Welcome to the Introduction to LaTeX and Overleaf for Scientific Articles

LaTeX vs. Traditional Text Editors

Fancy looking documents!

Reproducible complex formatting and text layout with ease!

Typical looking documents.

Manual text formatting and layout!

Concepts discussed in the **LaTex and Overleaf** course:

- What typesetting is
- What LaTeX is used for and how it came to be
- How Overleaf makes writing LaTeX documents easier
- How to work with a template for a scientific article intended for arXiv
- How to collaborate on Overleaf
- How to get help if you run into issues

Overall Course Learning Objectives:

This course will demonstrate how to:

1. Be familiar with what LaTeX is and how it came to be
2. Understand the benefits of LaTeX and Overleaf
3. Get started writing in Overleaf with a template for a scientific manuscript
4. Add references to a document in Overleaf
5. Get help if you encounter challenges
6. Work collaboratively on Overleaf

Overleaf

Overleaf is a free online LaTeX editor, meaning it helps you write text using LaTeX to determine how your document will look.

- To start, you need to make an account on Overleaf.
- To do so, you can go to https://www.overleaf.com/register in your browser.
- There are several ways to register:
  1. Register with an IEEE account
  2. Register with your Google account
  3. Register with your ORCID account
  4. Register with a different email address
  5. Log in with an institution. This is worth checking in case your university has a license, as this may provide you with more features.

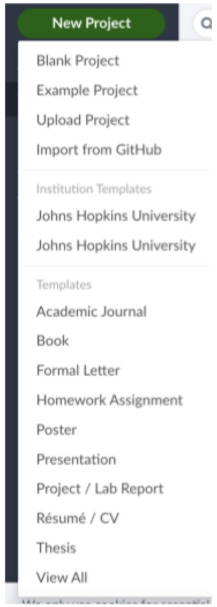
The free version only allows you to have one collaborator on a project, so collaboration is more limited. However, you can still benefit from many of the other features. Note that students can receive a discount as well.

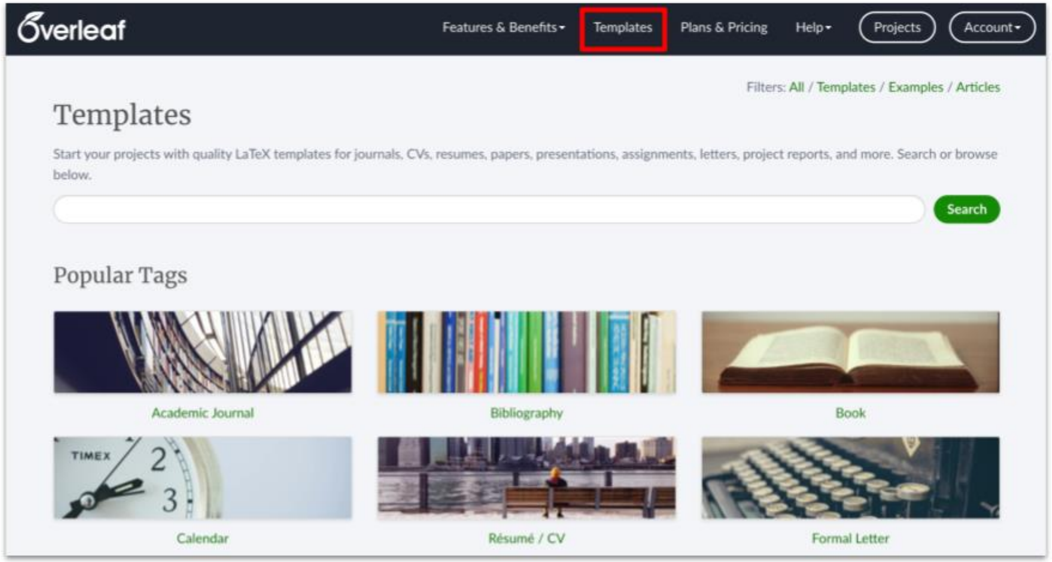
If possible, it may be worth seeing if your institute would get an organizational license.

Working with Templates

- Overleaf is organized with projects. When you create a project, you can start from scratch or start from a template.

Let’s start with a template to help us better understand the features of Overleaf. We can search through all the templates by clicking on “View All” button of the project menu. We can also click on the “templates” tab to get to the same template search page.





Templates

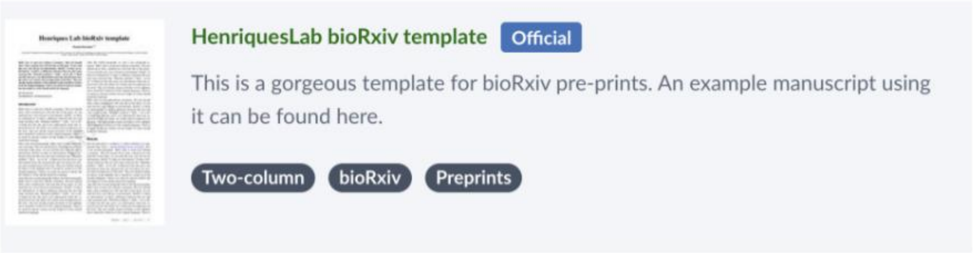
Start your projects with quality LaTeX templates for journals, CVs, resumes, papers, presentations, assignments, letters, project reports, and more. Search or browse below.

Popular Tags

https://www.overleaf.com/

CC-BY hutchdatascience.org

- You can see that there are many templates to choose from! There are templates for writing a letter, writing a thesis, writing a CV, writing a book and more.



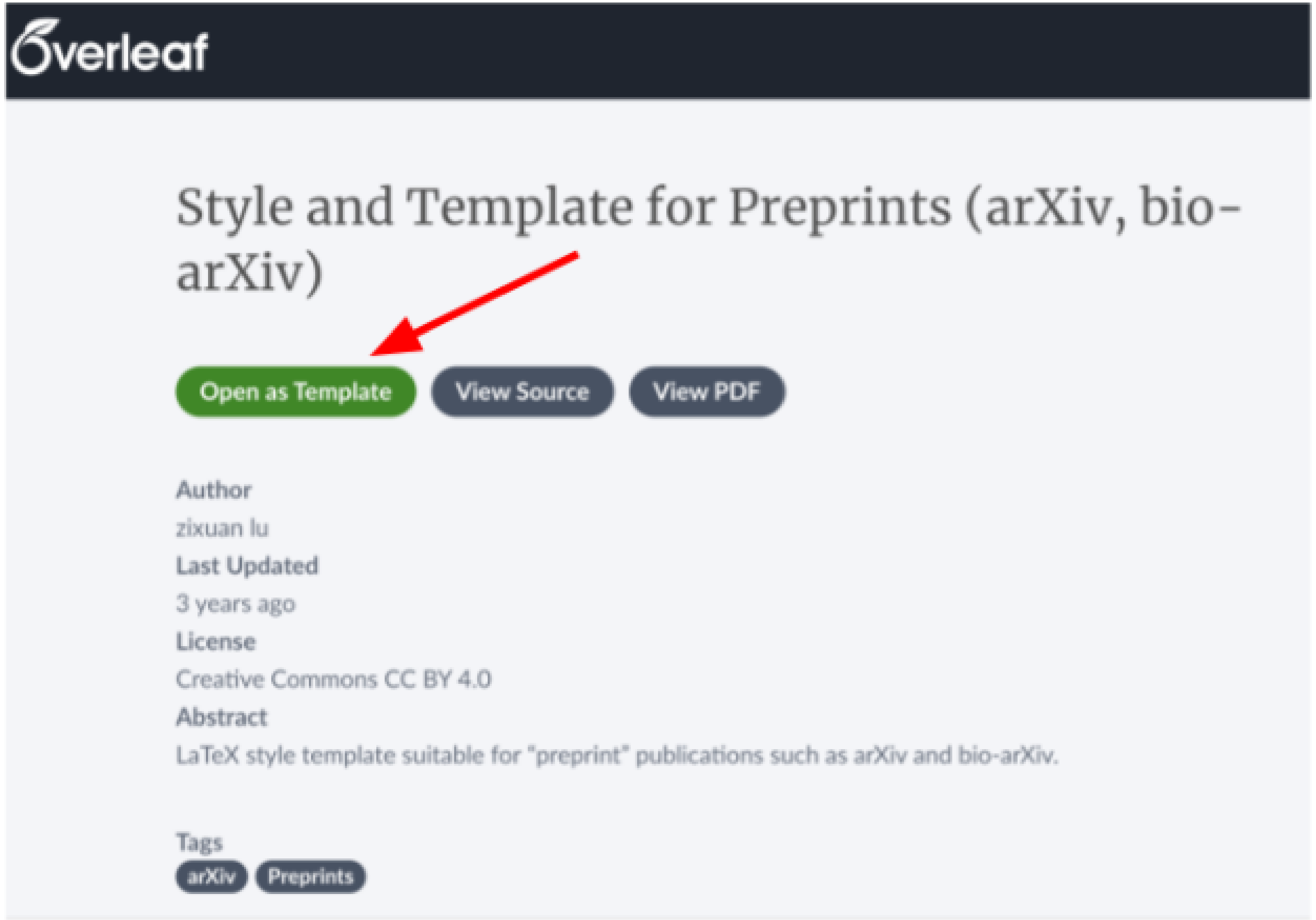
HenriquesLab bioRxiv template Official

This is a gorgeous template for bioRxiv pre-prints. An example manuscript using it can be found here.

Tip! - If you need to write a thesis or you have a student that needs to write one, check to see if your university or department has a LaTeX template that you can use to help everyone save time on formatting and spend more time writing the content.

- Let’s search for a term, such as “preprint” to see if the templates for preprints.
- You will see that a few of the templates have a blue “official” tag. This is the best option if you find one for the publisher that you are interested in.

- We will use a template for arXiv.
- You can go to this link (https://www.overleaf.com/latex/templates/style-and-template-for-preprints-arxiv-bio-arxiv/pkzcrhzcdxmc)to start with the same particular template to create a manuscript for arXiv. (https://arxiv.org/)
- Click the “Open as Template” button to get started.



Style and Template for Preprints (arXiv, bio-arXiv)

Author
zixuan lu

Last Updated
3 years ago

License
Creative Commons CC BY 4.0

Abstract
LaTeX style template suitable for "preprint" publications such as arXiv and bio-arXiv.

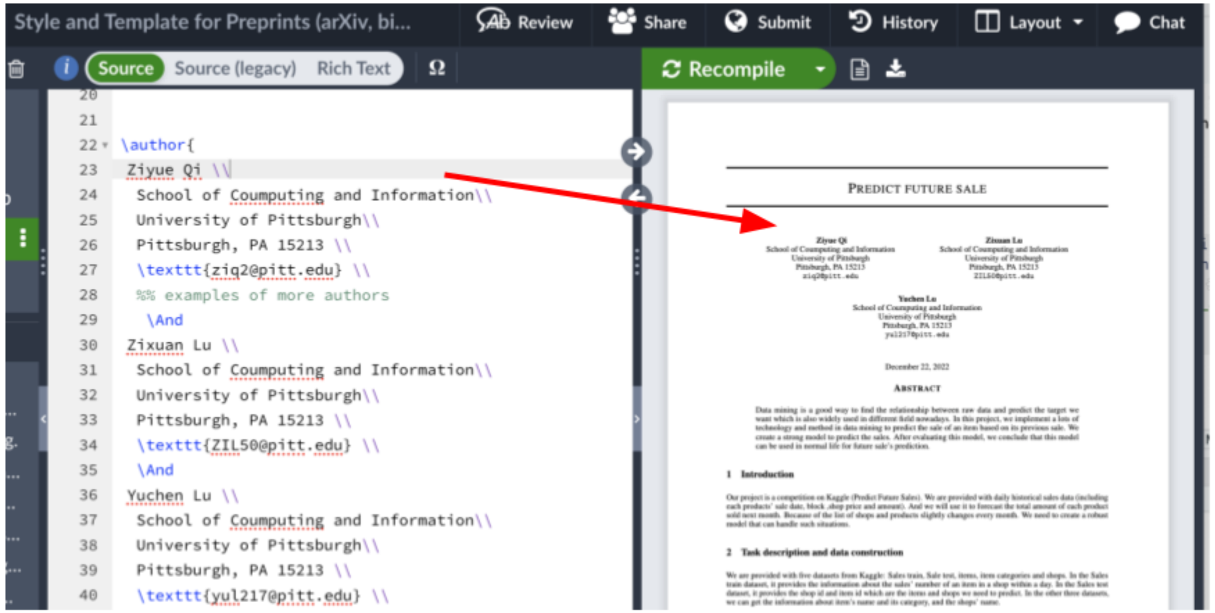
Tags

Viewing documents

- Now that we have opened a template, we will start by understanding how we can view our work on Overleaf.
- Overleaf easily shows you what the rendered version of your text will look like, taking into account the LaTeX tags that indicate how the text should be arranged and styled. The default view is to see the raw text (aka source text) on the left and the rendered/compiled version on the right.

Editor

PDF Viewer



```
\author{
 Ziyue Qi \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{ziq2@pitt.edu} \\
  %% examples of more authors
   \And
 Zixuan Lu \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{ZIL50@pitt.edu} \\
  \And
 Yuchen Lu \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{yul217@pitt.edu} \\
```

PREDICT FUTURE SALE

Ziyue Qi
School of Coumputing and Information
University of Pittsburgh
Pittsburgh, PA 15213
ziq2@pitt.edu

Zixuan Lu
School of Coumputing and Information
University of Pittsburgh
Pittsburgh, PA 15213
ZIL50@pitt.edu

Yuchen Lu
School of Coumputing and Information
University of Pittsburgh
Pittsburgh, PA 15213
yul217@pitt.edu

December 22, 2022

ABSTRACT

Data mining is a good way to find the relationship between raw data and predict the target we want which is also widely used in different field nowadays. In this project, we implement a lots of technology and method in data mining to predict the sale of an item based on its previous sale. We create a strong model to predict the sales. After evaluating this model, we conclude that this model can be used in normal life for future sale's prediction.

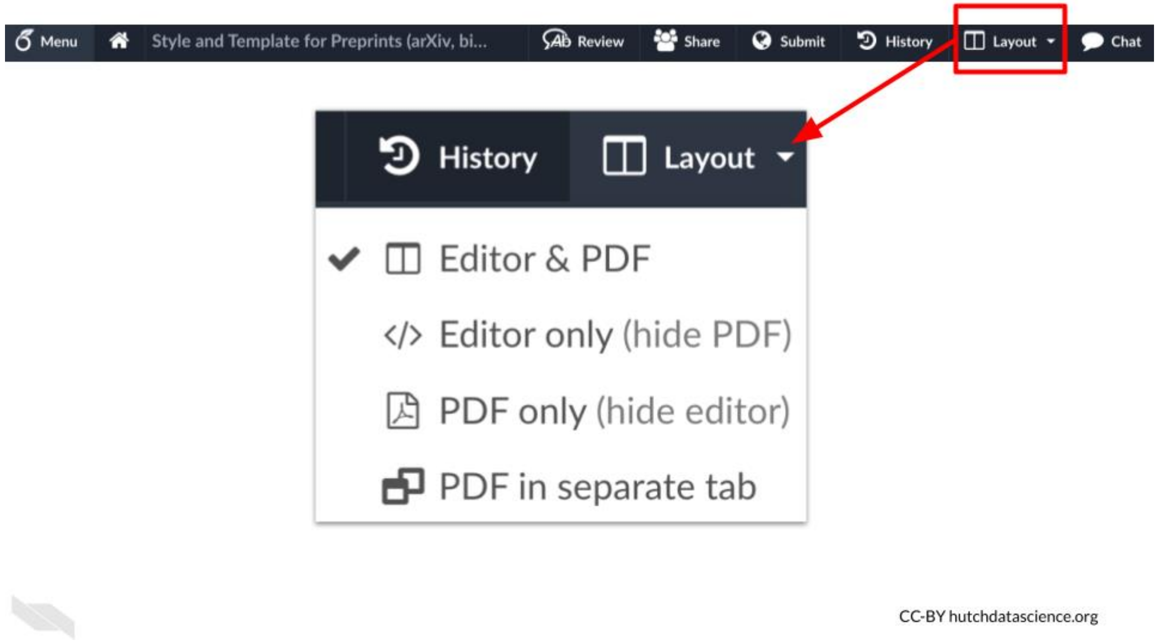
1 Introduction

Our project is a competition on Kaggle (Predict Future Sales). We are provided with daily historical sales data (including each products' sale date, block ,shop price and amount). And we will use it to forecast the total amount of each product sold next month. Because of the list of shops and products slightly changes every month. We need to create a robust model that can handle such situations.

