# Math 291: Lecture 9

Justin A. James

Minnesota State University Moorhead
web.mnstate.edu/jamesju
jamesju@mnstate.edu

March 27, 2017

# Outline

# Adding Sectioning to a Document

LaTeXhas several commands built in that help to organize longer documents into different sections. The most common of these commands are as follows:

- `\part{name of part}`
- `\chapter[shorter name]{full name of chapter}`
- `\section{name of section}`
- `\subsection{name of subsection}`
- `\subsubsection{name of subsubsection}`
- `\paragraph{paragraph heading}`
- `\subparagraph{subparagraph heading}`

# Adding Sectioning to a Document

**Notes:**

- Parts, chapters, section, subsections, and subsubsections are numbered within the document.
- Paragraphs and subparagraphs are not numbered, but, like the other categories, they are given a bold heading. Smaller categories get smaller heading fonts.
- All these commands are built into the article document class, with the exception of the chapter command, which is used in the report document class.

# Example:

- Go to my website http://web.mnstate.edu/jamesju/Spr2017/Content/M291-Week9.html
- Download ExWk9Lecture.tex.
- Download the nine images as well.
- Download IncludeExWk9.tex.
- Download ExampleBibLecture.bib as well. Be sure to save this as a .bib file.
- Put these all in the same folder.
- Open ExWk9Lecture.tex and build your document and open the .pdf file.

- The first Chapter on Gryffindor is sectioned appropriately.
- Add sectioning the Chapter on Slytherin in a similar way. You will need:
  - A chapter called Slytherin.
  - A section called The House.
  - A subsection called The People
  - A subsubsection called Tom Marvolo Riddle
  - A subsubsection called Draco Malfoy
  - A subsubsection called Severus Snape
  - A subsection called Some Spells

# Outline

# Placing Labeled Figures in a Document:

- Another nice feature that is built into LaTeX is the ability to include figures in the body of a document.
- We already learned how to include image files of various types into a document.
- Designating an image file as a figure allows the LaTeXcompiler some flexibility in where the figure is placed within the final compiled page.
- It also allows us to give the figure a label and to include it in list of figures at the beginning of the document.
- Here is the syntax for inputting a figure:

```
\begin{figure}[location: h, t, b, or p]
\begin{center}
\includegraphics[sizing command]{name of image file}
\end{center}
\caption{whatever caption you want underneath the figure}
\label{fig:<reference name for figure>}
\end{figure}
```

# Placing Labeled Figures in a Document:

**Notes:**

- The location designations are as follows:
  - *h* ("here"): place the figure as close to the current location as possible.
  - *t* ("top"): place the figure toward the top of the page.
  - *b* ("bottom"): place the figure toward the bottom of the page.
  - *p* ("page"): place the figure on its own page.
- The caption can also be placed above the figure. Just move the caption commands so it is before the centering command.
- A figure can also be placed flushleft or flushright on a page, although this often ends up looking a bit silly.

# Example:

- You'll notice that the image of Harry's Patronus is missing.
- Just before the chapter on Slytherin add the following text. Be sure to use the package graphicx.

```
\begin{figure}[h]
\caption{Harry's Patronus is a Stag}
\begin{center}
\includegraphics[scale=0.5]{HarryPatronus.eps}
\end{center}

\label{fig:Patronus}
\end{figure}
```

- Now, prior to the figure add the text

```
The Patronus Spell is on display in Figure \ref{fig:Patronus}.
```

# Outline

# Creating a Title Page:

- Now that we have some content in place, let's go back and create a title page, a table on contents, and a table of figures.
- To create a Title Page do the following:
  - Use the "title" command to add a title for your document.

    ```
    \title{title text}
    ```

  - Use the "author" command to add the author information.

    ```
    \author{name (or add \\ name 2, etc)}
    ```

  - Use the "date" command to add a date other than the current date (If this step is skipped, the current date will be supplied).

    ```
    \date{enter desired date}
    ```

  - Then type the following command at the very beginning of your document:

    ```
    \maketitle
    ```

# Creating a Table of Contents and a List of Figures:

- To create a table of contents, just type the following command at the beginning of your document:

```
\tableofcontents
```

- To create a list of figures in the document, just type the following command at the beginning of your document:

```
\listoffigures
```

# Outline

# Including and Inputting Files

- Sometimes, when we are creating a very large documents, you may want to create the final document by piecing several smaller documents together.
- LaTeX has nice commands for doing this:
- You can use either the "include" command or the "input" command.
  - To use these commands, we first create a separate .tex file that contains all of the material that we want to include as a portion of the larger document
  - This file has *only* the material we want to include (no beginning or preamble or packages. Not even begin and end document commands.)
  - When this material is included, it will be read by the compiler as if it was actually typed into the main document

# Example:

- Add the contents of this file to our running example by typing:

  ```
  \input{IncludeExWk9.tex}
  ```

  or

  ```
  \include{IncludeExWk9.tex}
  ```

- Compile the document and take a look to see how the new material is incorporated. You should compile **twice** in order to update the table of contents to include this new material.

