Writing Master's and Doctoral Theses Using LaTeX

Global Program in Economics & Management (GPEM)
Tohoku University, Japan*

March 2020

Introduction

This note will provide an example of how to use LaTeXfor Master's and Doctoral theses for the Global Program in Economics & Management (GPEM), Tohoku University.[1] As there is no standard template for theses in our department, we have prepared this note in order to provide a good example for students who are writing theses.[2] Before discussing about using a template, this note provides a brief explanation about how to use LaTeX, because students are probably new to this software.

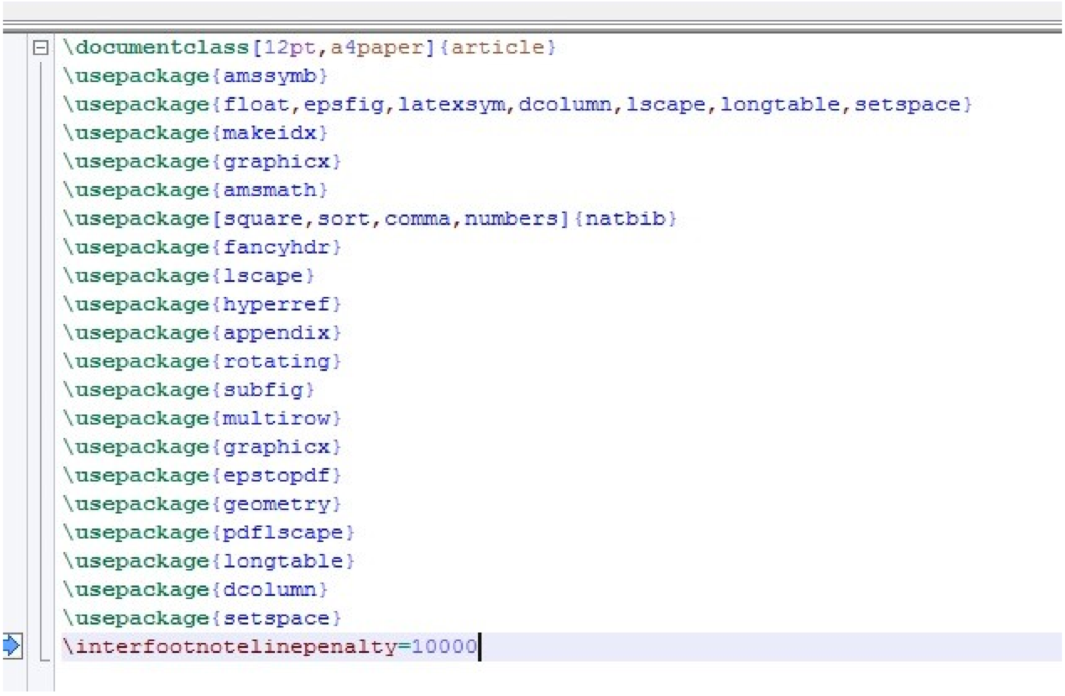
Installing LaTeX

In order to install LaTeX, you must first download MikeTex. It is best to install the complete package (which can be quite large∼1.5GB). After you have installed it from the setup file, update all of the packages. You will also need an editor. The most used ones are Texmaker and TeXnicCenter. However, you can also use a commercial version if you desire, such as WinEdt.

[*] This document has been prepared by a team of PhD students (Kirin Moonen, Gabriel Ivan Fuentes Cordoba, Koji Murayama and Lamia Bazzaoui) led by Jun Nagayasu, Director of GPEM.

[1] Master's theses are equivalent to Project Reports here. A Microsoft Word template is also available from the GPEM's homepage.

[2] This is an example of a thesis, and is not a standard. Therefore, you do not need to follow this format if it is not according to your preference.



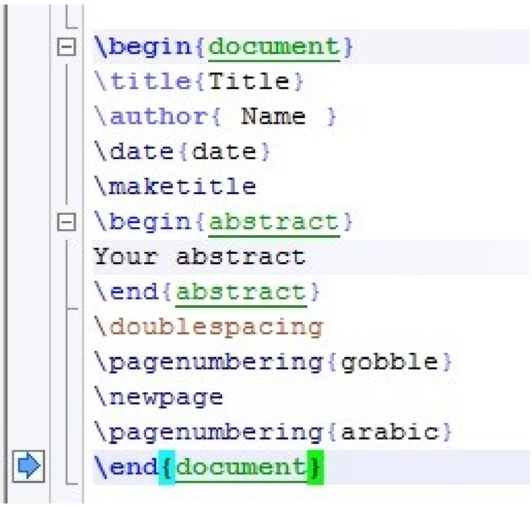
```
\documentclass[12pt,a4paper]{article}
\usepackage{amssymb}
\usepackage{float,epsfig,latexsym,dcolumn,lscape,longtable,setspace}
\usepackage{makeidx}
\usepackage{graphicx}
\usepackage{amsmath}
\usepackage[square,sort,comma,numbers]{natbib}
\usepackage{fancyhdr}
\usepackage{lscape}
\usepackage{hyperref}
\usepackage{appendix}
\usepackage{rotating}
\usepackage{subfig}
\usepackage{multirow}
\usepackage{graphicx}
\usepackage{epstopdf}
\usepackage{geometry}
\usepackage{pdflscape}
\usepackage{longtable}
\usepackage{dcolumn}
\usepackage{setspace}
\interfootnotelinepenalty=10000
```

A brief introduction

LaTeXis an open-source typesetting program that can create professional documents. Commands in LaTeXuse the backslash (\). You can add packages by including them before creating the document, by using `\usepackage {packagename}`. The first command that is used is the `\begin{document}`, which opens the document environment. It is ended by the `\end{document}`. This specific logic applies for many commands that can create figures, plots, tables, equations, and so forth. You compile a PDF document by selecting the TeX menu and the PDF submenu. Afterwards, it will start compiling, and will report any errors, warnings, or bad sectors. Errors are usually the result of conflicting commands, packages, or missing brackets. This document will deal with creating a LaTeXdocument structure, mathematics, tables, figures, and pictures, and will explain how to format text.

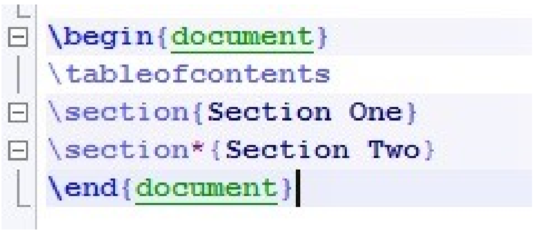
Structure

Creating the title page is simple, as you only need to use two commands. The first one is `\title{titlename}`, which should come after `\begin{document}`. The second command that will create the title page is `\maketitle`. You can



```
\begin{document}
\title{Title}
\author{ Name }
\date{date}
\maketitle
\begin{abstract}
Your abstract
\end{abstract}
\doublespacing
\pagenumbering{gobble}
\newpage
\pagenumbering{arabic}
\end{document}
```

also add an abstract, date, author, and footnotes, if needed. If you want the documents date to update every time that you create a pdf, merely add an asterisk (*) between the brackets. If you do not want to include a date on the title page, use `\date{}`. This will disable the date function when using `\maketitle`. However, you must put those commands between the title name and the execution command. The commands for these steps are `\date{date}`, `\author{name}`, and `\begin{abstract}`. For footnotes, please see below. It is best to use the `\newpage` after making the title page, to ensure that there is a clear separation from the main body of the document. You can follow this with the `\newpage` command to separate the title page and the section after it. In the case that you wish to erase the pagenumber from the title page, you can add `\pagenumbering{gobble}`. Add `\pagenumber{Arabic}` afterwards to re-enable `pagenumbering`.