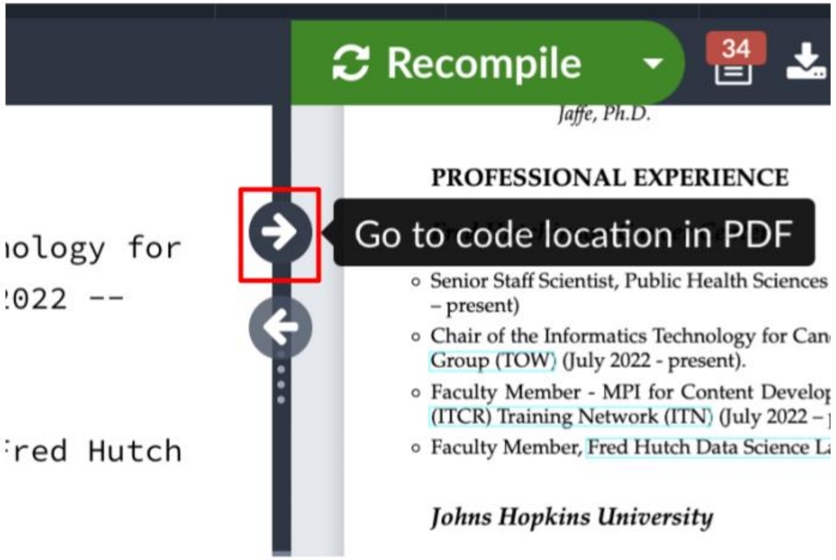
2 Task description and data construction

We are provided with five datasets from Kaggle: Sales train, Sale test, items, item categories and shops. In the Sales train dataset, it provides the information about the sales' number of an item in a shop within a day. In the Sales test dataset, it provides the shop id and item id which are the items and shops we need to predict. In the other three datasets, we can get the information about item's name and its category, and the shops' name.

CC-BY hutchdatascience.org



- You can also change the view to several other options:



```
ıology for
!022 --

red Hutch
```

Jaffe, Ph.D.

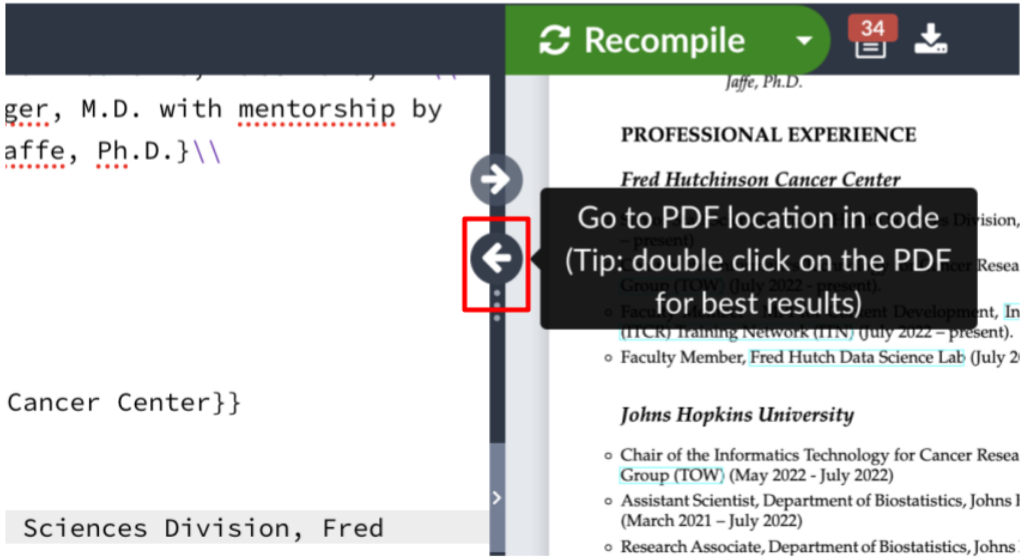
PROFESSIONAL EXPERIENCE

- Senior Staff Scientist, Public Health Sciences
  – present)
- Chair of the Informatics Technology for Can
  Group (TOW) (July 2022 - present).
- Faculty Member - MPI for Content Develop
  (ITCR) Training Network (ITN) (July 2022 –
- Faculty Member, Fred Hutch Data Science L

Johns Hopkins University

CC-BY hutchdatascience.org

- If you are using the default view and you want to see what the file looks like in the rendered form for the same location as that of the source text of the editor, you can click on the arrow (with the arrowhead facing to the right) in between.



```
ger, M.D. with mentorship by
affe, Ph.D.}\\

Cancer Center}}

 Sciences Division, Fred
```

Go to PDF location in code
(Tip: double click on the PDF
for best results)

Jaffe, Ph.D.

PROFESSIONAL EXPERIENCE

Fred Hutchinson Cancer Center

Division,
– present)
cer Resea
Group (TOW) (July 2022 – present).
Facu ent Development, In
(ITCR) Training Network (ITN) (July 2022 – present).

- Faculty Member, Fred Hutch Data Science Lab (July 2

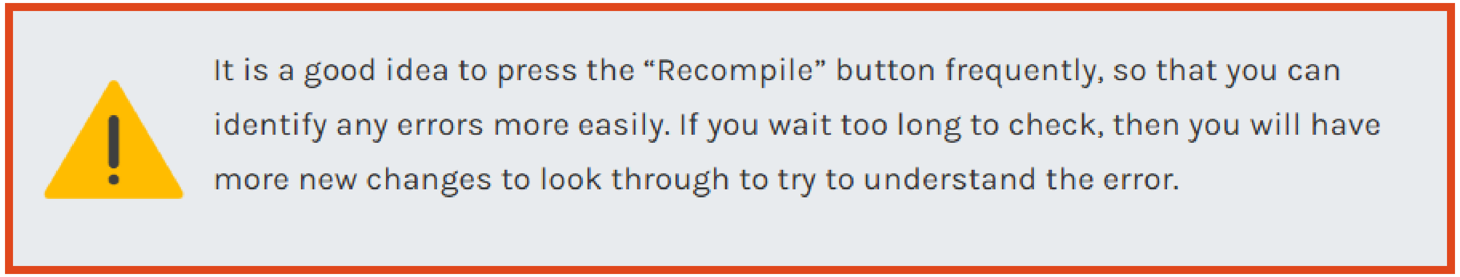
Johns Hopkins University

- Chair of the Informatics Technology for Cancer Resea Group (TOW) (May 2022 - July 2022)
- Assistant Scientist, Department of Biostatistics, Johns I (March 2021 – July 2022)
- Research Associate, Department of Biostatistics, Johns

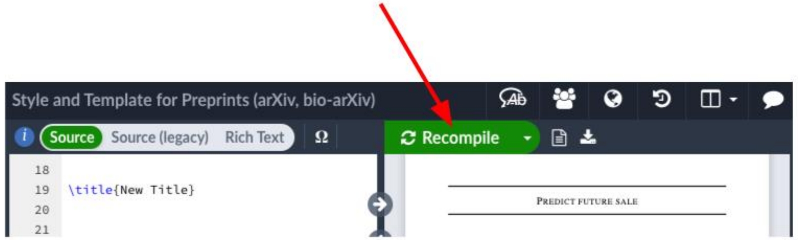
CC-BY hutchdatascience.org

- To do the opposite and go to the location in the source text where you are currently viewing in the rendered text, click the arrow button facing the source text editor.

- If you make a change in the source and want to see how it changes the look of the file, you need to press the “Recompile” button.



It is a good idea to press the “Recompile” button frequently, so that you can identify any errors more easily. If you wait too long to check, then you will have more new changes to look through to try to understand the error.



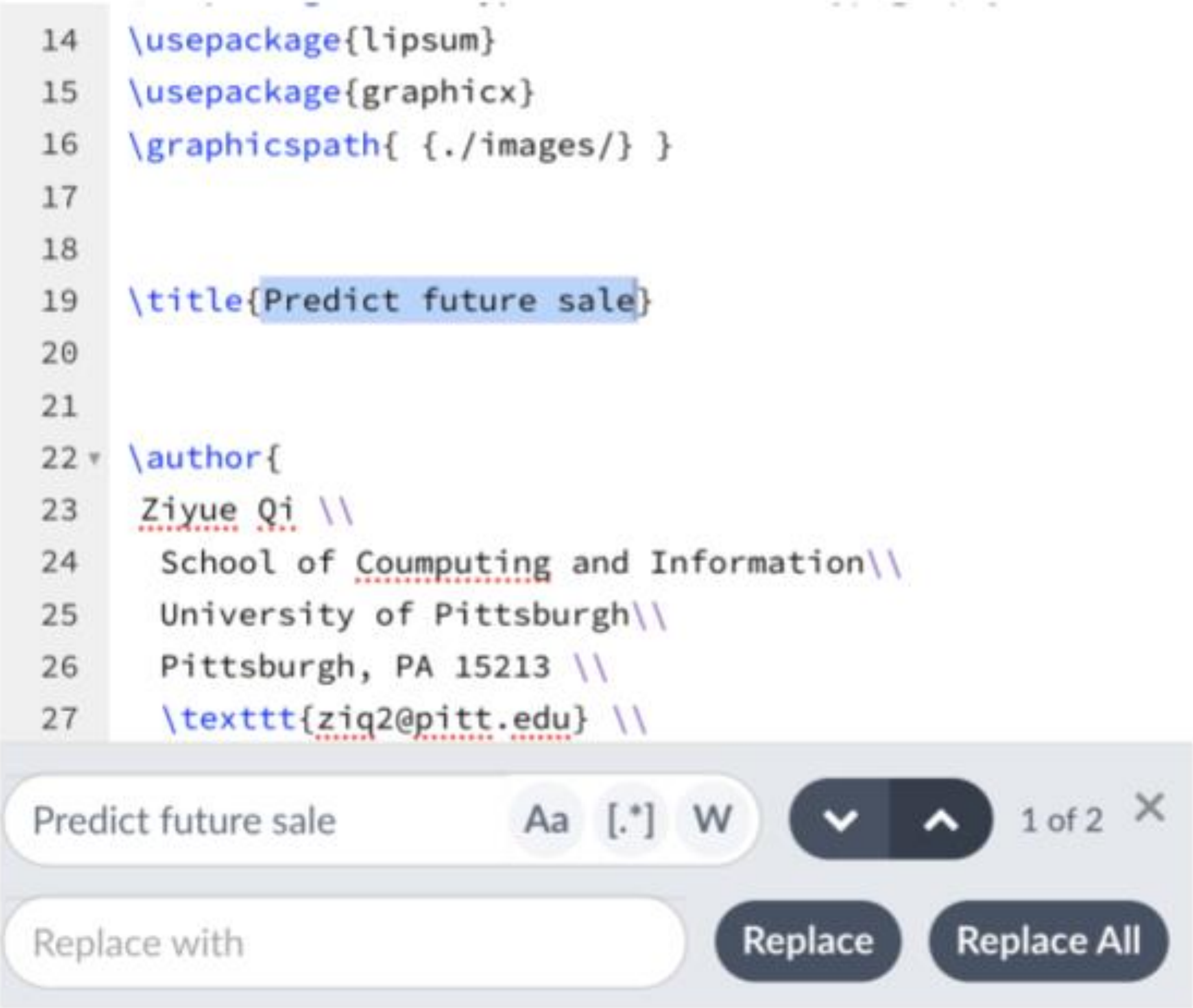
Style and Template for Preprints (arXiv, bio-arXiv)

```
\title{New Title}
```

Making changes

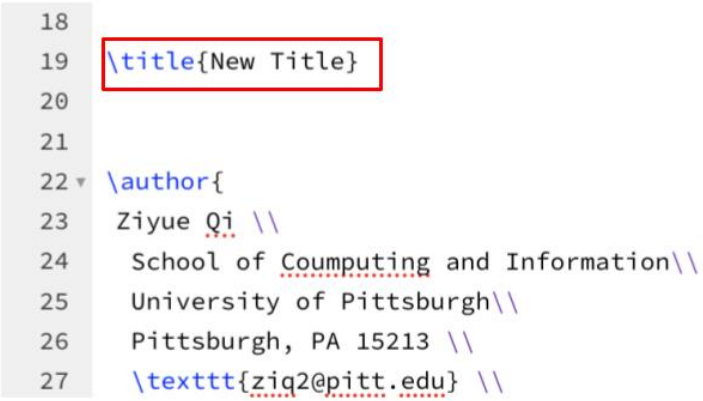
Now let’s try a simple change to start getting used to writing with LaTeX.

First we will change the title, which is currently “Predict future sale”. We can search the source editor text by clicking on the editor pane and using the search and find keyboard shortcut (control and the F key). If we type “Predict future sale” in the text box, we find it on line 19 following \title



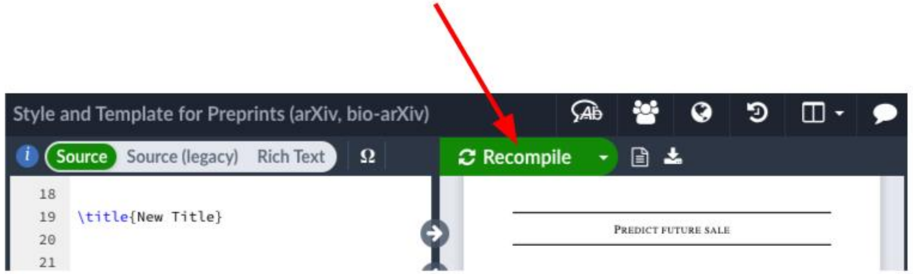
```
\usepackage{lipsum}
\usepackage{graphicx}
\graphicspath{ {./images/} }


\title{Predict future sale}


\author{
 Ziyue Qi \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{ziq2@pitt.edu} \\
```

Predict future sale



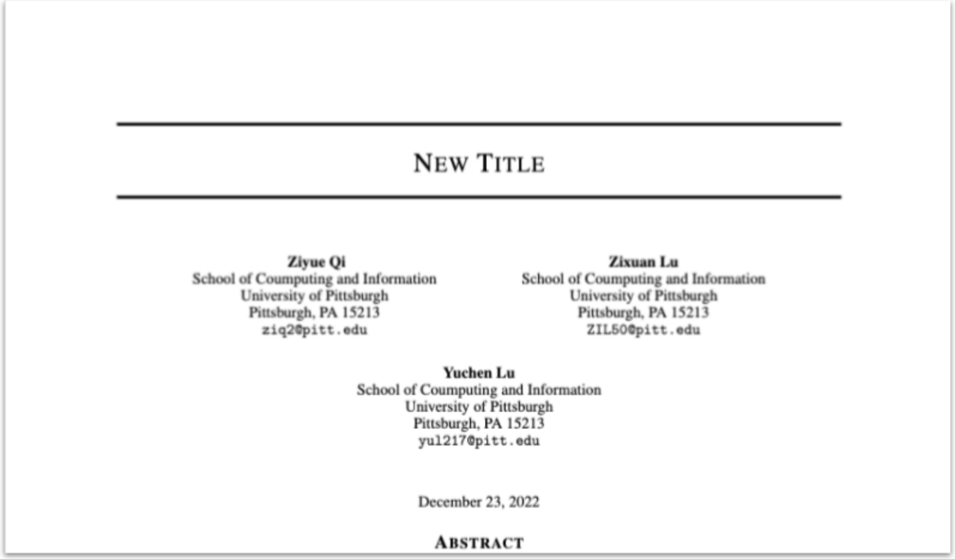
```

\title{New Title}


\author{
 Ziyue Qi \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{ziq2@pitt.edu} \\
```

- We can now replace the text with our own title. Be careful to make sure that the brackets are closed around the replacement text. Here we replace it with “New Title.”



```
\title{New Title}
```

- Now we can click the “Recompile button” to see the results.