# Outline

# Creating Labels

The syntax for labeling a numbered object is: `\label{NameOfLabel}`

- Labeling can be done anywhere within the body of the environment you are labeling.
- Labels for objects at lower (or higher) "levels" must be placed at the appropriate "level".
- You are **Strongly** encouraged to put the labels **immediately** after the command that makes the object you are labeling.
- The name you give you label can be whatever you want it to be.
- It is common, especially in long documents, to use something like: `\label{Fig:NameOfFigure}` or `\label{Chap:NameOfChapter}`.
- Using this syntax, the names are just easier to remember (but longer to type).
- The name you use **never** appears in your final document.

# Practicing With Labels

- Using the document we have created, go through the document and add labels to:
  - A Chapter
  - A Section
  - A Subsection
  - A Theorem
  - A Figure
  - An enumerated item

# Outline

# Referencing Labels

- When we reference a label, the compiler will insert the number associated with the object we previously labeled in the place where we type the reference command.
- Note that **only** the number will be inserted. We will usually add our own accompanying text to smooth out the phrasing of the reference. For example, we may want to use phrases like:
  - "In Chapter 1,..."
  - "...back in Section 4.2, ..."
  - "...as we see in Figure 4.3..."
  - "...using Theorem 1.2 ..."
  - "...as in part (b) above ..."

# Referencing Labels

- Specific labels are referenced using the command: `\ref{NameOfLabel}`.
- Note that you will need to run LaTeX **twice** in order for the labels to appear correctly.
- LaTeX will show *warnings* (**not** errors) when the labeling doesn't work.
- For this reason, when we compile, we may need to read the log file a bit more carefully when we are using labels and references.
- If a reference is bad, ?? will appear in place of the label in your final document.

# Referencing Labels

- Put references to each of the labels you previously defined in your sample document.
- Add additional text as needed to make the references read reasonably well.
- Notes:
  - References to a subsection (or subsubsection) will have the form "2.3.2"
  - References don't seem to interact well with "parts" (you may not get the part number as part of your reference).
  - When referencing things in an enumerate environment, you don't get the chapter, section, etc., numbers. Only the part within the enumerate environment.

# Outline

# Building Bibliographies

- LaTeX has a related program called BibTeX that will automatically build bibliographies, including their references within a document.
- BibTeX requires both a *separate* file with a .bib extension and using the BibTeX build command as well as LaTeX .
- The BibTeX entries look somewhat complicated, but mathematicians, being who we are, have worked to make life easier...

# Creating the .bib file

One nice way to get .bib entries in the correct format is to obtain them from MathSciNet:
MathSciNet is a fairly comprehensive reference archive for articles in mathematical journals. (Use it on campus – it is not free!)

(1) Go to MathSciNet:
http://www.ams.org/mathscinet/search.html

(2) Do a **publications** search by typing in the name of your favorite author (Fulghesu).

(3) Check the first box and a second box from an article published in 2010.

(4) In the pull-down menu, change it to: Citations (BibTeX).

(5) Choose: Retrieve Marked.

(6) Open ExampleBibLecture.bib

(7) Use Copy and paste to add these into the .bib file.

# Creating .bib entries

There are several different types of bibliographic entries. Standard ones are:

- article (requires: author, title, journal, year)
- book (requires: author or editor, title, publisher, year)
- inbook (requires: author or editor, title, chapter and/or pages, publisher, year)
- misc (optional: author, title, howpublished, month, year, note, key (for alphabetizing))

For a fuller list, and other detailed information, see:
http://bibliographic.openoffice.org/bibtex-defs.html

# Hints for the .bib file

**The Citation Key**

- The Key comes after the typed entry. (@ARTICLE{CitationKey})
- MathSciNet gives each article a unique key, but it is not very handy reference.
- Typically, we will change this to a tag that is easier to remember and reference. For example, Fulghesu2012 and Fulghesu2010.

Any field in a .bib reference that is neither required nor optional is ignored - so you may include whatever you want in your .bib entries.

# Using the .bib file

- To refer to one of our bibliographic references in a document, we use the command:
- `\cite{CitationKey}`
- Practice by adding a citation referencing one of the papers in the .bib file.

To generate a bibliography, at the very end of your .tex file (just prior to the `\end{document}`), add the following commands:

`\bibliographystyle{plain}`

`\bibliography{NameOfBibFile}` (Do not include the .bib extension in the file name.)

# Running BibTeX

- You will need to compile your main document four times:
  1. Run LaTeX (regular build)
  2. Run BibTeX (from the pull-down menus: Build - Current File - BibTeX)
  3. Run LaTeX
  4. Run LaTeX

# Running BibTeX, continued

- Note that in the bibliography in your final document, only those papers actually cited appear.
- Your .bib file can contain (and often does) any paper you've ever referenced. You just continue adding references as needed.
- Try bibliographystyles of plain, alpha, etc.
- More bibliographystyles, and examples of how they format the various types of documents, can be found on the web.
- You might also want to investigate the commands `\begin{thebibliography}` and `\bibitem` for use without a .bib file.