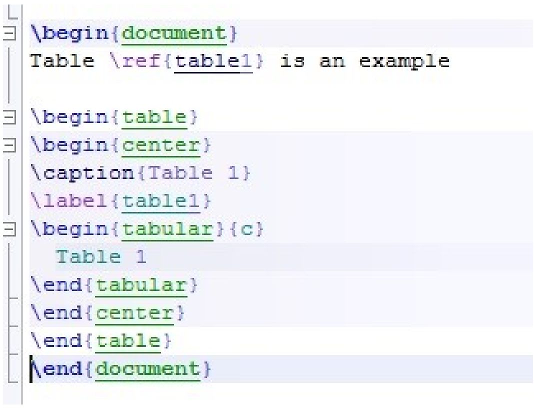


```
\begin{document}
\tableofcontents
\section{Section One}
\section*{Section Two}
\end{document}
```

Figures	Fig.
Tables	Tab.
Equations	Eq.

Creating chapters is done by using `\section{chaptername}`. You can add an asterisk (*), texttt\section*{chaptername}, which disables the numbering for that specific chapter. Subsections are added through texttt\subsection `{subname}`. The table of contents, list of figures, and list of tables are added by `\tableofcontents`, `\listoffigures`, and `\listoftables`, respectively. Remember that by using `\begin{table}`, `\begin{figure}`, and `\begin{equation}` will be automatically listed in the List of Tables and List of Figures.

How do you reference to tables, figures, and equations in the main text? When creating tables, figures, equations, and footnotes, etcetera, you have to add the command `\label` within the specific environment. Be sure to make these labels distinct, often the following references are used:



```
\begin{document}
Table \ref{table1} is an example

\begin{table}
\begin{center}
\caption{Table 1}
\label{table1}
\begin{tabular}{c}
  Table 1
\end{tabular}
\end{center}
\end{table}
\end{document}
```

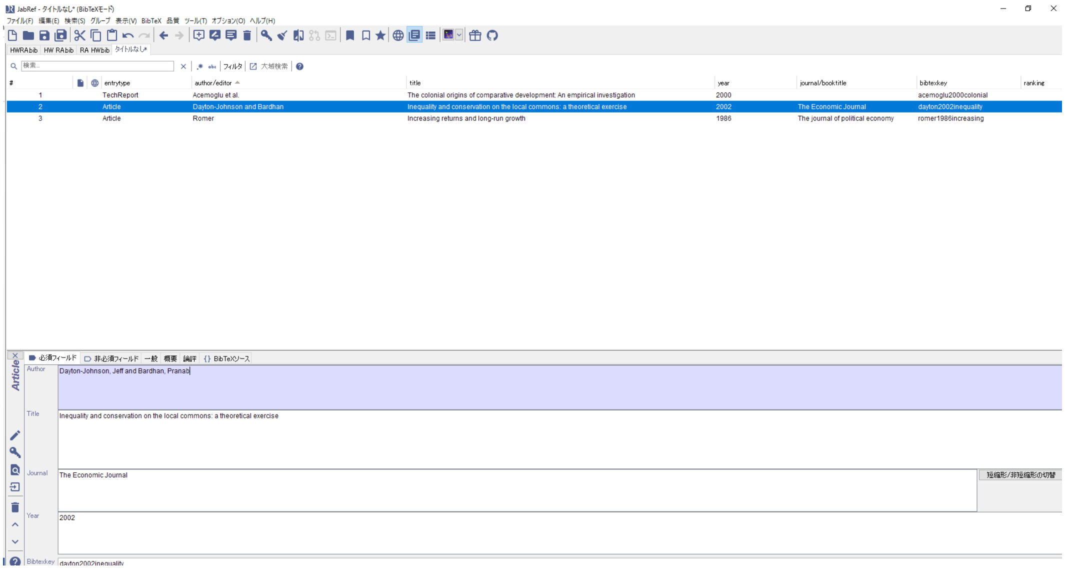
You can reference to them by adding `\refname`, as seen in the following example. However, this only references to the number of the table or figure.

References and bibliography

To create a bibliography, you must use a program that creates a .bib file, such as JabRef (open-source). You can normally download the reference from the journal in which the article was published. They will normally have a BiBTeX extension, which you can either download or copy and paste into a reference editor, such as JabRef. In the case that you do not want to use a third-party program for references, you can also use your LaTeX program to create a reference database. Firstly, you have to create a file with a .bib extension. One example for having a reference to an article could be:

```
   @article{Gettys90,
author = {Jim Gettys and Phil Karlton and Scott McGregor},
title = {The {X} Window System, Version 11},
journal = {Software Practice and Experience},
volume = {20},
number = {S2},
year = {1990},
abstract = {A technical overview of the X11 functionality.  This
is an update
of the X10 TOG paper by Scheifler \& Gettys.}
}
```

However, the structure of the reference can change depending on what you are specifically referencing, e.g. an article, technical report, book, etc. Please research what type of reference you are including in your document.



#	entrytype	author/editor	title	year	journal/booktitle	bibtexkey	ranking
1	TechReport	Acemoglu et al.	The colonial origins of comparative development: An empirical investigation	2000		acemoglu2000colonial	
2	Article	Dayton-Johnson and Bardhan	Inequality and conservation on the local commons: a theoretical exercise	2002	The Economic Journal	dayton2002inequality	
3	Article	Romer	Increasing returns and long-run growth	1986	The journal of political economy	romer1986increasing	