NEW TITLE

Ziyue Qi
School of Coumputing and Information
University of Pittsburgh
Pittsburgh, PA 15213
`ziq2@pitt.edu`

Zixuan Lu
School of Coumputing and Information
University of Pittsburgh
Pittsburgh, PA 15213
`ZIL50@pitt.edu`

Yuchen Lu
School of Coumputing and Information
University of Pittsburgh
Pittsburgh, PA 15213
`yul217@pitt.edu`

December 23, 2022

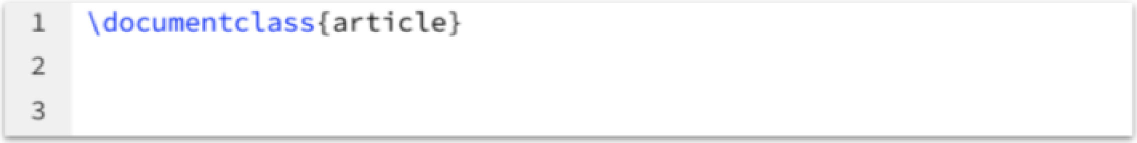
ABSTRACT

- After overleaf has finished compiling, you can see that the title has been changed.

Writing with Overleaf

- **Document Class**

At the top of the template you will notice `\documentclass{article}` .

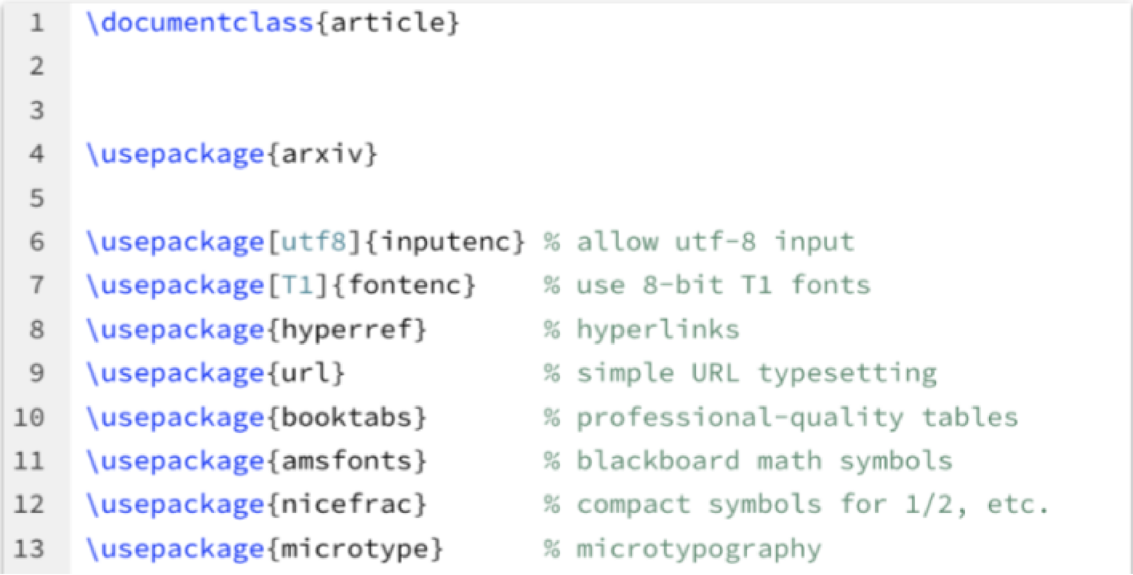


```
\documentclass{article}


```

This specifies general typesetting information about the type of document that we intend to make. For example, it often specifies font size, the overall layout of the text, and alignment of various features of the text. Since we are writing a scientific article, the specification here is article.

Packages

Next you will see that `\usepackage{}` is repeated several times with different information in the brackets:



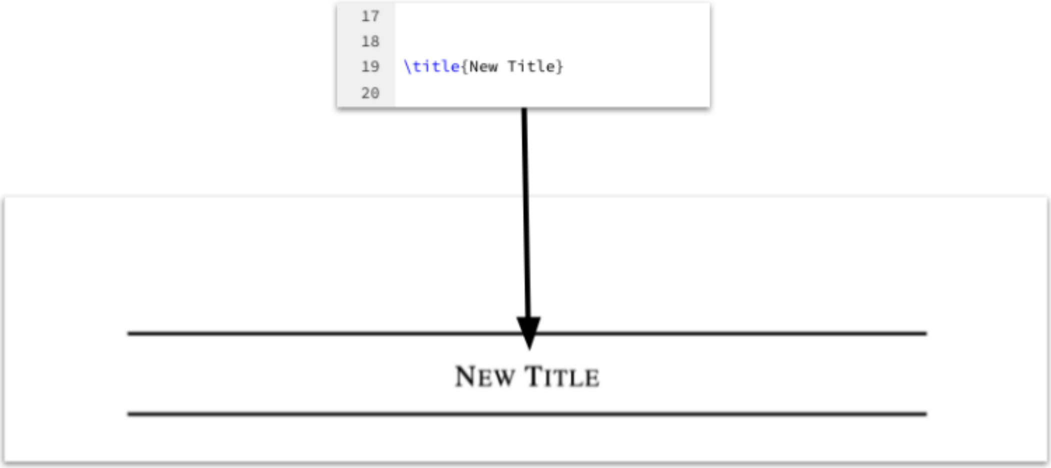
```
\documentclass{article}


\usepackage{arxiv}

\usepackage[utf8]{inputenc} % allow utf-8 input
\usepackage[T1]{fontenc}    % use 8-bit T1 fonts
\usepackage{hyperref}       % hyperlinks
\usepackage{url}            % simple URL typesetting
\usepackage{booktabs}       % professional-quality tables
\usepackage{amsfonts}       % blackboard math symbols
\usepackage{nicefrac}       % compact symbols for 1/2, etc.
\usepackage{microtype}      % microtypography
```

We will refer to these tags with brackets such as `\usepackage{}` as **commands** from now on (as this is what the are generally referred to) and they cause a change to either the text within the brackets or the overall document.

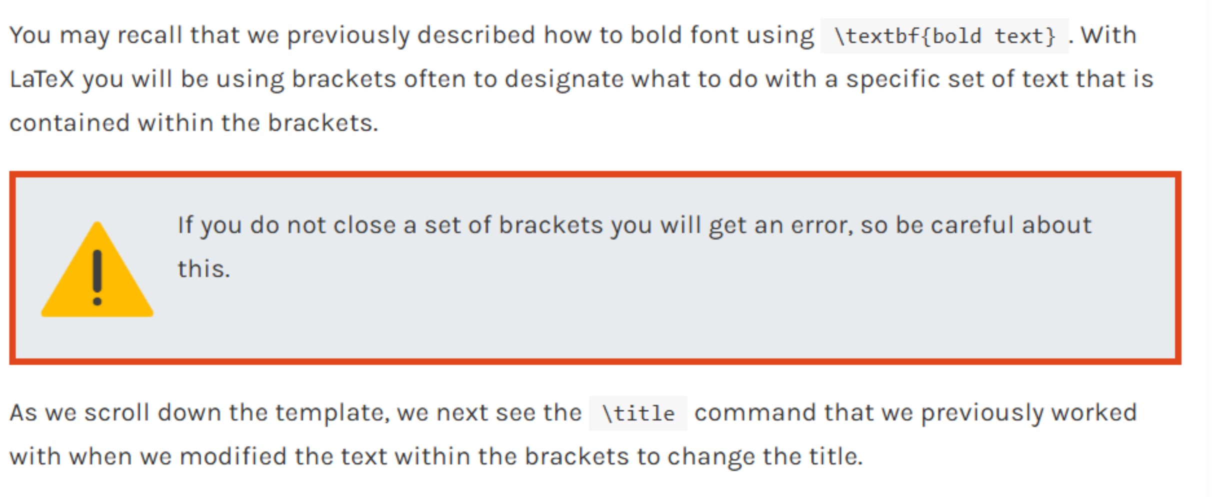
The `\usepackage{}` command installs packages, which are collections of code that help you do additional things with your documents. Packages need to be installed before commands from those packages can be used. Commands from these packages will be utilized later in the template. **It is recommended that you leave this code as is, and only modify the rest of the template until you learn more.**

In addition to determining what commands you can use, packages will also determine how the content is formatted or laid out.

Author section



```latex
\title{New Title}
```

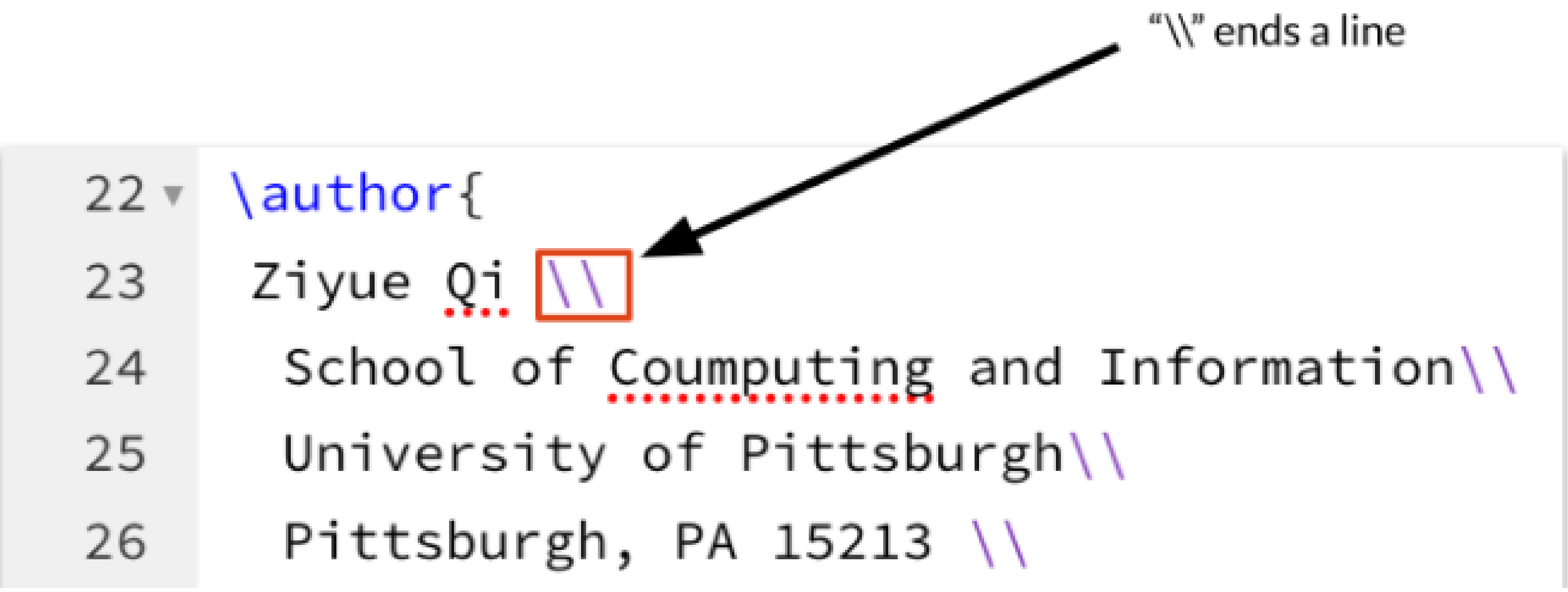


You may recall that we previously described how to bold font using `\textbf{bold text}`. With LaTeX you will be using brackets often to designate what to do with a specific set of text that is contained within the brackets.

⚠ If you do not close a set of brackets you will get an error, so be careful about this.

As we scroll down the template, we next see the `\title` command that we previously worked with when we modified the text within the brackets to change the title.

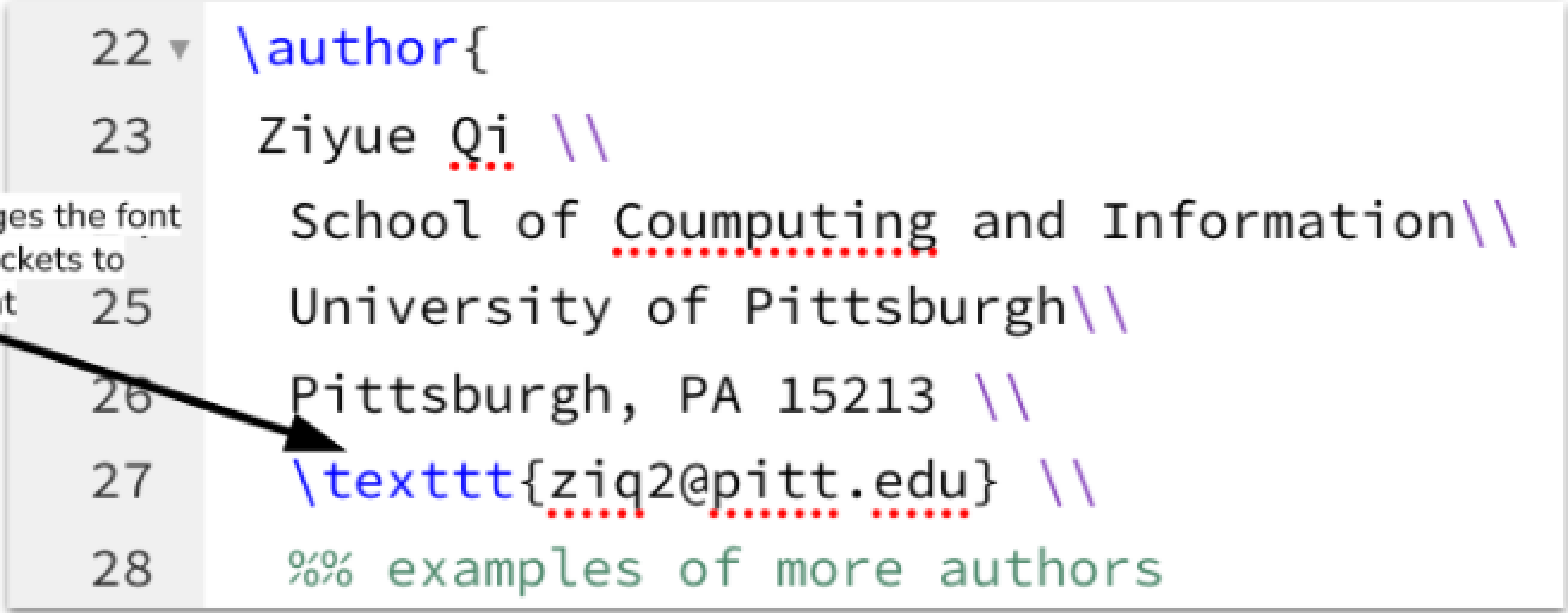
Then we see the `\author` command, which adds authors to the paper. These will be formatted in the way that is shown on the template. When you see `\\` two backslashes, this indicates that the line is finished and a new one is to be made. For tables or formatting like the authors, it should work well, but it is best to **avoid** using this for line breaks within the paragraphs that you might include in the paper.



