs/fig1}} \\
    \subfigure[cap 3]{\label{fig.1c}\includegraphics[scale=0.10]{figs/fig1}}
    \subfigure[cap 4]{\label{fig.1d}\includegraphics[scale=0.10]{figs/fig1}}
    \subfigure[cap 5]{\label{fig.1d}\includegraphics[scale=0.10]{figs/fig1}}
  \end{center}
  \caption{Master caption.}
\end{figure}
%~~~~~
```

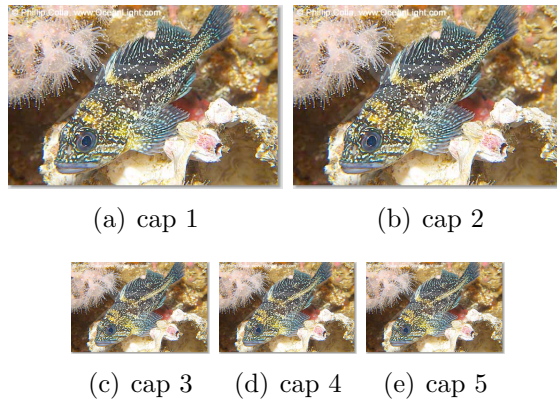


Figure 1: Master caption.

## 4 Tables

Table 1 is a very basic table produced by this code:

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\begin{table}[htdp]
\caption{default}
\begin{center}
\begin{tabular}{ c c c }
\hline
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9 \\
\hline
\end{tabular}
\end{center}
\end{table}
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Opening the table environment creates a float. Notice the same placement specifiers as used for the figure environment. The `\caption{}` command creates a caption. We open a centering environment so that the table is centered on the page. Opening the tabular environment starts the table itself. The tabular arguments are how the columns should be

aligned: c is for centered, r is for right aligned, and l is for left aligned. The command `\hline` creates a horizontal line. Rows are broken with `\\` and columns are broken with `&`.

Table 1: default

1	2	3
4	5	6
7	8	9

**Exercise 4.1:** Re-create Table 2. You will need to use the inline math mode for the greek letters, i.e. `$\gamma$`. Also, center the columns with symbols in them, but left align the columns with text.

Table 2: Some greek letters.

$\omega$	omega	$\Omega$	Omega
$\lambda$	lambda	$\Lambda$	Lambda
$\gamma$	gamma	$\Gamma$	Gamma

**Exercise 4.2:** The `booktabs` package creates more professional looking tables. Make sure you include the `booktabs` package and try replacing `\hline` with the `booktabs` commands `\toprule`, `\bottomrule`, and `\midrule`. Also, create a horizontal line that only stretches across some columns, i.e. for columns 1 through 2: `\cmidrule{1-2}`

**Exercise 4.3:** `\begin{tabular}{c l c l c}` Note the distinction between the lower case letter “l” and the vertical bar | (located on the `\` key). For left alignment, use the letter. To get vertical lines between columns, use the vertical bar between the column alignment arguments to `tabular`.

**Exercise 4.4:** The `multirow` package gives us two commands for grouping rows and columns: `\multicolumn{#columns}{orientation}{text}` and `\multirow{#rows}{width}{text}`. For

the width parameter, we can either specify a width (2in) or we can use `*` to get latex to choose the best fitting width. Add this heading to Table 2 using the `multirow` package and the `booktabs` package:

```
\toprule
\multicolumn{2}{c}{lowercase}&\multicolumn{2}{c}{uppercase} \\
\midrule
```

**Exercise 4.5:** Next, use the `multirow` command to group rows.

```
\multirow{3}{*}{familiar}& $\omega$ & omega & $\Omega$ & Omega \\
& $\lambda$ & lambda & $\Lambda$ & Lambda \\
& $\gamma$ & gamma & $\Gamma$ & Gamma \\
```

Don't forget to add the empty column on the other grouped rows that don't have the `multirow` command on them.

**Exercise 4.6:** Put it all together to reproduce Table 3

Table 3: Some harder greek letters.

	lowercase		uppercase	
familiar	$\omega$	omega	$\Omega$	Omega
	$\lambda$	lambda	$\Lambda$	Lambda
	$\gamma$	gamma	$\Gamma$	Gamma
unfamiliar	$\xi$	xi	$\Xi$	Xi
	$\upsilon$	upsilon	$\Upsilon$	Upsilon

**Exercise 4.7:** Frequently, it is desirable to produce tables where a column of numbers are aligned so that they center around a decimal point. This can be done using the specifier `@{text}` which inserts some text in every row. This command suppresses the inter-column space normally placed between columns. Try the following table:

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\begin{table}[h]
\begin{center}
\begin{tabular}{r@{.}l}
3    & 14159    \\
16   & 2         \\
123  & 456       \\
\end{tabular}
\end{center}
\end{table}
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

**Exercise 4.8:** By default, if the text in a column is too wide for the page, LaTeX won't automatically wrap it. Using `p{width}` as a specifier in the `tabular` environment, you can define a special type of column which will wrap-around the text as in a normal paragraph. For example, try re-creating a version of Table 4 using the following:

```
\begin{tabular}[h]{l | p{3in}}
```

**Exercise 4.9:** You can resize tables with `\resizebox{width}{height}{tabular object}`. Using the symbol `!` for the height preserves the original width/height ratio. Try resizing one of your tables with:

```

\resizebox{5in}{!}{
\begin{tabular}{...}
...
\end{tabular}
}

```

**Exercise 4.10:** You can also try `\scalebox{ratio}{tabular object}`

**Exercise 4.11:** Use the `landscape` environment to rotate your table to landscape orientation. This environment requires the `lscap` package, so be sure to add `lscap` in your

Table 4: Example of columns with wrapped text. Also, useful list of packages for making sophisticated tables.

Package	Description
hhline	Do whatever you want with horizontal lines
array	gives you more freedom on how to define columns
colortbl	make your table more colorful
supertab	for tables that need to stretch over several pages
longtable	same as above. Note: footnotes do not work properly in a normal tabular environment. If you replace it with a longtable environment, footnotes work properly
xtabular	Yet another package for tables that need to span many pages
tabulary	modified tabular* allowing width of columns set for equal heights
array	Does several things, including letting you create paragraph columns where text is bottom aligned or centered.

pre-amble with other packages. This package was not included in the original test.tex file and so may not be available offline to some users.

```
\begin{landscape}
\begin{table}
....
\end{table}
\end{landscape}
```

## 5 Advanced Typesetting

### 5.1 Working with counters, references and labels

**Counters** Latex uses counters to keep track of the numbering of sections, equations, tables, figures, etc. Here is a partial list of important counters that are automatically created: chapter, section, part, paragraph, page, equation, figure, table, footnote, section, subsection, subsubsection.

**Exercise 5.1:** Use the command `\setcounter{counter.name}{new value}` to change the current page number to page ten: `\setcounter{page}{10}`. Then change it back.

**Exercise 5.2:** Change the table numbering so that it skips table 1 by placing `\setcounter{table}{2}` before the first table float is created.

**Exercise 5.3:** You can simply advance a counter using `\addtocounter{counter}`

**Exercise 5.4:** You can see the value of a counter by placing `\the` in front of the counter name, e.g. `\thesection` will print the section number.

**Exercise 5.5:** Importantly, you can reformat how numbering appears using `\renewcommand{}{}`. To reformat how the counter appears, you can use any of the following: `\arabic{counter}`, `\alph{counter}`, `\Alph{counter}`, `\roman{counter}`, `\Roman{counter}`. For example place the following in your preamble:

```
\renewcommand{\thetable}{I-\arabic{table}}
```

**Labels and References** Any counter can be dynamically referenced. First, you use the command `\label{lab.text}` to marcate a section, table, figure or equation. Then later, you use the command `\ref{lab.text}` to reference it. For equations, the label should be placed within the equation environment. For sections, you only need to place the label after



the section has been created. For figures and tables, the label should be placed after the `\caption{text}` command, because this is when the counter is officially advanced.

It is a good idea to have a naming system for choosing labels. For example, I usually follow this convention:

- `tab.name` for tables
- `fig.name` for figures
- `eq.name` for equations
- `sec.name` for sections

where “name” is something short and memorable. The goal is to choose labels that are easy to remember so that you spend minimal time going back to look them up.

**Exercise 5.6:** Label and then references at least one of each: table, figure, equation and section. Note that for the references to work, you must compile twice. This is because LaTeX collects the labels on the first pass, and fills in the references on the second pass.

**Exercise 5.7:** Try out `\pageref{your.label.here}`. This will return the page number instead of the counter number.

**Exercise 5.8:** You can automatically change the section heading style for an appendix. Place the command `\appendix` towards the end of your document and then create some new section headings afterwards. Notice that you now have a new style of section numbering for your appendix.

## 5.2 Titles, Line Numbers, Headers and Footers

**Exercise 5.9:** You can automatically create a formal title by defining a few fields and then adding the command `\maketitle` where you want the title to show up (usually immediately below `\begin{document}`). To define the title page fields, add the following to your preamble:

```
\title{Best Article Ever}
\author{your name here!}
\date{December 31, 1999}
...
\begin{document}
\maketitle
```

**Exercise 5.10:** If you fail to define the date field, the latex will default to today's date. If you prefer no date, you can put `\date{}`

**Exercise 5.11:** You can add line numbers with `\usepackage{lineno}` and the following command inside your document: `\linenumbers`

**Exercise 5.12:** Insert a table of contents into your document by adding the command `\tableofcontents` after `\maketitle`. You will need to compile twice. Notice that this command creates a file with the `.toc` suffix. This is a simple text file, and if you open it you will see the latex code used to typeset your table of contents.

**Exercise 5.13:** In exactly the same way, you can add `\listoffigures` or `\listoftables`. Again, text files are produced with the suffixes `.lof` and `.lot` and you will need to compile twice. Latex will place the list where you place the commands. You may prefer to have these lists at the end of your document.

**Exercise 5.14:** These two commands draw on the captions to generate the lists. If you prefer a shorter title be used in the list, you can go back to the caption in the figure or table and use the options: `\caption[short caption for list]{real caption}`

**Exercise 5.15:** Notice that latex adds page numbers to your document by default. Try removing the page numbers by adding the command `\pagestyle{empty}` in your preamble.

**Exercise 5.16:** Suppose you want to put the page number as page x of y. By default, latex only keeps track of the current page and not the total number of pages. But you can

use the `lastpage` package to keep track of the total number of pages. Like with `\ref{}`, you will need to compile twice to make this work. Try adding this to your preamble:

```
\usepackage{lastpage}
\cfoot{\thepage\ of \pageref{LastPage}}
```

**Exercise 5.17:** You can also create customized headers and footers using the `fancy header` package. Add the following to your preamble

```
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhead[L]{Date updated: \today{}}
\fancyhead[R]{\LaTeX~Exercises}
```

There are a couple of new things going on in the example above. I am using the command `\today{}` to print out the current date in the heading. I am using the command `\LaTeX` (case sensitive) to get the name to print out in the official L<sup>A</sup>T<sub>E</sub>X way. And I am using the symbol `~` to preserve a space in between L<sup>A</sup>T<sub>E</sub>X and Exercises. Try removing the `~` and see that the words are pushed together.

**Exercise 5.18:** Add a footer using `\fancyfoot[selector]{text}`. The selectors can be L for left, R for right, or C for center. Also, try leaving the selector out altogether and see what happens.

### 5.3 Working with the EA style file

**Exercise 5.19:** Go get the `ecologyLatexClass.tar`. This is a style file and supporting materials for the journal *Ecological Applications* written by Todd Jobe, a UNC postdoc. Unzip these files and compile the example file.

**Exercise 5.20:** Next, put a copy of `ecology.cls` into the same directory as your `exercises.tex` file. Change the document class of your document from “article” to “ecology.” Compile this document, ignoring the errors that come up. Observe how this style file changes your document. Particularly note what it does with figures and tables.

**Exercise 5.21:** Fill out the empty fields of the title page (Running Head, etc.). Look to the preamble in the example file to learn how to do this.

**Exercise 5.22:** Use the environments for abstract and keywords to add an abstract and keywords to your document. Look to the example file for a demonstration of this.

## 5.4 Manipulating White Space

Latex recognizes a few very specific measurements, Table 5 lists the types of units that are available to you when using spacing commands.

Table 5: Measurement units used in LaTeX

pt	a point is 1/72.27 inch, that means about 0.0138 inch or 0.3515 mm.
bp	a big point is 1/72 inch, that means about 0.0139 inch or 0.3527 mm.
mm	a millimeter
cm	a centimeter
in	inch
ex	roughly the height of an 'x' in the current font
em	roughly the width of an 'M' (note the uppercase) of the current font

**Exercise 5.23:** Add a vertical space of three inches between two paragraphs: `\vspace{3in}`

**Exercise 5.24:** Add a small blank space inside the middle of a word: `\hspace{1em}`

Here is a list of several built in length parameters that control different parts of the page layout:

- `\baselineskip` The normal vertical distance between lines in a paragraph
- `\baselinestretch` Multiplies `\baselineskip`

- `\pagewidth` The width of the page
- `\pageheight` The height of the page
- `\parindent` The normal paragraph indentation
- `\parskip` The extra vertical space between paragraphs
- `\tabcolsep` The default separation between columns in a tabular environment
- `\textheight` The height of text on the page
- `\textwidth` The width of the text on the page
- `\topmargin` The size of the top margin (bottom margin is `pageheight-topmargin`)

**Exercise 5.25:** Choose a parameter and use the `set length` command to simply change the value: `\setlength{\parskip}{5ex}`. Notice that you can place this command anywhere in your document (including the preamble) and it will only change the parameter after the command is issued.

**Exercise 5.26:** Alternatively, make changes relative to the the default value:

```
\addtolength{\baselineskip}{1in}
```

**Exercise 5.27:** By default, latex will indent new paragraphs. I generally prefer that paragraphs not be indented but that there be spaces between paragraphs. To achieve my preferred formatting, I add this to the preamble:

```
\setlength{\parindent}{0in}  
\setlength{\parskip}{2ex}
```

**Exercise 5.28:** Use the `geometry` package options in the preamble to change the margin size of the document, e.g.:

```
\usepackage[left=1in,right=1in]{geometry}
```

## 6 Bibliographies

### 6.1 Bibliographies without BibTeX

The very simplest bibliographies are created with an environment called `thebibliography`.

**Exercise 6.1:** Try this code to generate a short reference list:

```
\begin{thebibliography}{}  
\bibitem{lamport.1994}  
  Leslie Lamport,  
  \emph{\LaTeX: A Document Preparation System}.  
  Addison Wesley, Massachusetts, 2nd Edition, 1994.  
\end{thebibliography}
```

Each bibliographic item is started with `\bibitem{citekey}`. Next you write out the reference. The line breaks above are for readability, latex will actually ignore them.

**Exercise 6.2:** if you would like to cite this reference, type `\cite{lamport.1994}`. As before, you will need to compile twice. By default, latex uses numbered citations. To change this we require the `bibtex` and the `natbib` package.

### 6.2 BibTeX

Make sure you have these files in the same directory as your tex file: `example.bib`, `ea.bst`

**Exercise 6.3:** Now we will use `bibtex` to create a reference list. There are four steps.

1. Cite the references in your document. Place this code in somewhere in your document

```
\nocite{ives.2003,kuparinen.2009,kuparinen.2009a,maechler.2005,zwillinger.1992}
```

2. Call the bibliography style file:

```
\bibliographystyle{ea}
```

The command `\bibliographystyle{}` can be placed anywhere in the document, including the preamble, so long as it appears before you issue the command `\bibliography{}`.

3. Call the bibliography data file

```
\bibliography{example}
```

Place the command where you want the reference list to appear and be sure the leave off the .bib suffix.

4. Compile. In TeXnic Center, the inclusion of bibtex happens in a single step along with the usual compile command. However in any unix-like system, including TeXshop on OSX, this is done in four steps:

1. compile LaTeX (shift-apple-L): gather cite keys
2. compile BibTeX (shift-apple-B): looks up cite keys in .bib file, generates .bbl file
3. compile LaTeX (shift-apple-L): inserts reference list
4. compile LaTeX (shift-apple-L): inserts citations where cite keys are

This should produce the following:

**Exercise 6.4:** Bibtex generates a .bbl file. This is also a simple text file. You can open it up and see how your reference list is generated.

**Exercise 6.5:** Let's cite some of these references within the text now. We would like to use author/year format and so we require the natbib package. Make sure that you have `\usepackage{natbib}` in your preamble. Now try: `\citep{maechler.2005}`. This should produce (?)

**Exercise 6.6:** Use the command `\bibpunct` to change the citation style:

```
\bibpunct{[ ]}{,}{a}{,}{,}
```

It has six mandatory arguments:

1. the opening bracket symbol, default = (
2. the closing bracket symbol, default = )
3. the punctuation between multiple citations, default = ;
4. the letter ‘n’ for numerical style, or ‘s’ for numerical superscript style, any other letter for author-year, default = author-year;
5. the punctuation that comes between the author names and the year
6. the punctuation that comes between years or numbers when common author names are suppressed (default = ,)

**Exercise 6.7:** Now, refer to Table 6 for the full list of citation commands in natbib. Try each one out.

Table 6: Natbib commands

Citation command	Output
<code>\citet{goossens93}</code>	Goossens et al. (1993)
<code>\citep{goossens93}</code>	(Goossens et al., 1993)
<code>\citet*{goossens93}</code>	Goossens, Mittlebach, and Samarin (1993)
<code>\citep*{goossens93}</code>	(Goossens, Mittlebach, and Samarin, 1993)
<code>\citeauthor{goossens93}</code>	Goossens et al.
<code>\citeauthor*{goossens93}</code>	Goossens, Mittlebach, and Samarin
<code>\citeyear{goossens93}</code>	1993
<code>\citeyearpar{goossens93}</code>	(1993)



**Exercise 6.8:** We provided several bibliography style files: `ecology.bst`, `science.bst`, `nature.bst` and `plos.bst`. Make sure these files are in your directory and sexperiment with the various style files. For example,

```
\bibliographystyle{science}
```