Author: Dayton-Johnson, Jeff and Bardhan, Pranab

Title: Inequality and conservation on the local commons: a theoretical exercise

Journal: The Economic Journal

Year: 2002

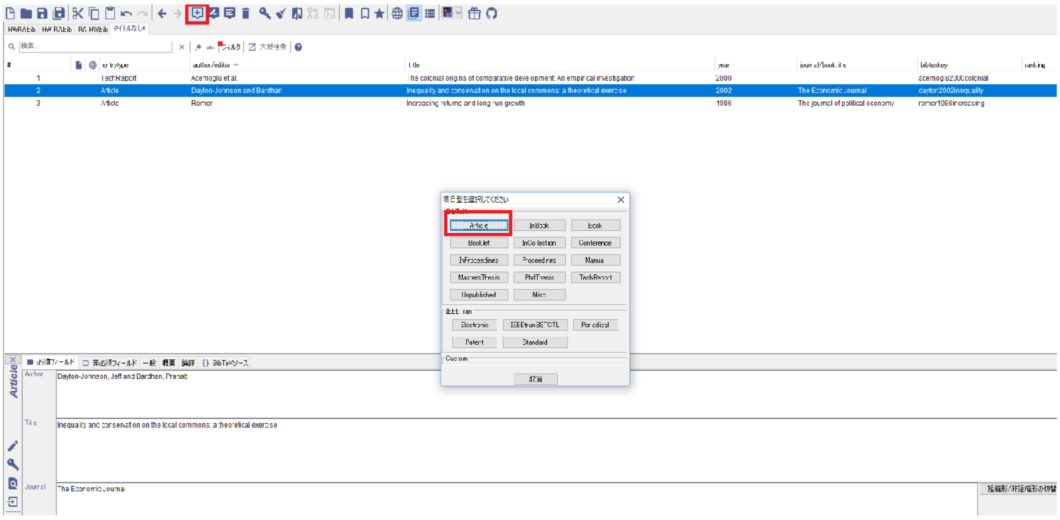
Bibtexkey: dayton2002inequality

JabRef

About JabRef

As mentioned above, JabRef is a free software package for managing BibTeX databases (.bib). It can be downloaded from http://www.jabref.org/. The image above shows what JabRef looks like.

In order to create a new entry, you should click on the New BibTex entry icon (highlighted in the upper part of the figure below). You will then have to choose the type of document you will use in your reference. You can choose from options like article, thesis, book, etc. In this example, we have chosen article (see figure below). There are two common ways to manage the reference list. One option is to manually write or copy the reference from the journal, google scholar, or IDEAS, as explained above. The second option is easier, therefore we only focus on it here. First, you have to get the reference from the journal used. Then, you must paste the reference in the lower part of the BibTeX source tag. After that, you should proceed to the next reference, until you have finished the reference list. In the case of using the website IDEAS, you need to click on citations, choose the BibTeX option, and download it. The citation will appear in a new window, and you just need to copy it and paste it into JabRef. JabRef can sort the reference list alphabetically by author, so you do not have to worry about that. When you are done adding documents to the database, you should save the bibliography using the Save button in the toolbar. The document will be saved as a .bib



#	entrytype	author/editor	title	year	journal/booktitle	bibtexkey	ranking
1	TechReport	Acemoglu et al.	The colonial origins of comparative development: An empirical investigation	2000		acemoglu2000colonial	
2	Article	Dayton-Johnson and Bardhan	Inequality and conservation on the local commons: a theoretical exercise	2002	The Economic Journal	dayton2002inequality	
3	Article	Romer	Increasing returns and long run growth	1986	The journal of political economy	romer1986increasing	

Author: Dayton-Johnson, Jeff and Bardhan, Pranab

Title: Inequality and conservation on the local commons: a theoretical exercise

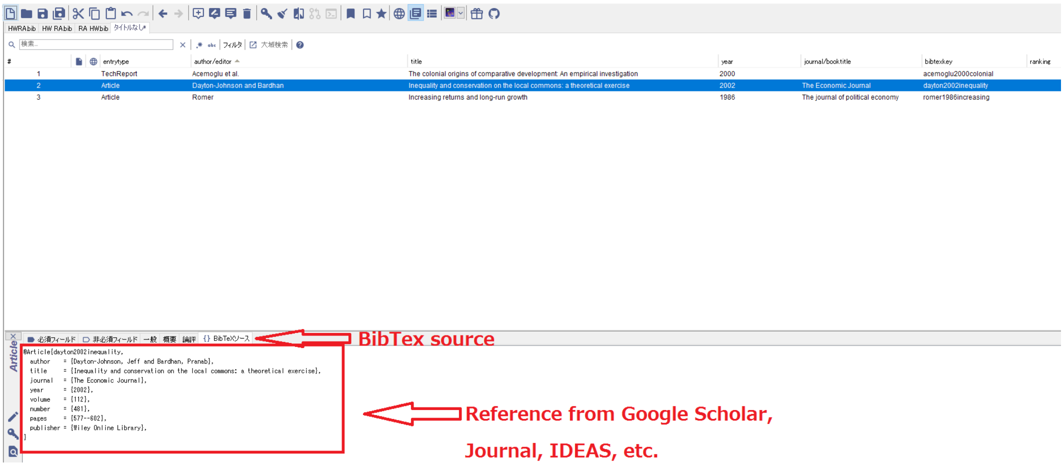
Journal: The Economic Journal

New entry on JabRef

file.

Now that we have learnt how to manage and save references using JabRef, next we are going to learn how to import references with the LaTeX compiler. First, you should open the compiler with the desired TeX file. Then, you should click on the open new file button, and choose the .bib document that you created with JabRef. You must save the .bib file into the same location as the TeX location. In order for LaTeX to read the .bib file, use the command `\bibliography{.bib file}` (remember that spaces in the .bib file name must be replaced by _). Afterwards, you can specify the style that should be used. This is done by using `\bibliographystyle{stylename}`. The last remaining step is to cite the authors that you have cited in the paper. You can do it manually by citing each one with the`\cite` command. However, this will take some time. Therefore, you can use `\nocite{*}`, which will cite all of the entries in the .bib file. Use the BiBTeX option in the compiling section to include the bibliography by using the normal compiler afterwards. The PDF that is created should have the bibliography included. The following images will help you to understand how to import .bib files with LaTeX.

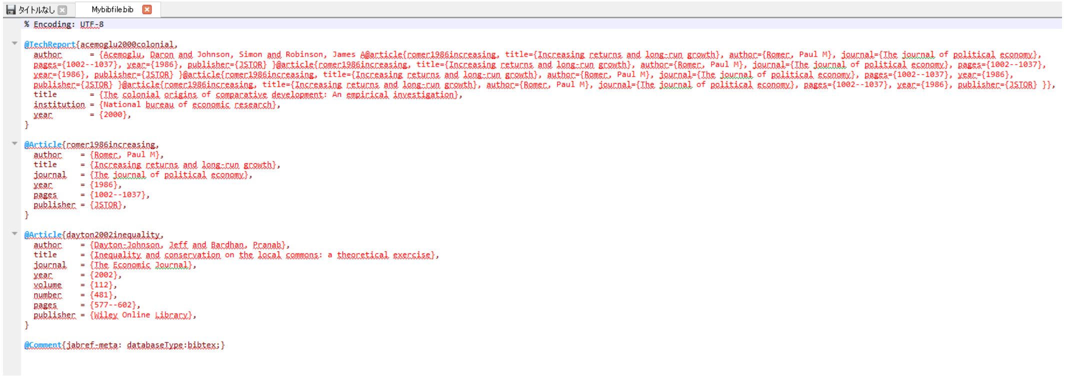
There are many compilers, the most popular are Texmaker, Texstudio and LyX. JabRef has an icon to cite directly to the LaTeX: compilers that we are using. Footnotes can be added with the `\footnotefootnote text`. In the case that the footnote text is extremely long, it will most likely be split between pages. You can add `\interfootnotelinepenalty=10000` at the top of the LaTeXdocument to increase the text size used for the footnotes.