“\\” ends a line

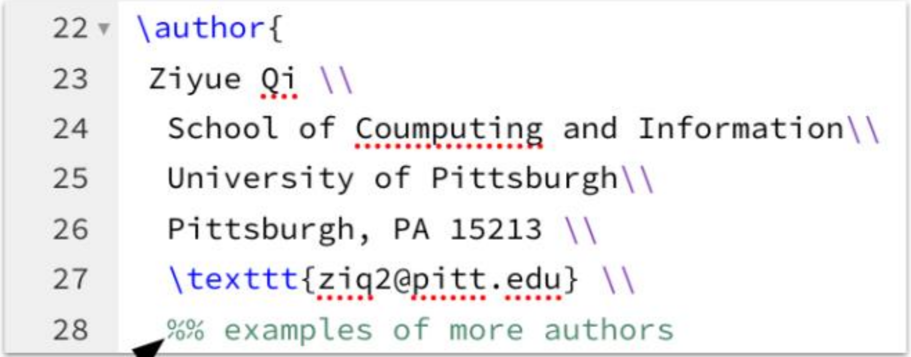
```
\author{
 Ziyue Qi \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
```

We also see another command `\textttt{}` used within the `\author{}` command to change the text to typewriter font.



```
\author{
 Ziyue Qi \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{ziq2@pitt.edu} \\
  %% examples of more authors
```

“\texttt” changes the font within the brackets to typewriter font

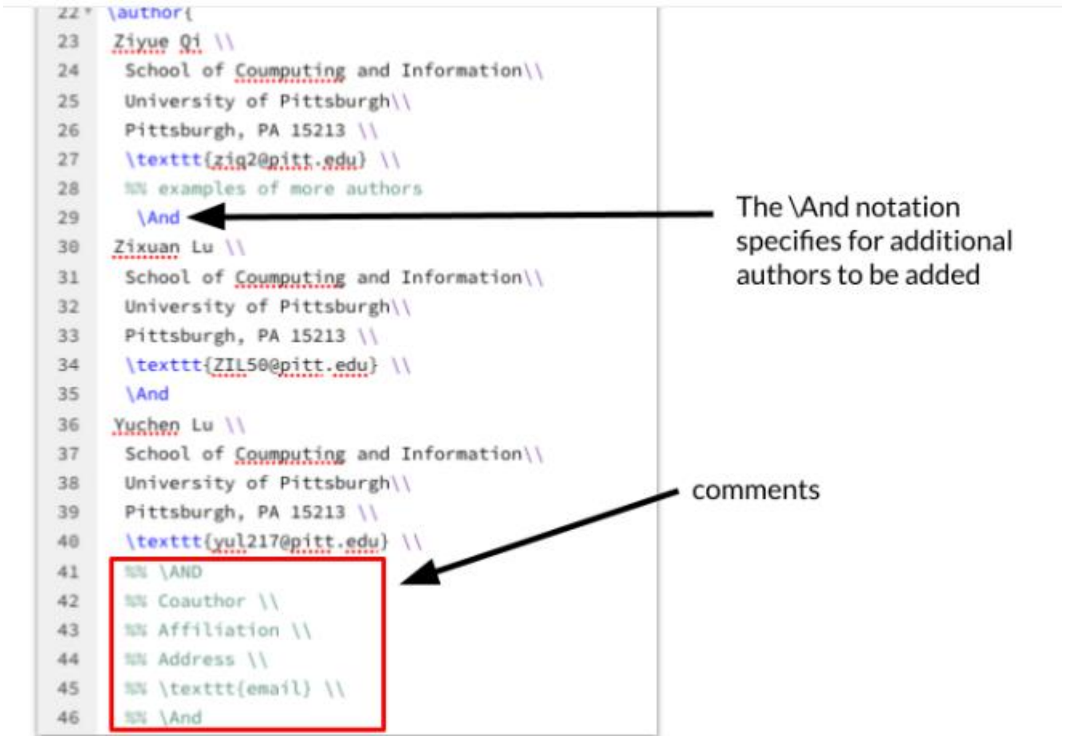


```latex
\author{
 Ziyue Qi \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{ziq2@pitt.edu} \\
  %% examples of more authors
```

“%%” creates a comment to add a note that will not show up in the final document

You may also notice `%% examples of other authors` is in green and does not show up in the rendered document. This is what is called a **comment** and it can be used to write notes about the material.

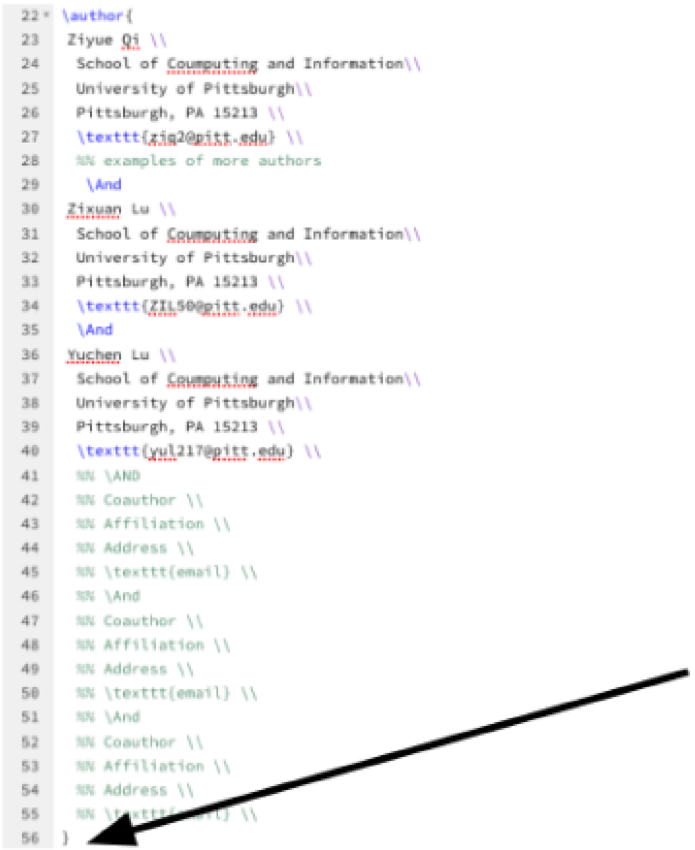
In the author section, the `\And` allows for additional authors to be added. It must be used between each author listing.



```
\author{
Ziyue Qi \\
 School of Coumputing and Information\\
 University of Pittsburgh\\
 Pittsburgh, PA 15213 \\
 \texttt{ziq2@pitt.edu} \\
 %% examples of more authors
  \And
Zixuan Lu \\
 School of Coumputing and Information\\
 University of Pittsburgh\\
 Pittsburgh, PA 15213 \\
 \texttt{ZIL50@pitt.edu} \\
 \And
Yuchen Lu \\
 School of Coumputing and Information\\
 University of Pittsburgh\\
 Pittsburgh, PA 15213 \\
 \texttt{yul217@pitt.edu} \\
 %% \AND
 %% Coauthor \\
 %% Affiliation \\
 %% Address \\
 %% \texttt{email} \\
 %% \And
```

The \And notation specifies for additional authors to be added

comments

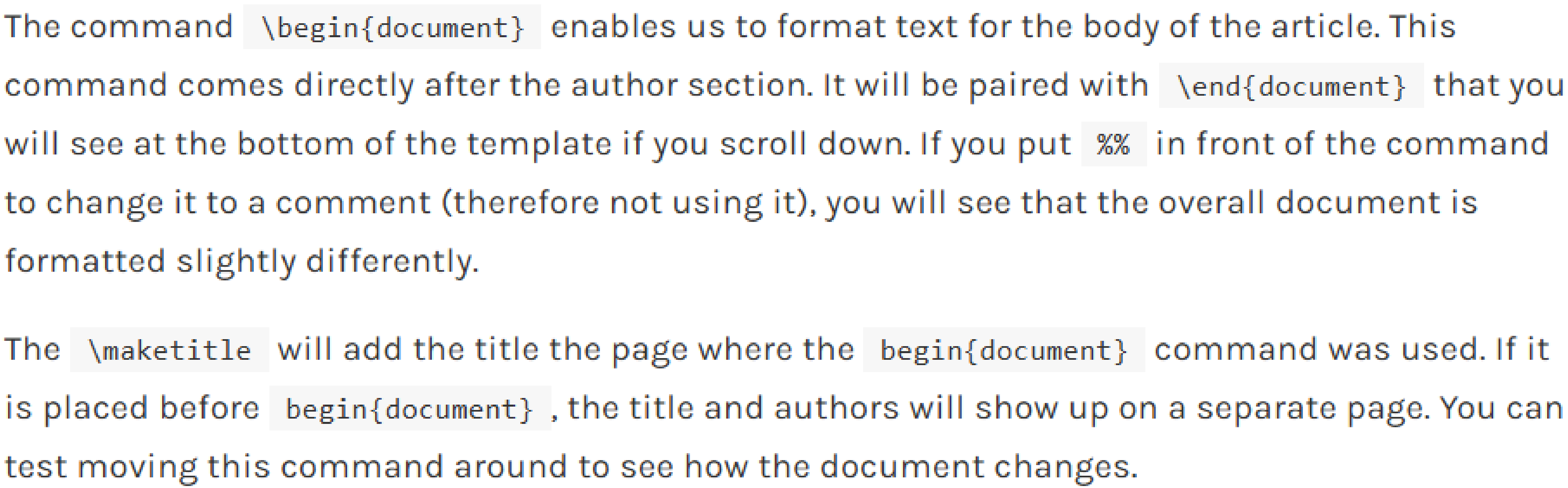


```
\author{
 Ziyue Qi \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{ziq2@pitt.edu} \\
  %% examples of more authors
   \And
 Zixuan Lu \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{ZIL50@pitt.edu} \\
  \And
 Yuchen Lu \\
  School of Coumputing and Information\\
  University of Pittsburgh\\
  Pittsburgh, PA 15213 \\
  \texttt{yul217@pitt.edu} \\
  %% \AND
  %% Coauthor \\
  %% Affiliation \\
  %% Address \\
  %% \texttt{email} \\
  %% \And
  %% Coauthor \\
  %% Affiliation \\
  %% Address \\
  %% \texttt{email} \\
  %% \And
  %% Coauthor \\
  %% Affiliation \\
  %% Address \\
  %% \texttt{email} \\
}
```

Need to close the original brackets from the author command

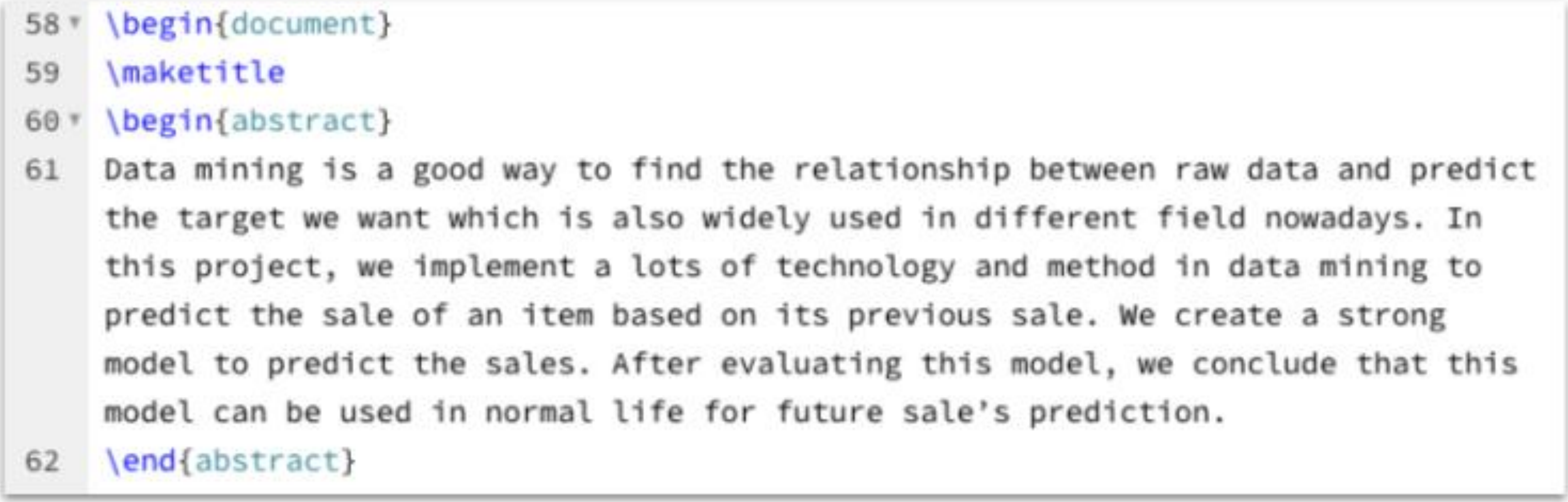
- Finally, the author section needs to be completed by closing the brackets.

Formatting the Document Body



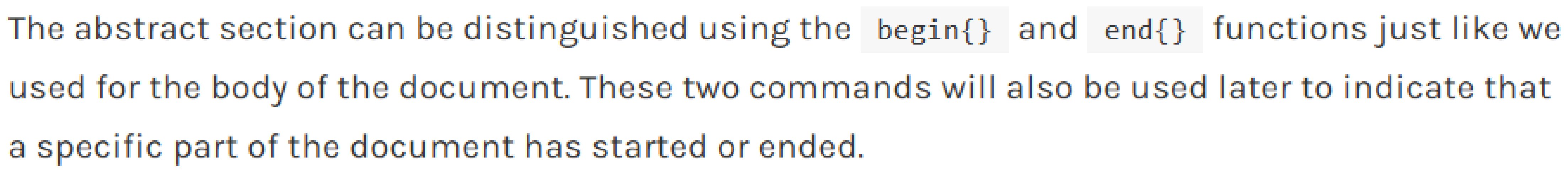
The command `\begin{document}` enables us to format text for the body of the article. This command comes directly after the author section. It will be paired with `\end{document}` that you will see at the bottom of the template if you scroll down. If you put `%%` in front of the command to change it to a comment (therefore not using it), you will see that the overall document is formatted slightly differently.

The `\maketitle` will add the title the page where the `begin{document}` command was used. If it is placed before `begin{document}`, the title and authors will show up on a separate page. You can test moving this command around to see how the document changes.

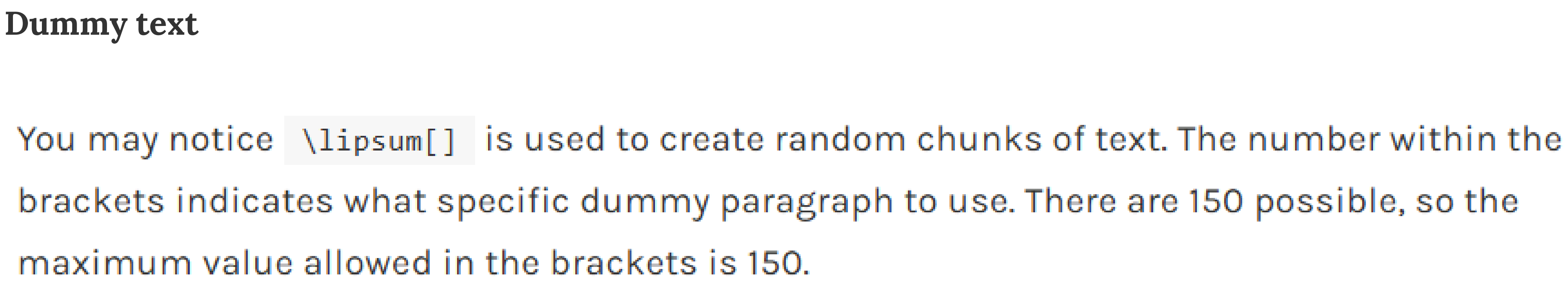


```latex
\begin{document}
\maketitle
\begin{abstract}
Data mining is a good way to find the relationship between raw data and predict
the target we want which is also widely used in different field nowadays. In
this project, we implement a lots of technology and method in data mining to
predict the sale of an item based on its previous sale. We create a strong
model to predict the sales. After evaluating this model, we conclude that this
model can be used in normal life for future sale’s prediction.
\end{abstract}
```

Abstract



The abstract section can be distinguished using the `begin{}` and `end{}` functions just like we used for the body of the document. These two commands will also be used later to indicate that a specific part of the document has started or ended.



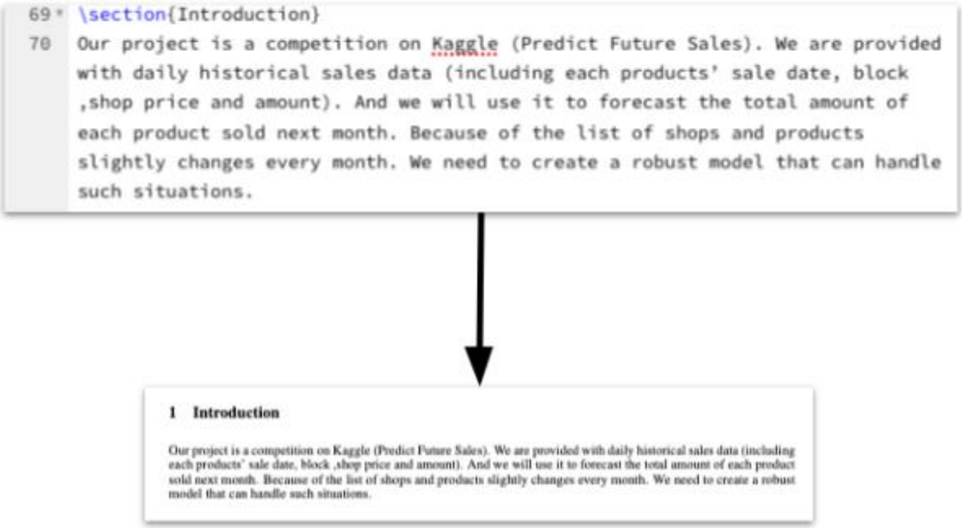
Dummy text

You may notice `\lipsum[]` is used to create random chunks of text. The number within the brackets indicates what specific dummy paragraph to use. There are 150 possible, so the maximum value allowed in the brackets is 150.