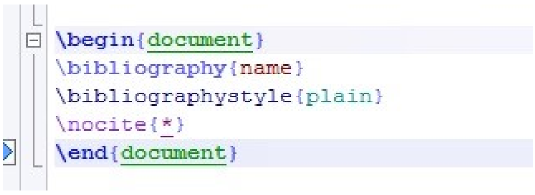
#	entrytype	author/editor	title	year	journal/booktitle	bibtexkey	ranking
1	TechReport	Acemoglu et al.	The colonial origins of comparative development: An empirical investigation	2000		acemoglu2000colonial	
2	Article	Dayton-Johnson and Bardhan	Inequality and conservation on the local commons: a theoretical exercise	2002	The Economic Journal	dayton2002inequality	
3	Article	Romer	Increasing returns and long-run growth	1986	The journal of political economy	romer1986increasing	

```
@Article{dayton2002inequality,
  author    = {Dayton-Johnson, Jeff and Bardhan, Pranab},
  title     = {Inequality and conservation on the local commons: a theoretical exercise},
  journal   = {The Economic Journal},
  year      = {2002},
  volume    = {112},
  number    = {481},
  pages     = {577--602},
  publisher = {Wiley Online Library},
}
```

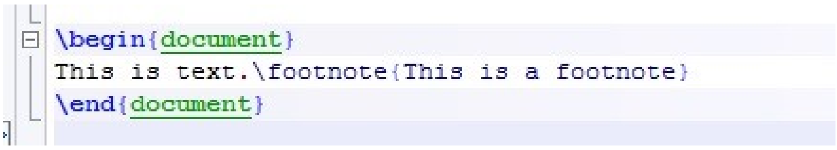
Bibtex source on JabRef



```
% Encoding: UTF-8

@TechReport{acemoglu2000colonial,
  author      = {Acemoglu, Daron and Johnson, Simon and Robinson, James A@article{romer1986increasing, title={Increasing returns and long-run growth}, author={Romer, Paul M}, journal={The journal of political economy},
  pages={1002--1037}, year={1986}, publisher={JSTOR} }@article{romer1986increasing, title={Increasing returns and long-run growth}, author={Romer, Paul M}, journal={The journal of political economy}, pages={1002--1037},
  year={1986}, publisher={JSTOR} }@article{romer1986increasing, title={Increasing returns and long-run growth}, author={Romer, Paul M}, journal={The journal of political economy}, pages={1002--1037}, year={1986},
  publisher={JSTOR} }@article{romer1986increasing, title={Increasing returns and long-run growth}, author={Romer, Paul M}, journal={The journal of political economy}, pages={1002--1037}, year={1986}, publisher={JSTOR} }},
  title       = {The colonial origins of comparative development: An empirical investigation},
  institution = {National bureau of economic research},
  year        = {2000},
}

@Article{romer1986increasing,
  author    = {Romer, Paul M},
  title     = {Increasing returns and long-run growth},
  journal   = {The journal of political economy},
  year      = {1986},
  pages     = {1002--1037},
  publisher = {JSTOR},
}

@Article{dayton2002inequality,
  author    = {Dayton-Johnson, Jeff and Bardhan, Pranab},
  title     = {Inequality and conservation on the local commons: a theoretical exercise},
  journal   = {The Economic Journal},
  year      = {2002},
  volume    = {112},
  number    = {481},
  pages     = {577--602},
  publisher = {Wiley Online Library},
}

@Comment{jabref-meta: databaseType:bibtex;}
```



```
\begin{document}
\bibliography{name}
\bibliographystyle{plain}
\nocite{*}
\end{document}
```



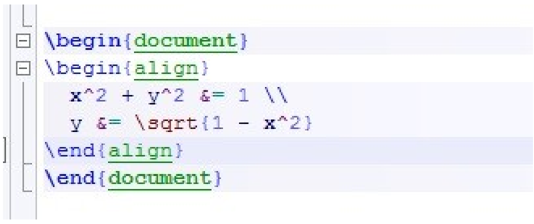
```
\begin{document}
This is text.\footnote{This is a footnote}
\end{document}
```

Should you wish to reference to a footnote, add \label{} into the footnote command environment as above.

Mathematics

One advantage of LaTeX is its ability to create professional looking equations. You can create equations in two separate ways, by either creating an equation environment by using \begin{equation} or \begin{align} if it will include several equations. The other method is to open a capsule (/(equation/)). However, the results are quite different. As mentioned before, when we use a \begin{} command, it automatically numbers the equation. This can be disabled by including an asterisk \begin{equation*}. Mathematical symbols can be used in either method. Using them outside either method will result in an error. You can use dollar signs ($) at the start and end of the mathematical symbol, e.g. δ, if you only wish to use that specific symbol in the text. In addition, you can add captions and labels to equations. Captions are added by \caption{captiontext}, either above or below the equation. Labels will allow you to reference them in the text through \label{labelname}. Should you have an equation that includes steps, or is composed out of several equations, you can use the align command \begin{align}.

The align command uses the ampersand symbol (&) and aligns them on the base of it. Be sure to use \\for row breaks. If the brackets for the fractions or matrices are out of place (in terms of scale), you can add texttttextbackslash left[at the top and \right] at the bottom to make the brackets scale correctly. This can be used for various commands.



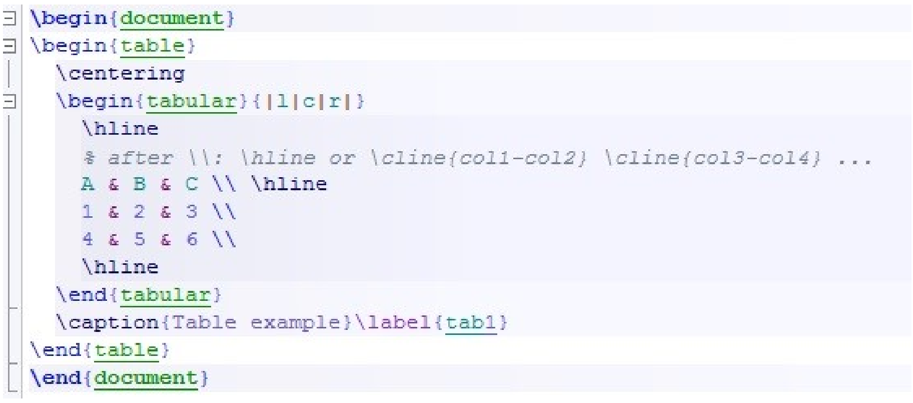
```
\begin{document}
\begin{align}
  x^2 + y^2 &= 1 \\
  y &= \sqrt{1 - x^2}
\end{align}
\end{document}
```