Sections

- `\section{section name}` - This will help you to create sections in the template. We don’t need to do anything to modify the text, it will automatically bold the text and number the sections (1, 2, 3 etc.).
- `\subsection{subsubsection name}` - This will help you to create subsections. These headings will be one level down from the section headings and will be numbered like 1.1, 1.2.
- `\subsubsection{subsubsection name}` - This will help you to create sections one level down from subsections. These heading will be numbered like 1.1.1, 1.1.2.

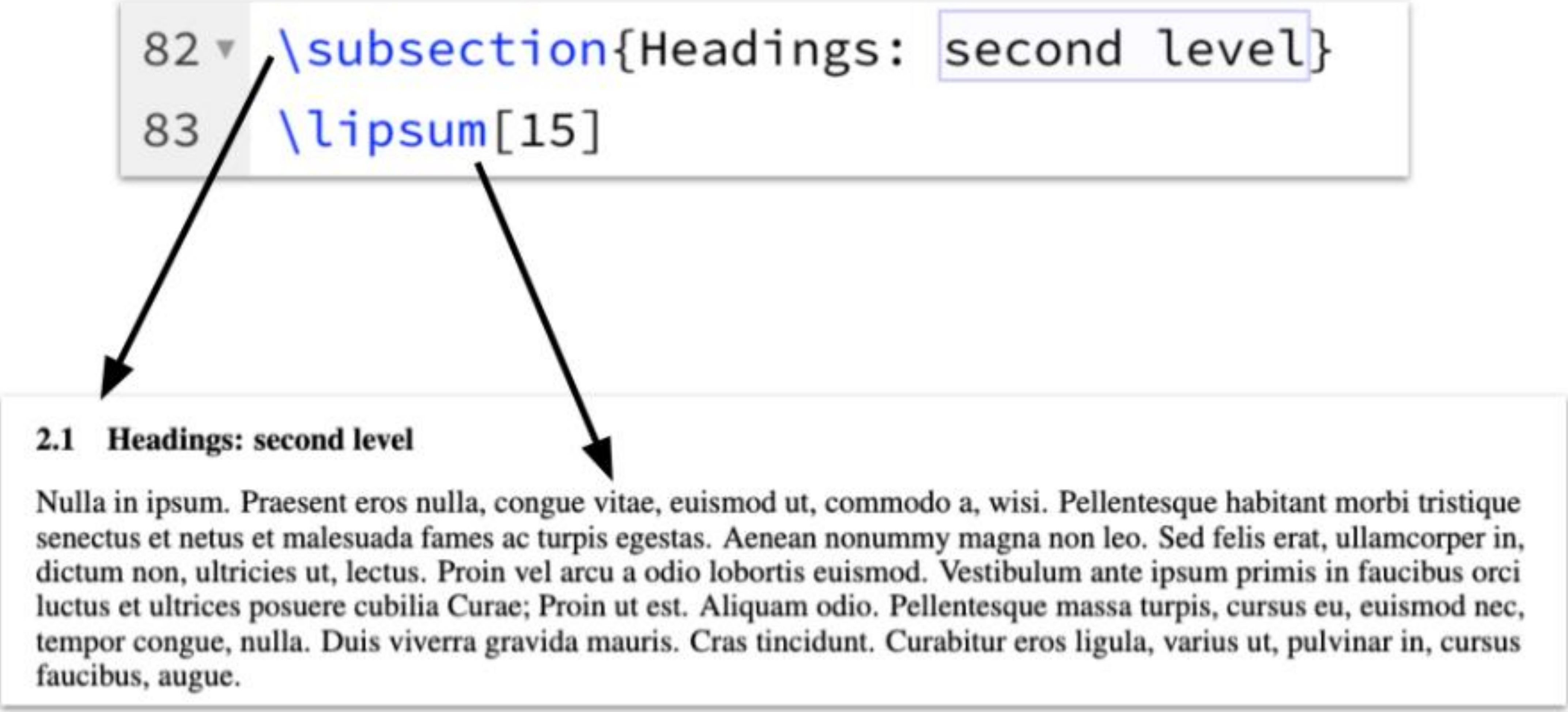
In the template you can see how these are formatted:



```
\section{Introduction}
Our project is a competition on Kaggle (Predict Future Sales). We are provided
with daily historical sales data (including each products’ sale date, block
,shop price and amount). And we will use it to forecast the total amount of
each product sold next month. Because of the list of shops and products
slightly changes every month. We need to create a robust model that can handle
such situations.
```

1 **Introduction**

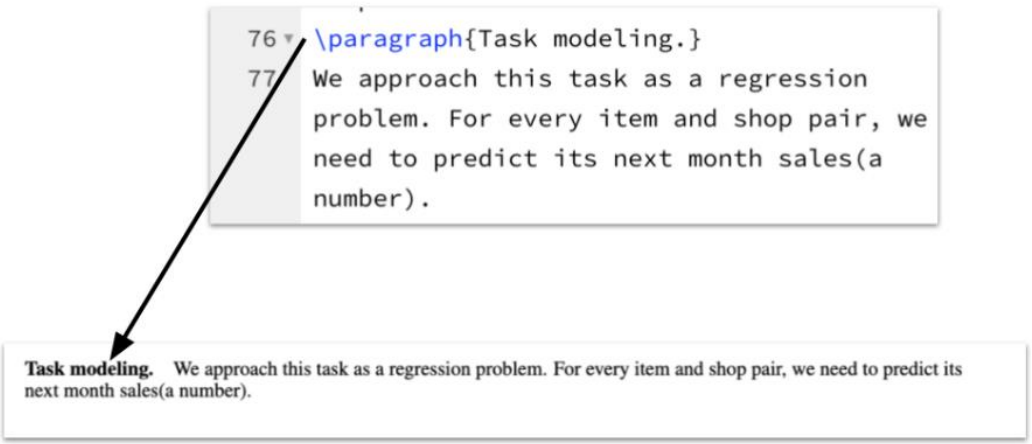
Our project is a competition on Kaggle (Predict Future Sales). We are provided with daily historical sales data (including each products’ sale date, block ,shop price and amount). And we will use it to forecast the total amount of each product sold next month. Because of the list of shops and products slightly changes every month. We need to create a robust model that can handle such situations.



```
\subsection{Headings: second level}
\lipsum[15]
```

2.1 Headings: second level

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.



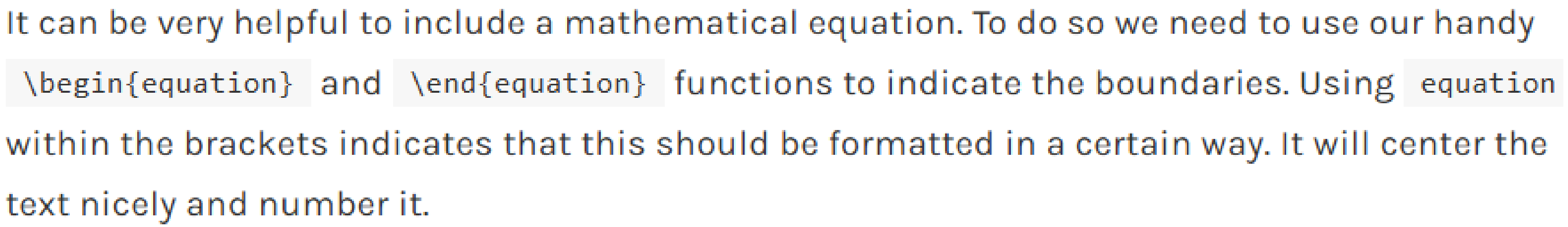
```
\paragraph{Task modeling.}
We approach this task as a regression
problem. For every item and shop pair, we
need to predict its next month sales(a
number).
```

Task modeling. We approach this task as a regression problem. For every item and shop pair, we need to predict its next month sales(a number).

You can specify if you want sections to be not be numbered when using an asterisks `*` between the command name and the brackets `section*{}`.

The `\paragraph{}` command works similarly, but without numbering. The text within the brackets is an optional word or phrase that will have bold font to start the paragraph. You can also leave it empty.

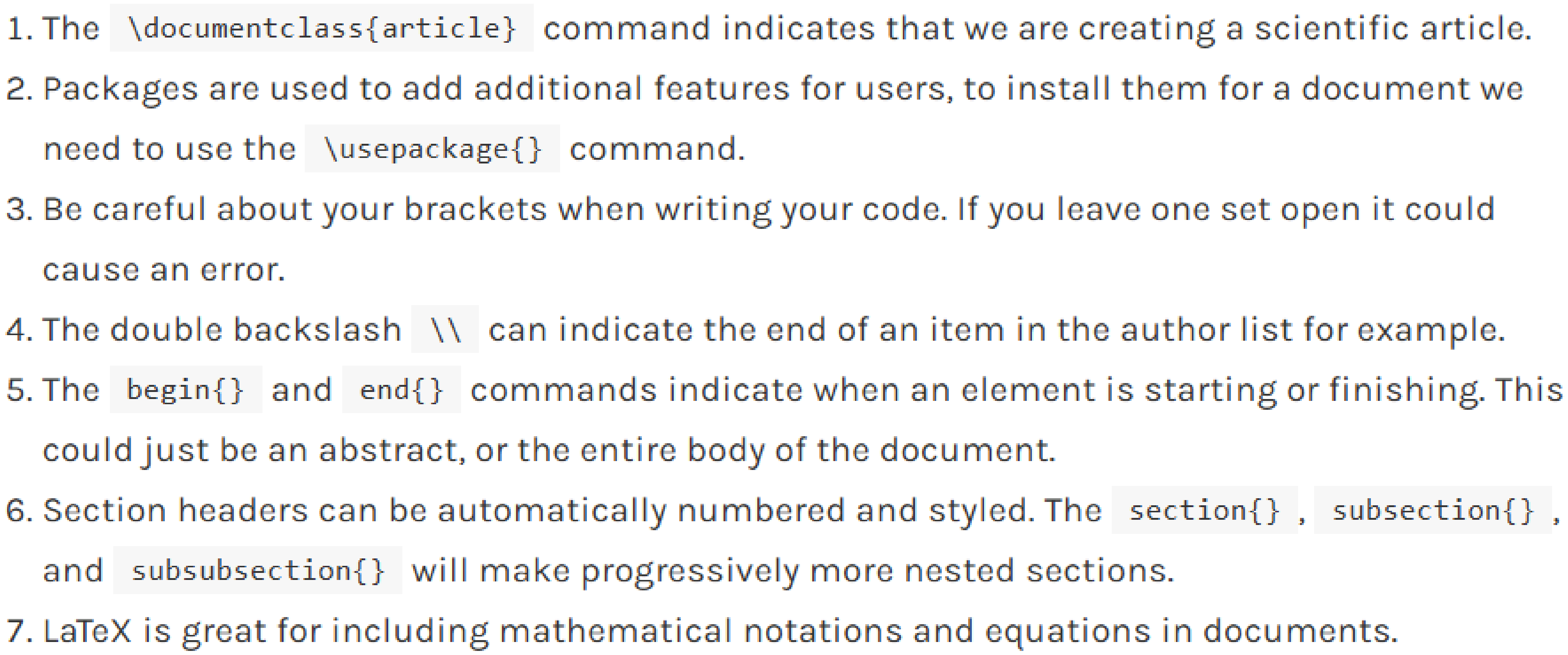
Equations



It can be very helpful to include a mathematical equation. To do so we need to use our handy `\begin{equation}` and `\end{equation}` functions to indicate the boundaries. Using `equation` within the brackets indicates that this should be formatted in a certain way. It will center the text nicely and number it.

For more information about mathematical expressions in overleaf see the Overleaf documentation.

Conclusion

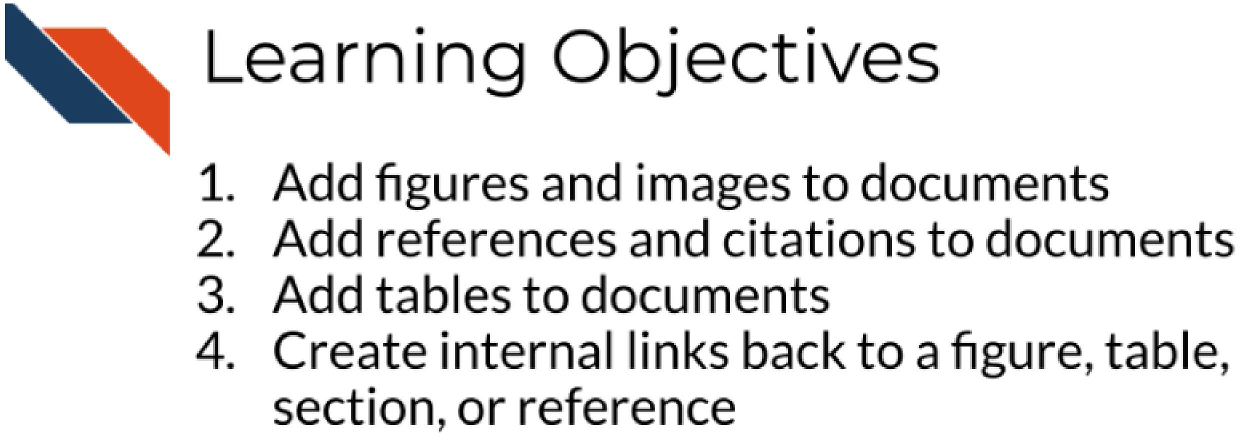


1. The `\documentclass{article}` command indicates that we are creating a scientific article.
2. Packages are used to add additional features for users, to install them for a document we need to use the `\usepackage{}` command.
3. Be careful about your brackets when writing your code. If you leave one set open it could cause an error.
4. The double backslash `\\` can indicate the end of an item in the author list for example.
5. The `begin{}` and `end{}` commands indicate when an element is starting or finishing. This could just be an abstract, or the entire body of the document.
6. Section headers can be automatically numbered and styled. The `section{}` , `subsection{}` , and `subsubsection{}` will make progressively more nested sections.
7. LaTeX is great for including mathematical notations and equations in documents.

References

- Wu, Charles. n.d. “LibGuides: LaTeX & BibTeX: LaTeX: \DocumentClass Command.” Accessed January 7, 2023. https://libguides.utsa.edu/c.php?g=522165&p=3570198.