Table and figures

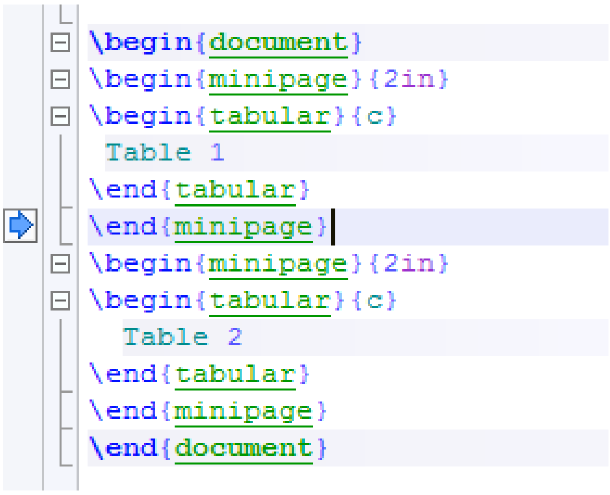
LaTeXalso allows you to make tables and include figures. Tables are creating by starting with `\begin{table}`. Tables automatically include `\centering` if you tab the selection in the menu pop-up (depends on your software). To start creating the table, you have to follow this with `\begin{tabular}` `{column|column|..}` . Text in the columns can be aligned either left (l), center (c), or right (r). To separate columns, you have to use — after each specification. You can label and add a caption in the same way as with equations. In order to create horizontal black lines, as seen in scientific papers, the command `\hline` is used. Values for the table can be inserted below and separated by column using the & symbol. The number of values should be equal to the number of columns specified. Row ends are specified by adding \\at the end. If mathematical symbols are required, remember that you should use $ before and after to include them.

If a table is quite long due to the number of columns, you can transform the page into landscape setting. Add `\begin{landscape}`before `\begin{table}` and `\end{landsacpe}` after `\end{table}`. This will ensure that the page will only be in landscape for the page that holds the table. You can also make the table smaller by using the smallbox or scalebox command, where 1 = 100% and 0.75 = 75% (25% reduction in size). Place the command `\scalebox{number}` { before the tabular command (`\begin{tabular}`) and add the last bracket } at the end of the table. You can also add two tables side-by-side by creating two minipages. Within the minipage, you can construct the table using the tabular command. You can see an example of this below:

Regression, summary statistics, and other types of academic tables need to include notes. LaTeX has a very useful environment to add notes. First, we start constructing our table with the Tabular environment, as described above. Then, after ending the tabular environment, we have to begin the table notes environment. The commands are as follows: `\begin{tablenotes}` `\footnotesize \itemsep 0pt \item \noindent $ \textbf{Notes:}$ HERE WRITE YOUR NOTE \end{tablenotes}`.

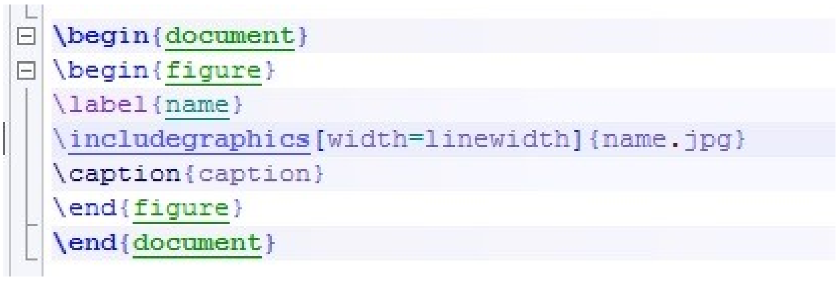


```
\begin{document}
\begin{table}
  \centering
  \begin{tabular}{|l|c|r|}
    \hline
    % after \\: \hline or \cline{col1-col2} \cline{col3-col4} ...
    A & B & C \\ \hline
    1 & 2 & 3 \\
    4 & 5 & 6 \\
    \hline
  \end{tabular}
  \caption{Table example}\label{tab1}
\end{table}
\end{document}
```



```
\begin{document}
\begin{minipage}{2in}
\begin{tabular}{c}
 Table 1
\end{tabular}
\end{minipage}
\begin{minipage}{2in}
\begin{tabular}{c}
  Table 2
\end{tabular}
\end{minipage}
\end{document}
```

Tabular Example



```
\begin{document}
\begin{figure}
\label{name}
\includegraphics[width=linewidth]{name.jpg}
\caption{caption}
\end{figure}
\end{document}
```

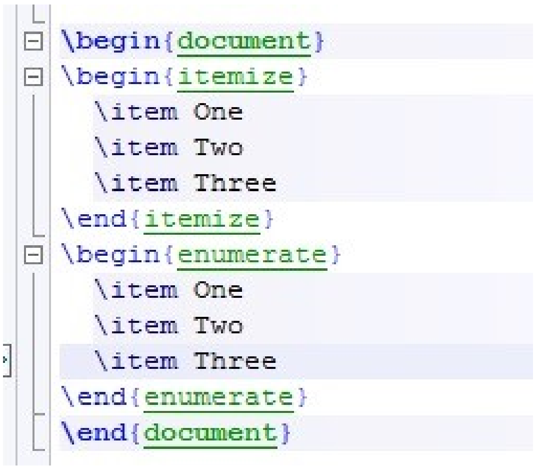
Pictures can be added by using the graphic package. If you wish to automatically number them, use `\begin{figure}` (use an asterisk if numbering is not required). Make sure that the picture is saved within the same location as the TeX location. Afterwards, use the command `\includegraphics[width=linewidth]{name.extension}`. Ensure that the file extension is the same as the one specified in the command. In addition, make sure that you control for spaces in the file name by adding __.

If you require a table or picture to be at specific location in the document, you can add a specification behind the command. If you wish the figure or table to be at the top of the document, add [!t]. Therefore, `\begin{figure}[!t]`. Other options are h for the same location and b for the bottom. In the case that you have too many figures and tables, where numbering becomes too complicated, you can reset them after a specific section. You can do this by adding the following code to the package amsmath: `Grouping code`.

When you compile a normal table using the tabular environment, you will notice that it does not stretch according to the page width. Therefore, if you want it to look better, you can add this command: While not usual, sometimes coloring is added to bring out each row separately. I would recommend staying with white and grey. This can be done by adding the rowcolour command: `\rowcolor{colour!number}`.

Text formatting

Sometimes we have to change the line spacing within a document. This can be done by opening a space environment by using `\begin{spacing}{ space number}` and closed with`\end{spacing}`. If you merely wish to double space, it becomes `\begin{doublespace}` and closed with `\end{doublespace}`. You can create text that is bold, italic, or emphasized by using `\textbf{}`, `\textit{}`, and `\emph{}`, respectively. You can change the text font by using specified text commands such as `\textrm{}` for roman font family, `\texttt{}` for typewriter font, and `\textsf{}` for sans seriff font. The text



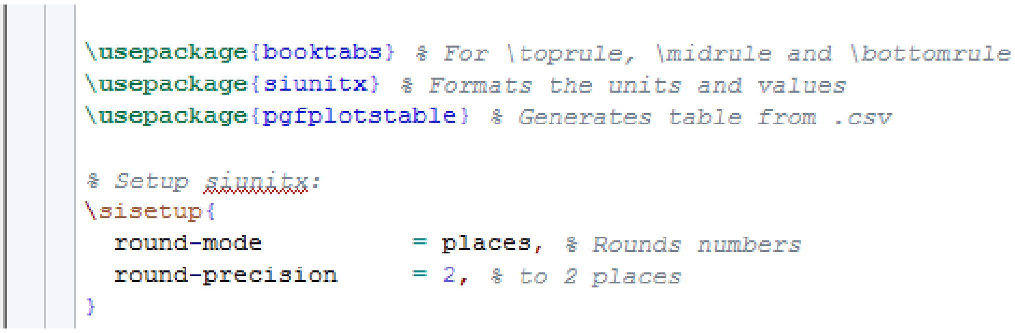
```
\begin{document}
\begin{itemize}
  \item One
  \item Two
  \item Three
\end{itemize}
\begin{enumerate}
  \item One
  \item Two
  \item Three
\end{enumerate}
\end{document}
```

size can be changed by applying commands such as \tiny, \footnotesize, \small, and etcetera. If you wish to use bullet points, there are various versions that can be used. Itemize and enumerate are both bullet points, but the difference is that one is numbered with dots.

With regards to paragraphs, you use the \par command. This will make the text after the command start as a new paragraph. You can also align the paragraph to the left, center, or right by using \begin{flushleft, center, or flushright}. As for indents, you can use \setlength{\parindent}{10ex} followed at the end of the indented text with \noident. In this case, 10ex is the length of how much will be indented (the length depends on the x of the font you are using).

Importing tables and generating plots

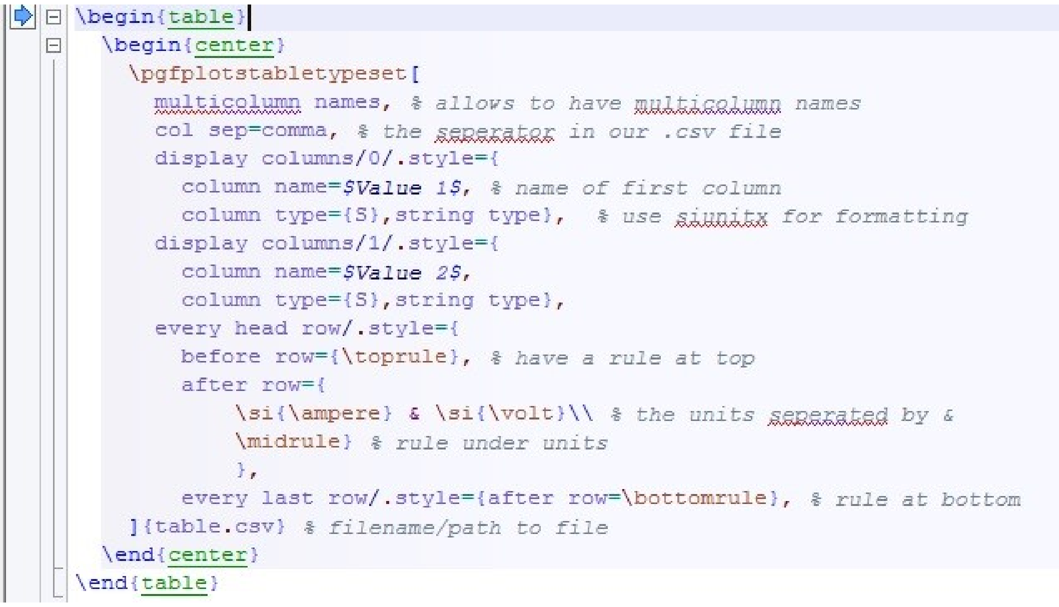
LaTeXpackages also give you the option to import tables from .csv files (Excel). This can increase your efficiency, as creating tables by hand takes time. In addition, you could also use an online table generator. However, there might be some command problems, depending on which packages you are using. You can also generate your own plot charts with LaTeXif needed. However, you will need to use the package called pdfplots to enable the commands for this. To generate a table, you must first include the packages booktabs, siunix, and pgfplotstable. These packages deal with the layout, formatting, and eventually generating the table from the .csv file. Make sure that the .csv files are in the right format, where commas are seen as column separators and newline is used as a row separator.



```
\usepackage{booktabs} % For \toprule, \midrule and \bottomrule
\usepackage{siunitx} % Formats the units and values
\usepackage{pgfplotstable} % Generates table from .csv

% Setup siunitx:
\sisetup{
  round-mode          = places, % Rounds numbers
  round-precision     = 2, % to 2 places
}
```

Before starting with making the table, you can copy the following code in the same space as \usepackage. It contains information on how LaTeX should process data into a table format.

The setup of siunitx deals with formatting the data from the .csv file. Round mode tells LaTeX to round up the numbers that are in the table, and round-precision tells it that it allows up to two places behind the comma. To generate the table, you can start with the normal \begin{table}, centering, label, and caption. Instead of using \begin{tabular}, use \pgfplotstable typeset[instead. We have to use a bracket [] to indicate from where the table starts and ends. When using the pgfplot package, you have to specify per column and what that column type is e.g. numeric, string. The booktabs package allows the usage of \toprule, \midrule, and \bottomrule. These commands act the same as \hline, but the line made by booktab is much thicker. It is best not to re-write these commands every time. Instead you can re-use by merely copying the code and changing it to your needs when required. If you want to add a row, you merely have to add the following:

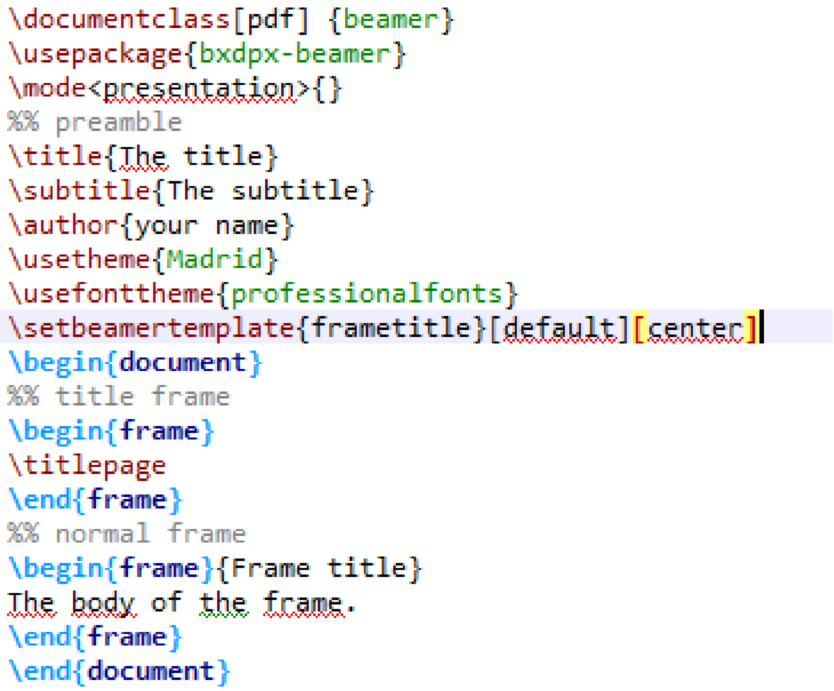


```
\begin{table}
  \begin{center}
    \pgfplotstabletypeset[
      multicolumn names, % allows to have multicolumn names
      col sep=comma, % the seperator in our .csv file
      display columns/0/.style={
        column name=$Value 1$, % name of first column
        column type={S},string type},  % use siunitx for formatting
      display columns/1/.style={
        column name=$Value 2$,
        column type={S},string type},
      every head row/.style={
        before row={\toprule}, % have a rule at top
        after row={
            \si{\ampere} & \si{\volt}\\ % the units seperated by &
            \midrule} % rule under units
            },
        every last row/.style={after row=\bottomrule}, % rule at bottom
    ]{table.csv} % filename/path to file
  \end{center}
\end{table}
```

Presentations in LaTeX

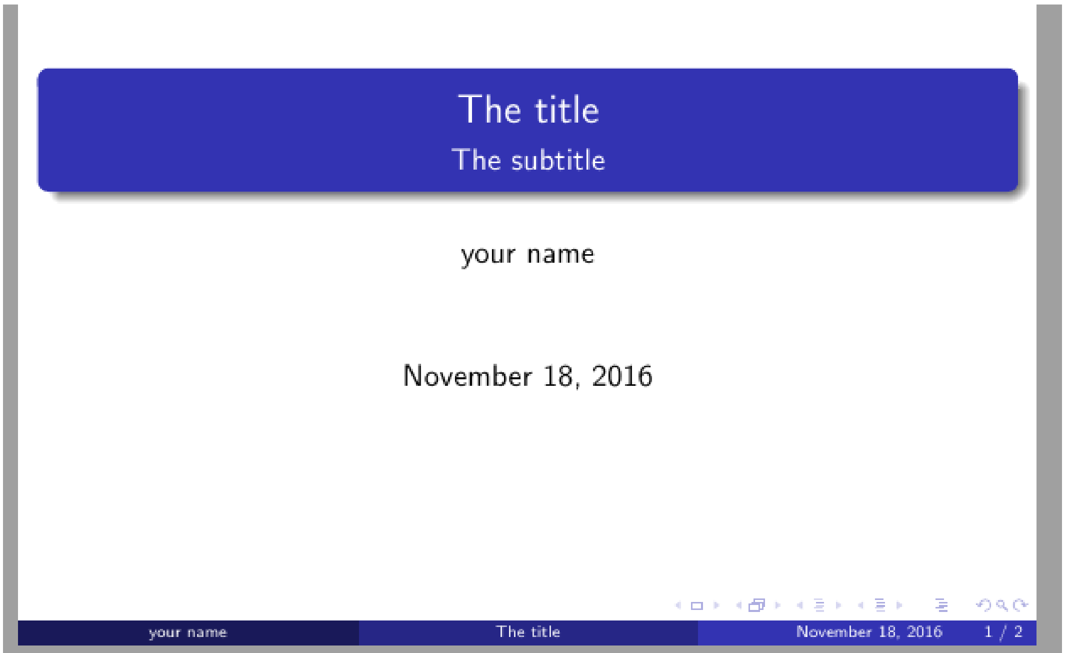
LaTeX permits us to create slide shows and presentations using the class. Beamer supports functionality for making PDF slides, complete with colors, overlays, environments, themes, etc. Beamer also supports bibliographies, appendices, and transitions. Short versions of the title, authors, and institution can be specified as optional parameters. It also supports figure and table environments, transparency effects, animation commands, and pauses environment.

The basic structure is the following



```
\documentclass[pdf] {beamer}
\usepackage{bxdpx-beamer}
\mode<presentation>{}
%% preamble
\title{The title}
\subtitle{The subtitle}
\author{your name}
\usetheme{Madrid}
\usefonttheme{professionalfonts}
\setbeamertemplate{frametitle}[default][center]
\begin{document}
%% title frame
\begin{frame}
\titlepage
\end{frame}
%% normal frame
\begin{frame}{Frame title}
The body of the frame.
\end{frame}
\end{document}
```

Where beamer is the document class. The package bxdpx-beamer is just a driver to support beamer. We can see that after the preamble, the rest of the document is quite similar to any other LaTeXstructure that we have learnt thus far. Using beamer, a presentation is prepared with a number of frames. Every time that you want to start a frame, you must begin the frame environment. Each frame consists of one or more slides. The headline of each frame is the upper part of the frame. It is important to notice that the headline, color, theme, and font are installed at the beginning, before the document environment. The pdf results of this basic template are shown in the figure below.