Elements

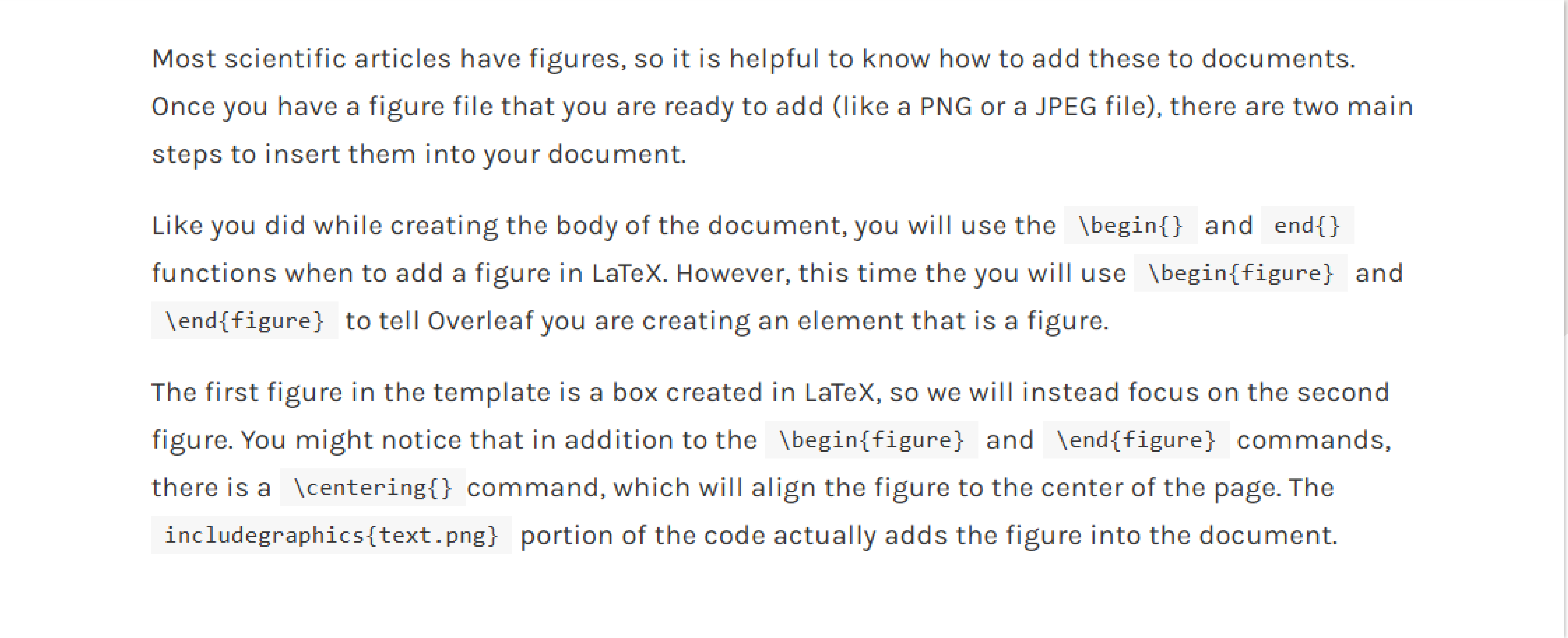


Learning Objectives

1. Add figures and images to documents
2. Add references and citations to documents
3. Add tables to documents
4. Create internal links back to a figure, table, section, or reference

You will notice that the template has a few examples of each of these elements that we will walk through now.

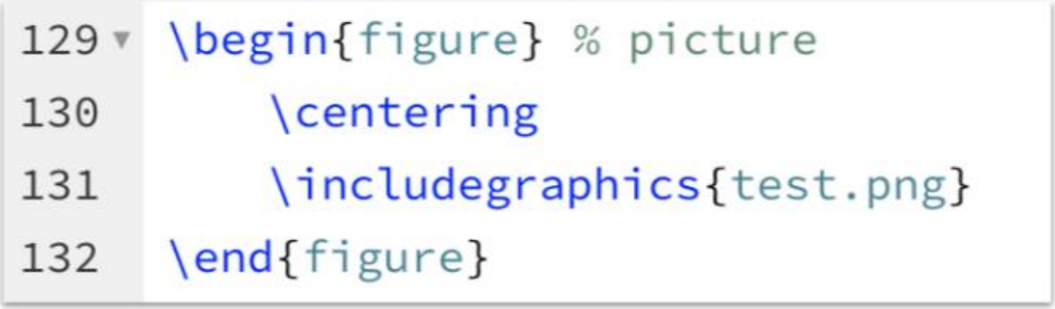
Figures and Images



Most scientific articles have figures, so it is helpful to know how to add these to documents. Once you have a figure file that you are ready to add (like a PNG or a JPEG file), there are two main steps to insert them into your document.

Like you did while creating the body of the document, you will use the `\begin{}` and `end{}` functions when to add a figure in LaTeX. However, this time the you will use `\begin{figure}` and `\end{figure}` to tell Overleaf you are creating an element that is a figure.

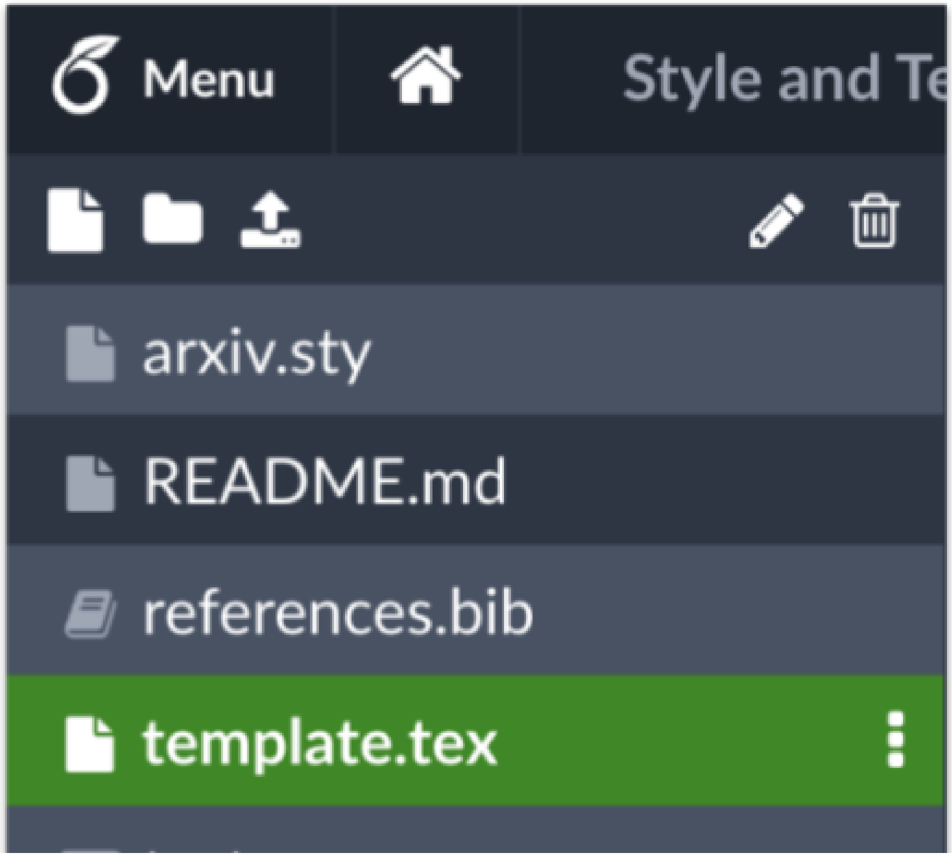
The first figure in the template is a box created in LaTeX, so we will instead focus on the second figure. You might notice that in addition to the `\begin{figure}` and `\end{figure}` commands, there is a `\centering{}` command, which will align the figure to the center of the page. The `includegraphics{text.png}` portion of the code actually adds the figure into the document.



```
\begin{figure} % picture
    \centering
    \includegraphics{test.png}
\end{figure}
```

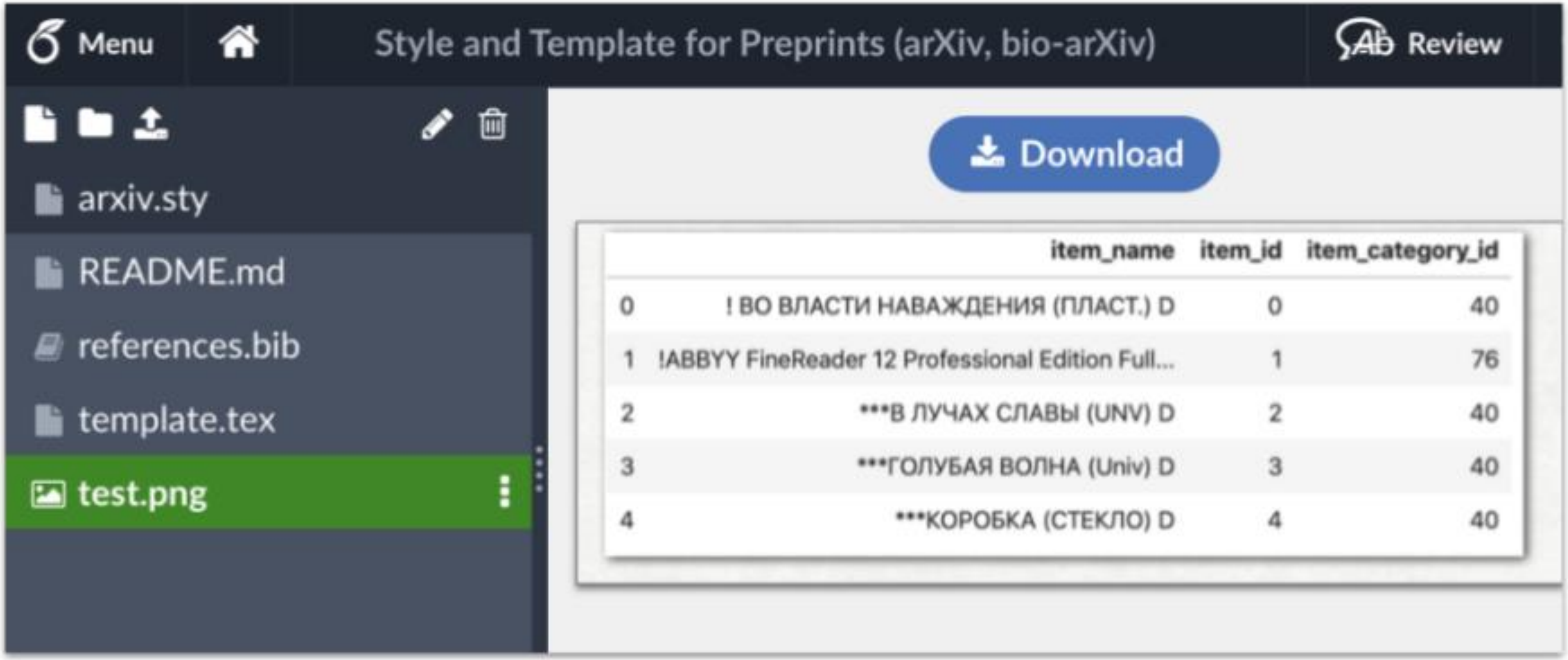
Note that comments with one percent sign can be used after code - thus the `% picture` is just telling you that this part of the code is adding a picture. People use different numbers of percent signs based on preference and convention, but just one percent sign is sufficient to turn anything following that into a comment.

You may notice on the left side of the template in Overleaf that there are a few files listed, including the name of the image file used in the code for the figure: `test.png`. The `template.tex` file is the file we have been working in.



Where we started automatically and where the main manuscript content text and typesetting code lives

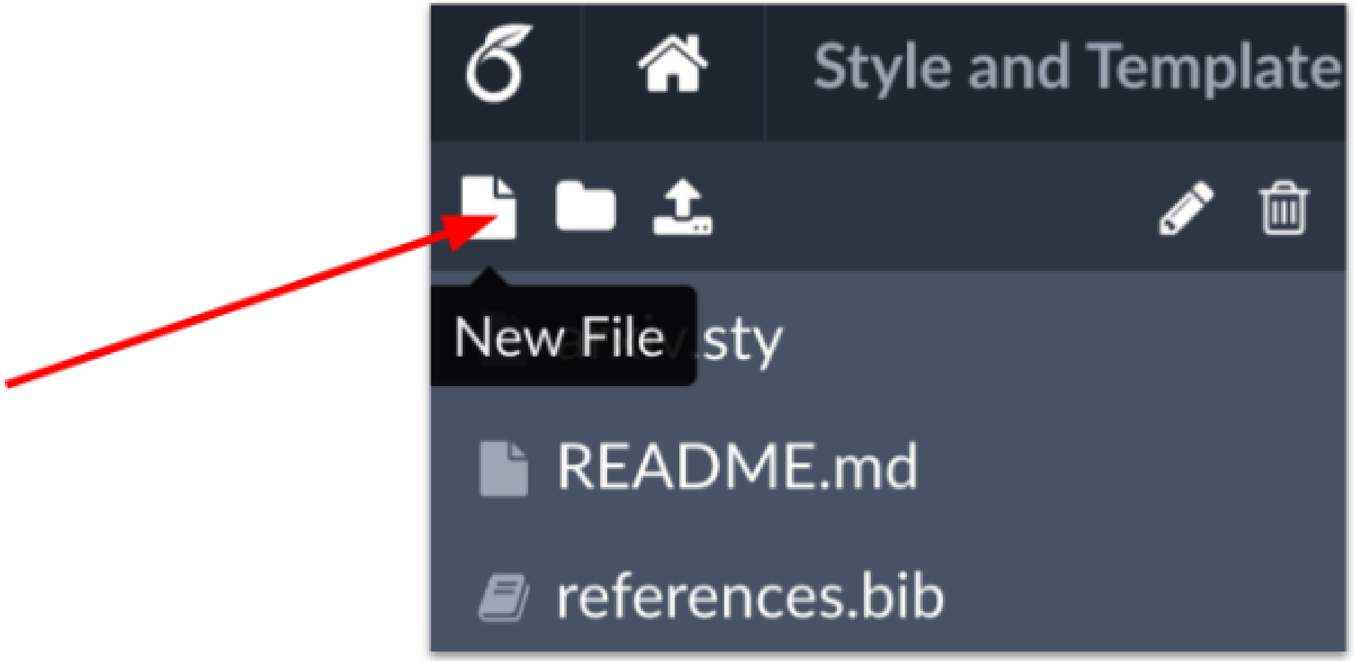
If you click on the name of the image file you will see a preview of the image.



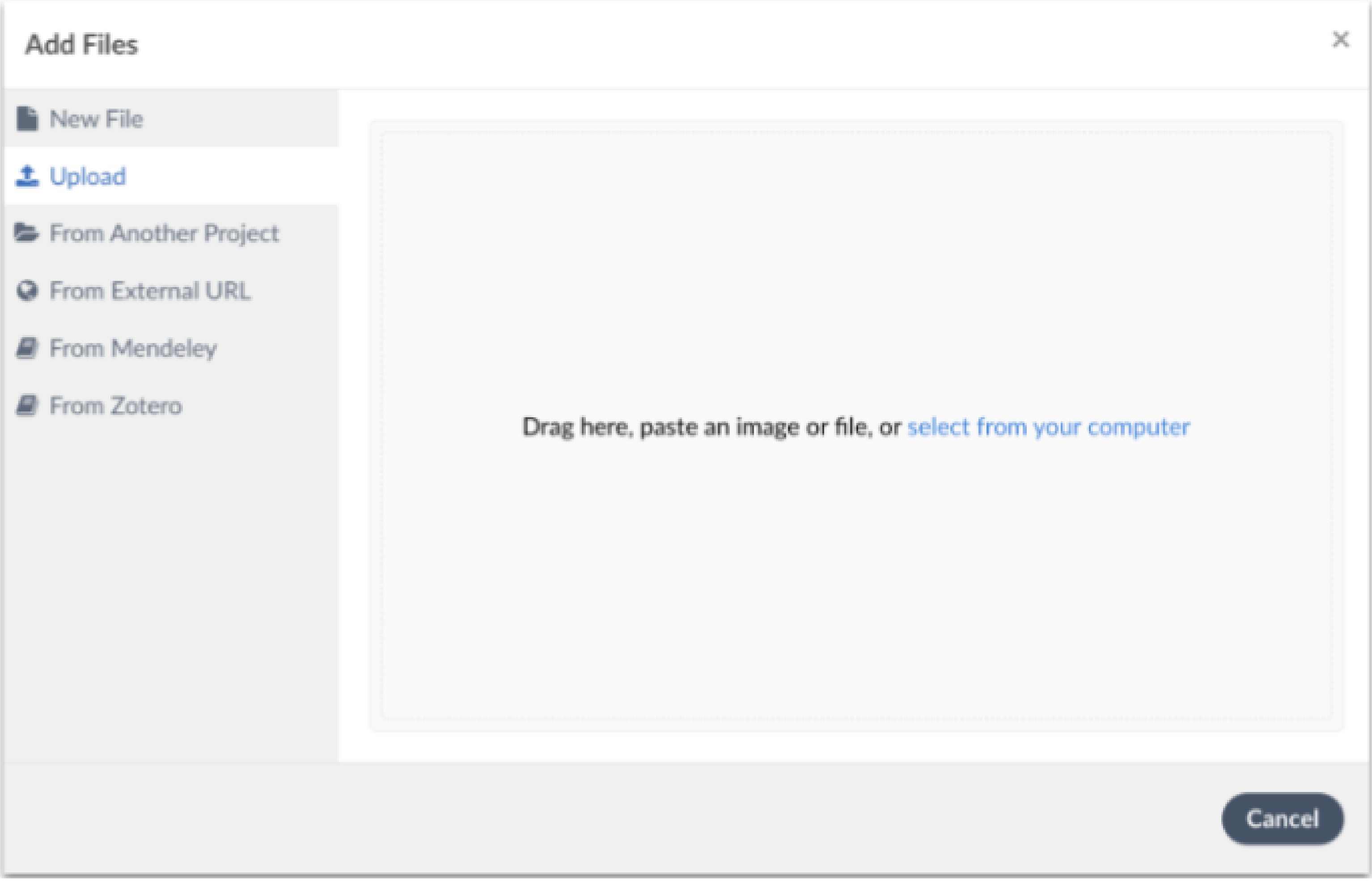
	item_name	item_id	item_category_id
0	! ВО ВЛАСТИ НАВАЖДЕНИЯ (ПЛАСТ.) D	0	40
1	!ABBYY FineReader 12 Professional Edition Full...	1	76
2	***В ЛУЧАХ СЛАВЫ (UNV) D	2	40
3	***ГОЛУБАЯ ВОЛНА (Univ) D	3	40
4	***КОРОБКА (СТЕКЛО) D	4	40

▶ **What are these other files?**

To upload an image file to add a new figure, you can click on the new file button, which is the icon that looks like a piece of paper with the right upper corner folded.

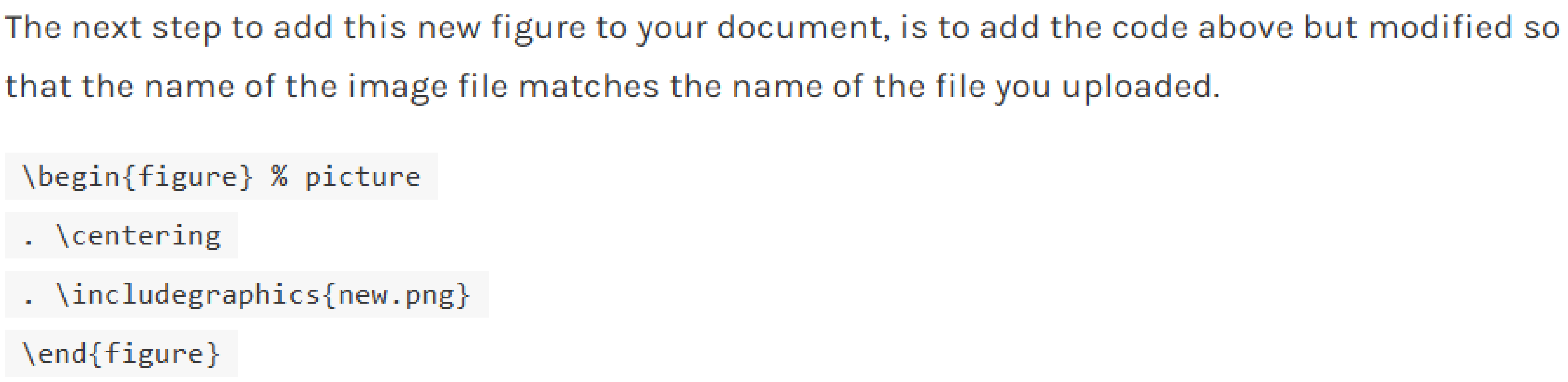


Then select the upload button to drag and drop a new image file from your computer.



Add Files

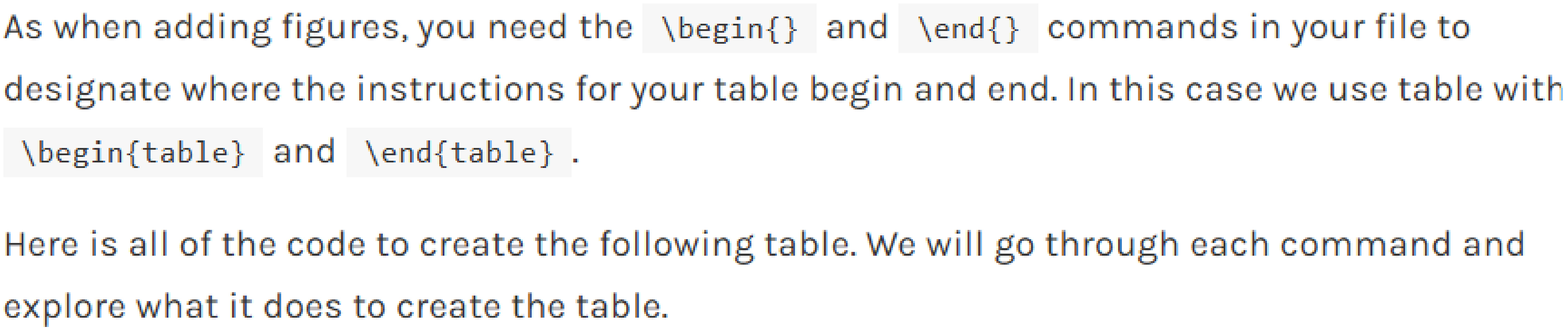
Drag here, paste an image or file, or select from your computer



The next step to add this new figure to your document, is to add the code above but modified so that the name of the image file matches the name of the file you uploaded.

```
\begin{figure} % picture
. \centering
. \includegraphics{new.png}
\end{figure}
```

Tables



As when adding figures, you need the `\begin{}` and `\end{}` commands in your file to designate where the instructions for your table begin and end. In this case we use table with `\begin{table}` and `\end{table}`.

Here is all of the code to create the following table. We will go through each command and explore what it does to create the table.

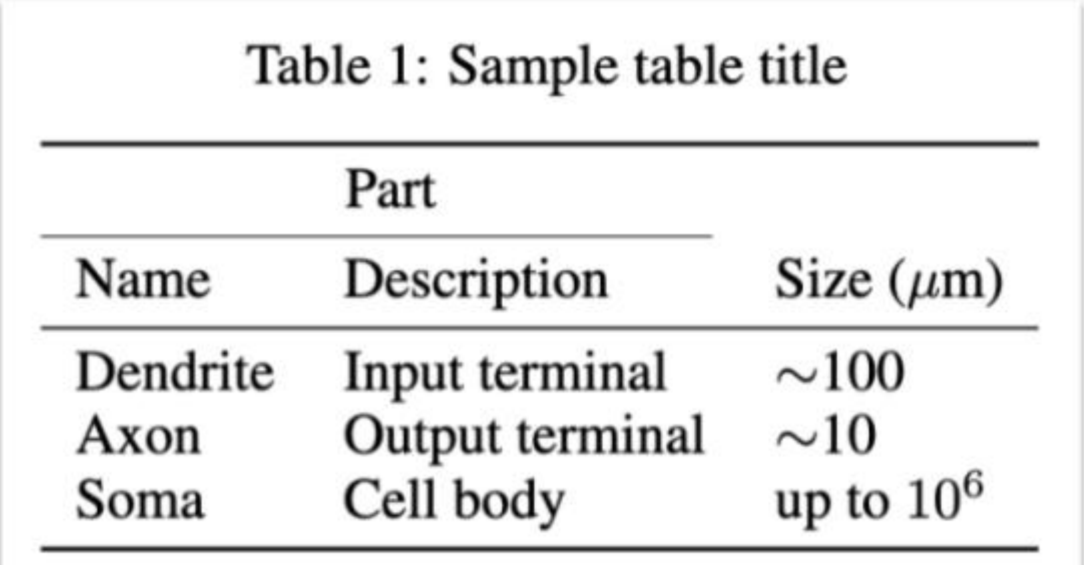
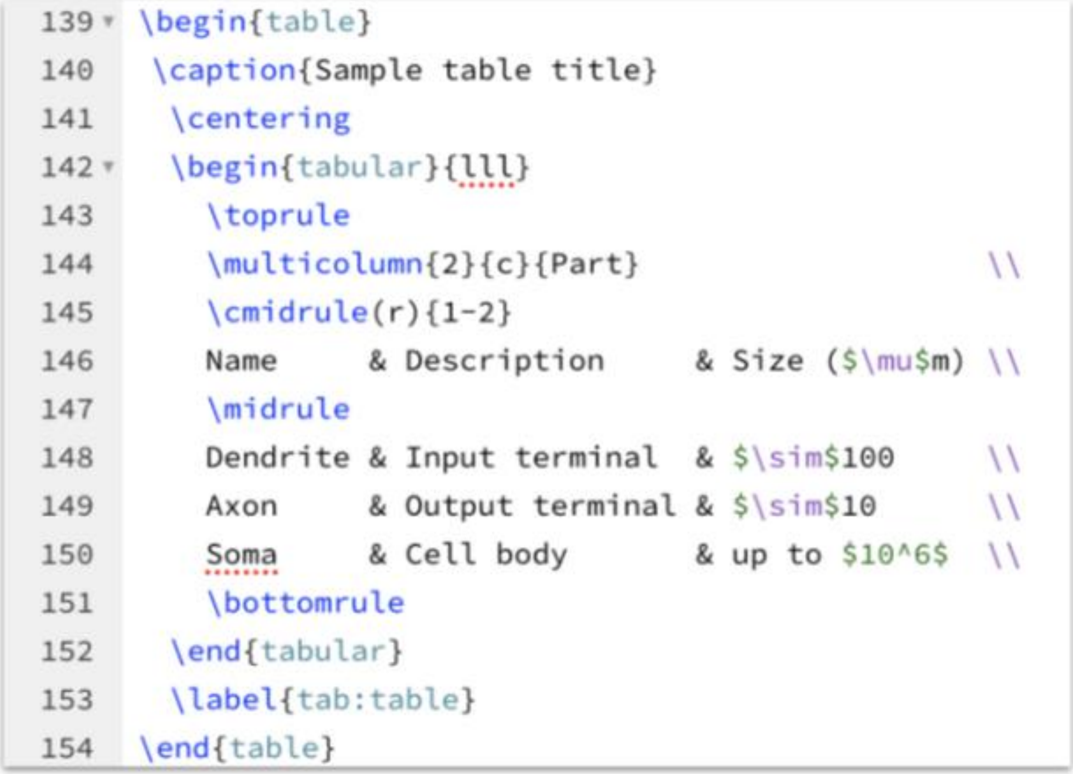


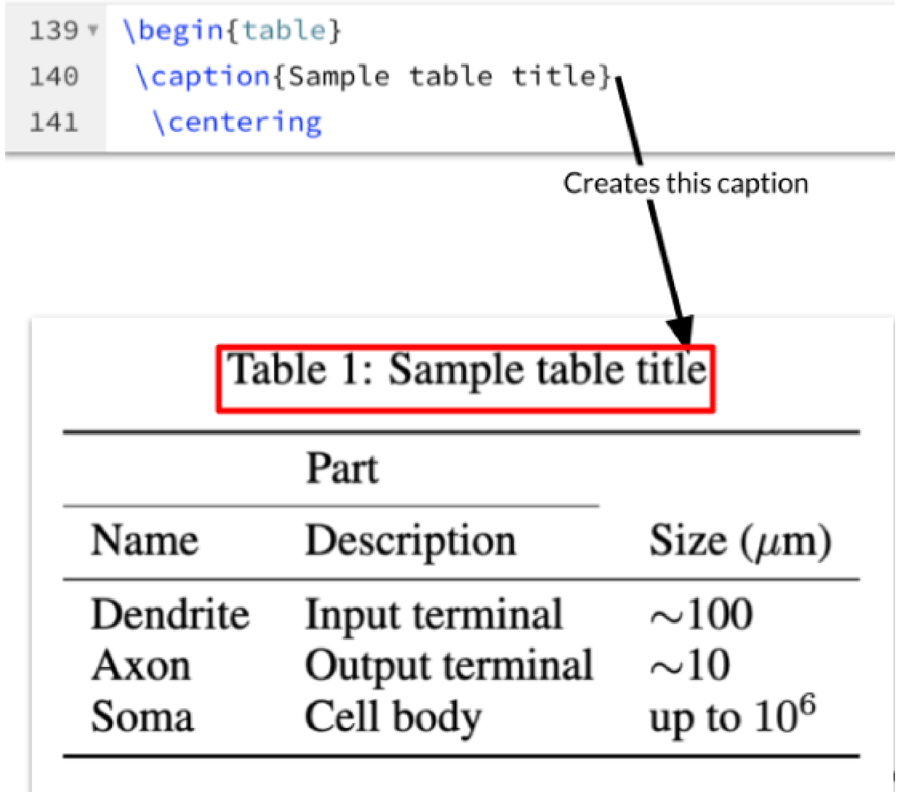
Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6



```
\begin{table}
 \caption{Sample table title}
  \centering
  \begin{tabular}{lll}
    \toprule
    \multicolumn{2}{c}{Part}                   \\
    \cmidrule(r){1-2}
    Name     & Description     & Size ($\mu$m) \\
    \midrule
    Dendrite & Input terminal  & $\sim$100     \\
    Axon     & Output terminal & $\sim$10      \\
    Soma     & Cell body       & up to $10^6$  \\
    \bottomrule
  \end{tabular}
  \label{tab:table}
\end{table}
```

We can add a caption using the `\caption{}` command. The table number will automatically be determined by the order of the tables. As before with the figure, the `\centering` command will then align the resulting table to be centered.

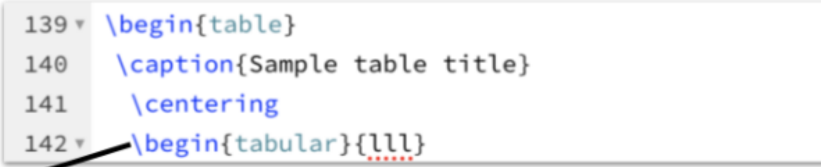


```
\begin{table}
 \caption{Sample table title}
  \centering
```

Creates this caption

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6



```latex
\begin{table}
 \caption{Sample table title}
  \centering
  \begin{tabular}{lll}
```

Makes the table “generally” be 3 columns wide with left alignment, unless otherwise specified

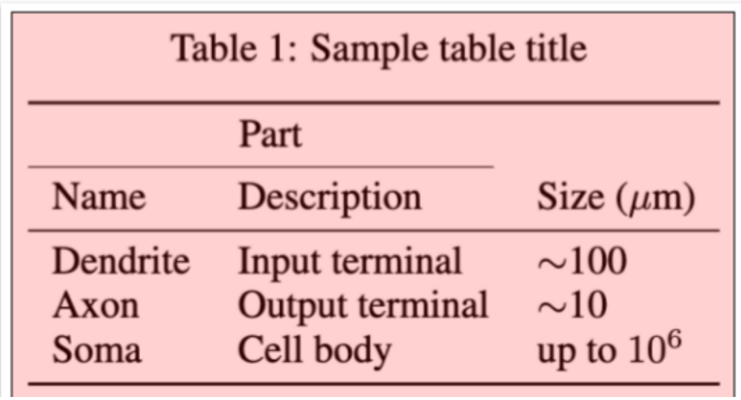
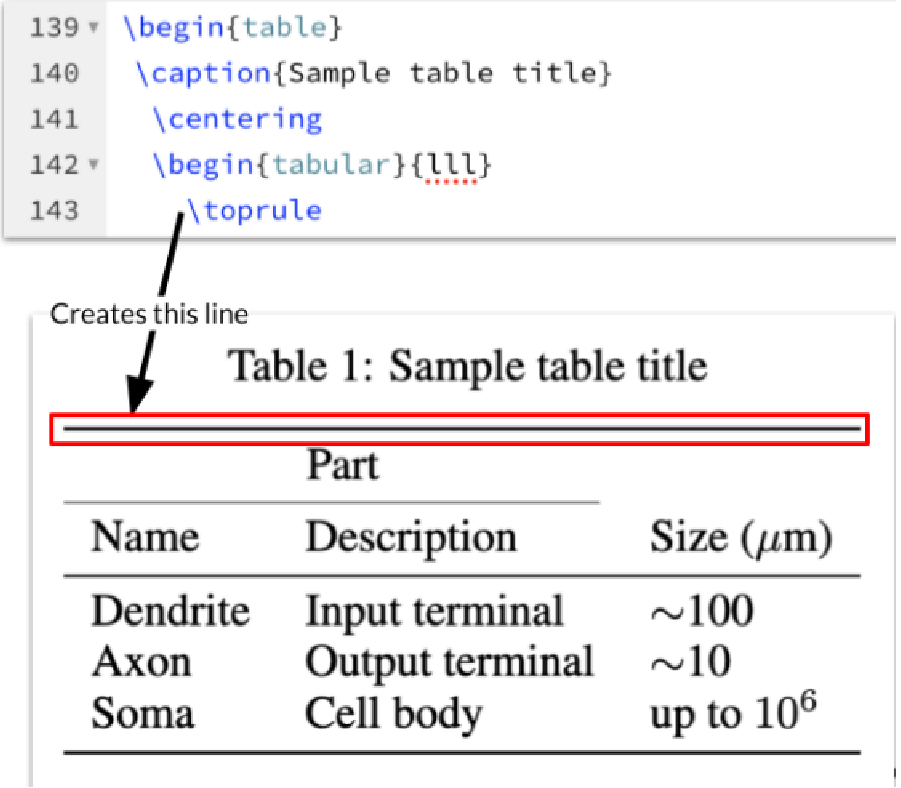


Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

To create the table in the template, we will first indicate how we want the table arranged using the `tabular` environment. The command `\begin{tabular}{lll}` indicates that we will have three columns that are left aligned.