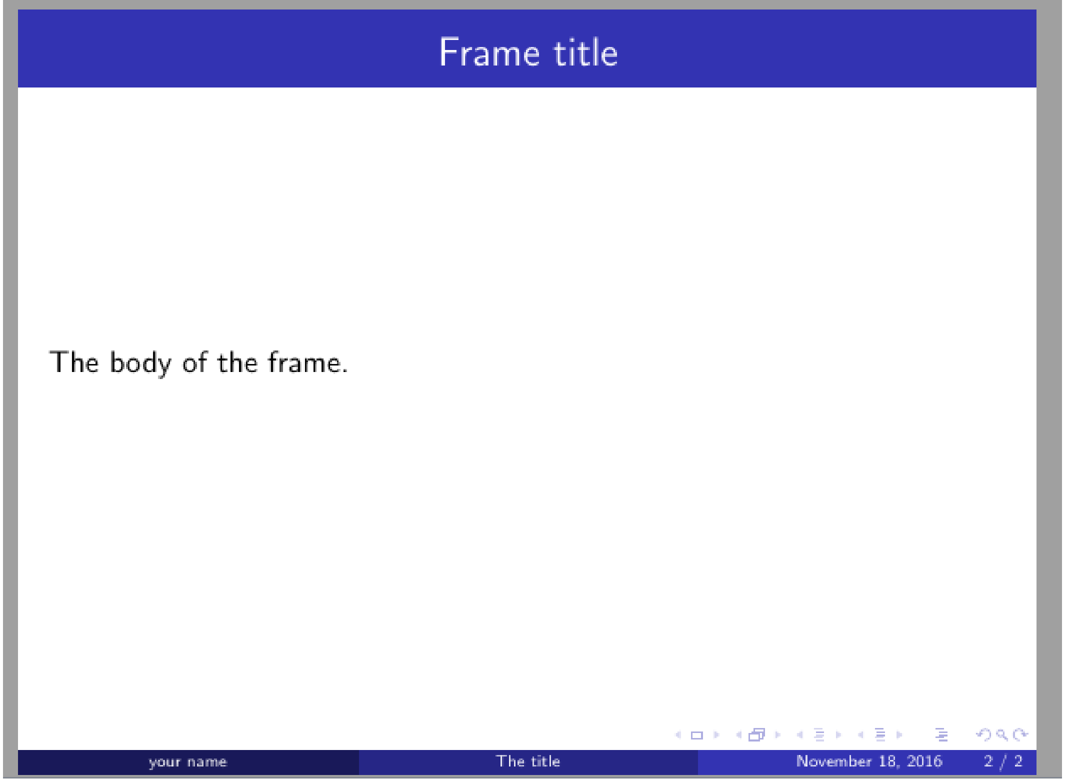
The title

The subtitle

your name

November 18, 2016

your name | The title | November 18, 2016 1 / 2



Frame title

The body of the frame.

your name | The title | November 18, 2016 2 / 2

Thesis

Every Master's and PhD student in the Department of Economics and Management of Tohoku University has to write a thesis in order to obtain their degree. Writing a thesis is demanding, and takes a lot of time. As we have learned so far, LaTeX is a powerful document editor, and it comes in handy when writing a thesis. Therefore, in this section, we will introduce briefly how to write a thesis using LaTeX.

We need to do a couple of things before getting into writing the thesis. First, we have to pick a template and use it for the thesis. We can create our own template, although this requires higher knowledge of LaTeX, or we can download an already structured template online, as the Department of Economics and Management of Tohoku University does not have a mandatory template or thesis style. There are a number of reliable websites from which we can download various types of templates (e.g. https://www.sharelatex.com/templates/thesis, http://www.latextemplates.com/template, etc.). For this introduction to LaTeX, we will use and follow a thesis style that was created by Steve R. Gunn and converted into a template by Sunil Patel. We can download this template from http://www.latextemplates.com/template/masters-doctoral-thesis. Secondly, we need to run the already-downloaded template. This is very simple, we just need to run in our compiler the main.tex file, in the folder that we just downloaded. The Master's and Doctoral Thesis file includes three folders, a BibTeX file, a PDF file, a TeX document named file, and many other types of files. Our documents of interest are the BibTeX file (our reference list), the main LaTeX file (we will edit our thesis there), and the PDF of the thesis.

We chose the Maters/Doctoral Thesis template by Gunn and Patel because it is carefully structured and very easy to edit. In the next page, we can see the cover page of a thesis created using the chosen template. We can edit every single part of it easily, and in the main document the authors added a great number of very useful observations that we discovered when writing our own theses. The main document brings a cover page, a declaration of authorship, quotation, abstract, acknowledgments, contents, list of figures, list of tables, abbreviations, symbols, and a dedication page. It also includes an example chapter, an example appendix, and a bibliography.

We need to personalize the thesis template when we start writing our own project. This template facilitates the writing process because it is already well structured, and we just have to save our editions in the main file. As we learned before, when we save in the main.tex file, the PDF file will be edited automatically.

We should read in detail the main document downloaded in the PDF.

TOHOKU UNIVERSITY

DOCTORAL THESIS

Thesis Title

Author:
John SMITH

Supervisor:
Dr. James SMITH

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

in the

Research Group Name
Graduate School of Economics and Management

January 8, 2017

Tohoku University

Abstract

Faculty Name
Graduate School of Economics and Management

Doctor of Philosophy

Thesis Title

by John SMITH

The Thesis Abstract is written here (and usually kept to just this page). The page is kept centered vertically so can expand into the blank space above the title too…

It explains the template, thesis features, and conventions, etc. Including figures, tables, equations, and footnotes is done in the same fashion as for any other LaTeX document. It is important to notice that each chapter of the thesis should be prepared separately in its own .tex file. Another advantage of the Gunn and Patel' thesis template is that it takes care of the table of contents, list of figures, and tables. Therefore, these lists do not need to be edited manually. However, it is possible to edit and personalize the table of contents, as you will see in the next subsection.

Table of Contents

The table of contents can be created in many ways; some ways might be very complicated, while others might be simple. We will learn how to create a table of contents using the packages tocloft and titletoc `\usepackage {tocloft}`, `\usepackage {titletoc}`. This package provides a great number of commands to elaborate the table of contents, list of figures, and other similar tables needed in the thesis.

To start writing the table of contents, we need to do a couple of things before beginning the document. First, we need to change the name of the Table of Contents using the following command `\renewcommand \contentsname{Table of Contents}`. Second, we need to establish the headings for every page of the Table of Contents, using `\newcommand\tocheading{\par\bigskip\MakeUppercase {\chaptername} \hfill Page\par}`. The centering titles for the Table of Contents can be done using `\renewcommand{\cfttoctitlefont\hfill \normalfont\MakeUppercase}`.

After personalize centering and titles, we have to format the front matter chapters, like acknowledgments, etc. We do this using the following command:

```
\newcommand\frontmatterchaptoc{%
\titlecontents{chapter} [1.5em]{\addvspace\baselineskip/}
{\contentslabel{1.5em}\MakeUppercase}
{\hspace*{-1.5em}\MakeUppercase}
{\titlerule*[1pc]{.}\contentspage}
}
```

For the main chapters, the following command is needed:

```
\newcommand \{ mainmatterchaptoc{%
\titlecontents{chapter}[5em]{\addvspace{\baselineskip}}
{\contentslabel{3em}\hspace*{-1em}\MakeUppercase}
{\MakeUppercase}
{\titlerule*[1pc]{.}\contentspage}
}
```

For sections and subsections, the following commands are included:

```
\titlecontents{section}[7em]{}{\hspace{-1em}}{}
{\titlerule*[1pc]{.}\contentspage}
\titlecontents{subsection} [7.5em]{}{\hspace{-1em}}
{}{\titlerule*[1pc]{.}\contentspage}
```

Finally, we can begin the document. After the cover page, use the commands `\frontmatter` and `\frontmatterchaptoc` to activate the formatting in the front matter. Then, write `\cleardoublepage`, `\tableofcontents` `\cleardoublepage`. Use the command `\mainmatter`, then `\addtocontents{toc}` `{\tocheading}` to add a heading to the first page of the Table of Contents, and use `\mainmatterchaptoc` to activate the chapters defined as main matter. This is all that you need to do to create a table of contents by yourself in LaTeX. A list of tables and a list of figures can both be created in the same fashion, using the same commands. The next page shows an example of a very simple table of contents, created using the commands explained in this section.

TABLE OF CONTENTS