The `\toprule` command adds a solid line at the top of the table. If you add this command again you will see two lines - test it out to see how it works!



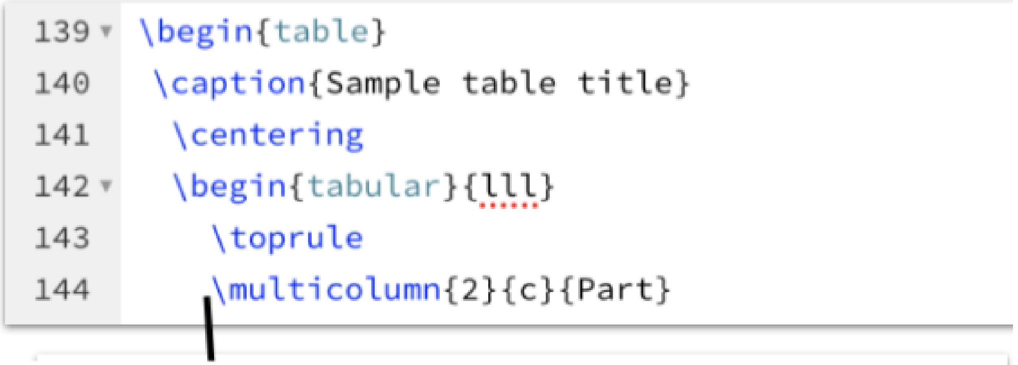
```
\begin{table}
 \caption{Sample table title}
  \centering
  \begin{tabular}{lll}
    \toprule
```

Creates this line

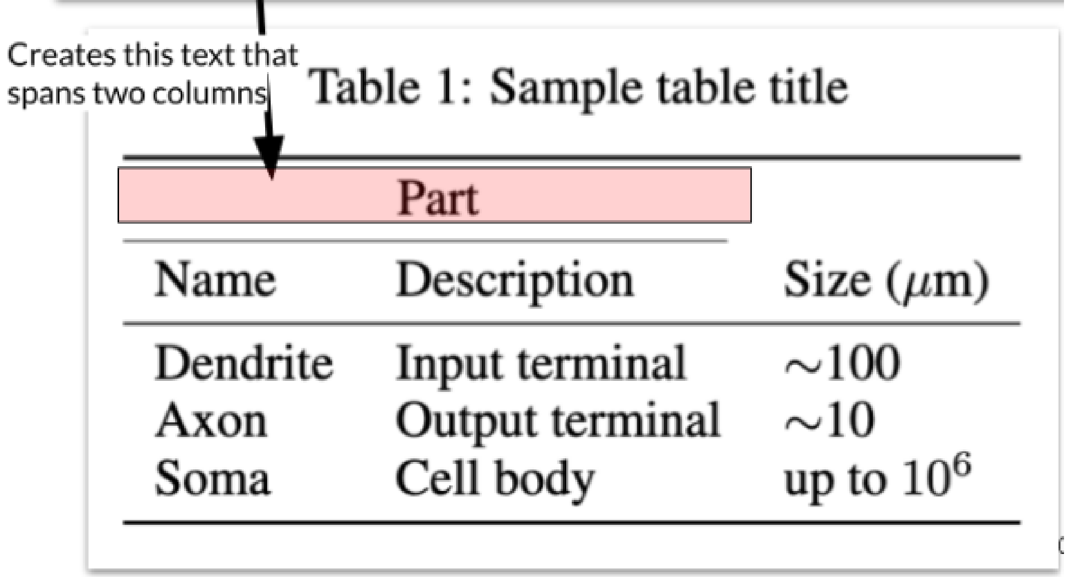
Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

- The command `\multicolumn{2}{c}{Part}\\` indicates that will will merge some columns together to create a “multicolumn” in this case the `{2}` means we will merge together 2 columns, the `{c}` indicates that it will be center aligned and the `{Part}` is the text we want for this. We need the `\\` to finish that row, otherwise “Part” will end up on the next row.



```
\begin{table}
 \caption{Sample table title}
  \centering
  \begin{tabular}{lll}
    \toprule
    \multicolumn{2}{c}{Part}
```

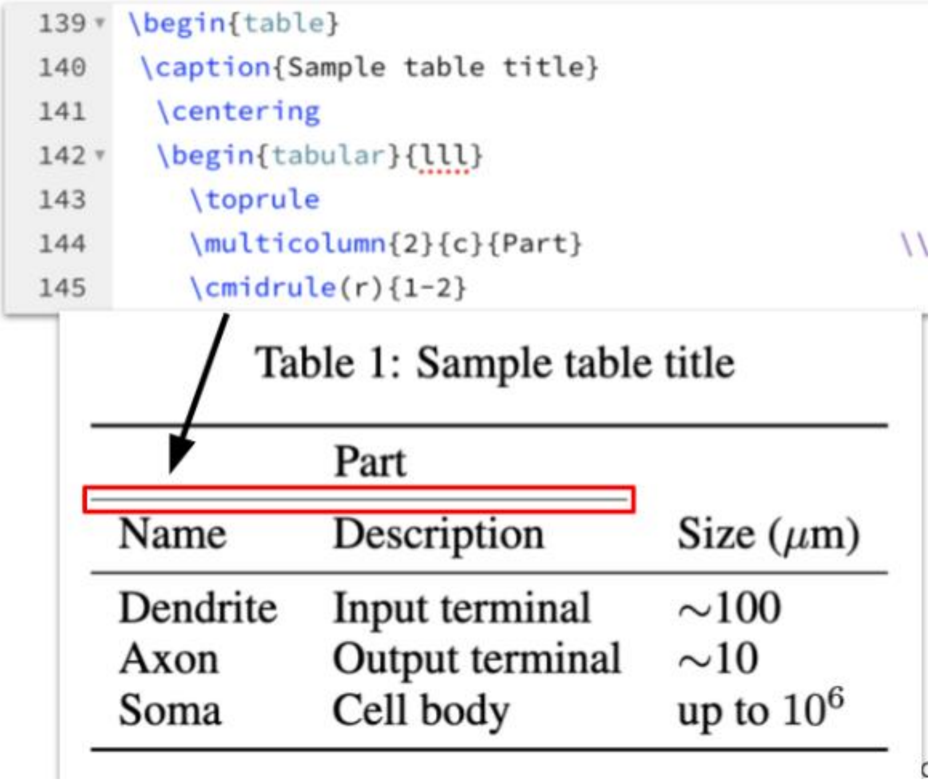


Creates this text that spans two columns

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

The `\cmidrule` command adds the line or “rule” under the multicolumn that says “Part”. This command creates lines that are not the full width of the table. The `(r) {1-2}` indicates that the line should be trimmed on the right side to leave a gap after the span of 2 column widths.

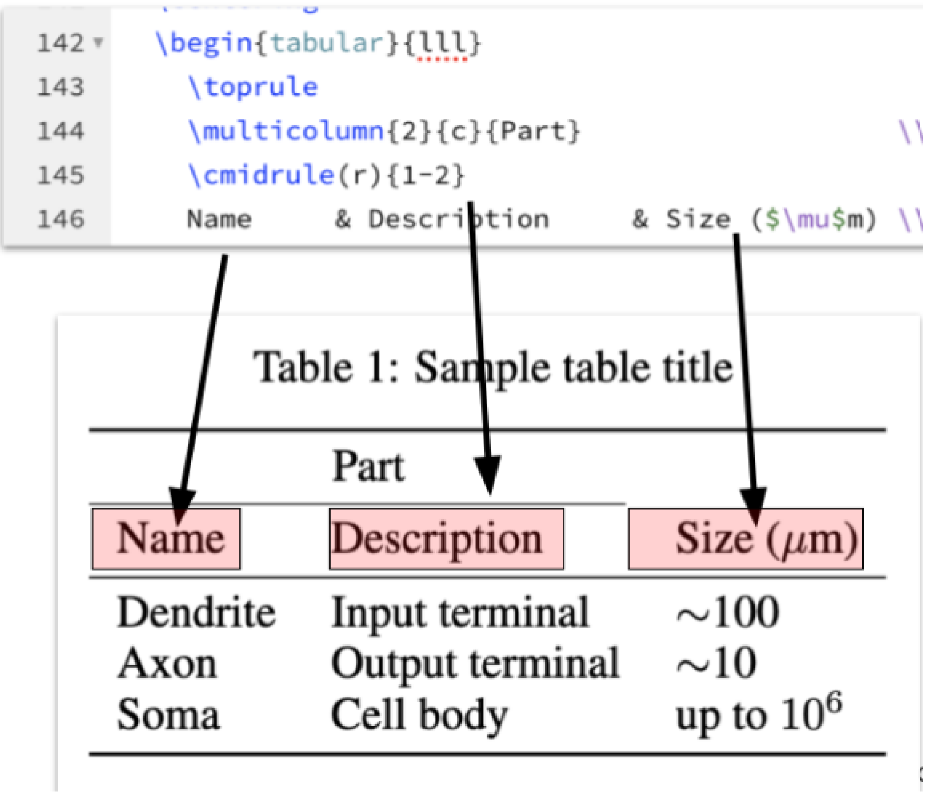


```
\begin{table}
 \caption{Sample table title}
  \centering
  \begin{tabular}{lll}
    \toprule
    \multicolumn{2}{c}{Part}                      \\
    \cmidrule(r){1-2}
```

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

Now we are ready to put some text within our table cells. We can simply type the words with an `&` in between the text for each cell to indicate where the column breaks are. The `\\` indicates when we are done with that row. Since we have a special character to represent mu, we can use mathematical notation by using a dollar sign `$`.

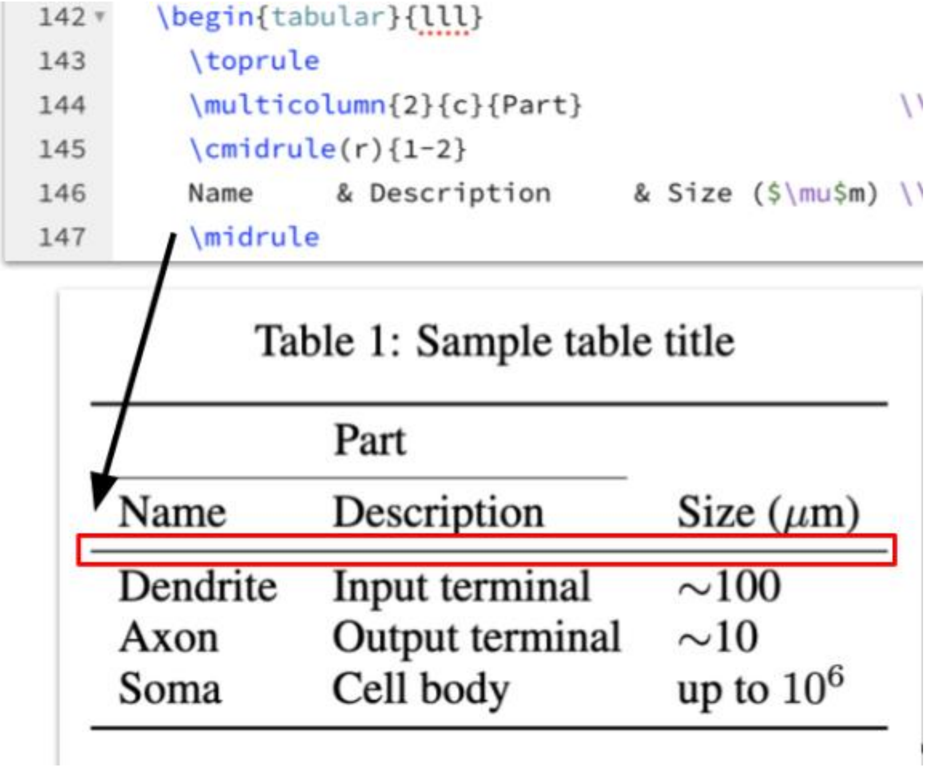


```
\begin{tabular}{lll}
  \toprule
  \multicolumn{2}{c}{Part}                              \\
  \cmidrule(r){1-2}
  Name      & Description      & Size ($\mu$m) \\
```

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

To add a line under these values, we can use the `\midrule` command.

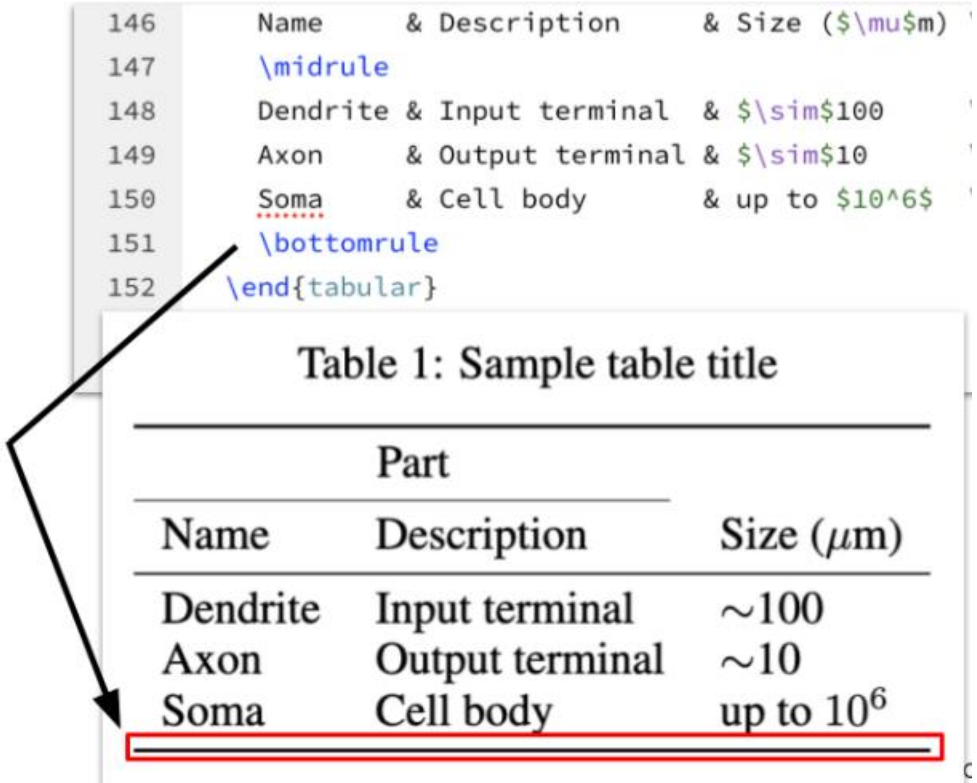


```
\begin{tabular}{lll}
  \toprule
  \multicolumn{2}{c}{Part}                        \\
  \cmidrule(r){1-2}
  Name     & Description     & Size ($\mu$m) \\
  \midrule
```

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

To add more text within the rows after this line, we simply type words, using an `&` to indicate column breaks and the `\\` to indicate the end of the row. The `$` is also used to create mathematical notations.

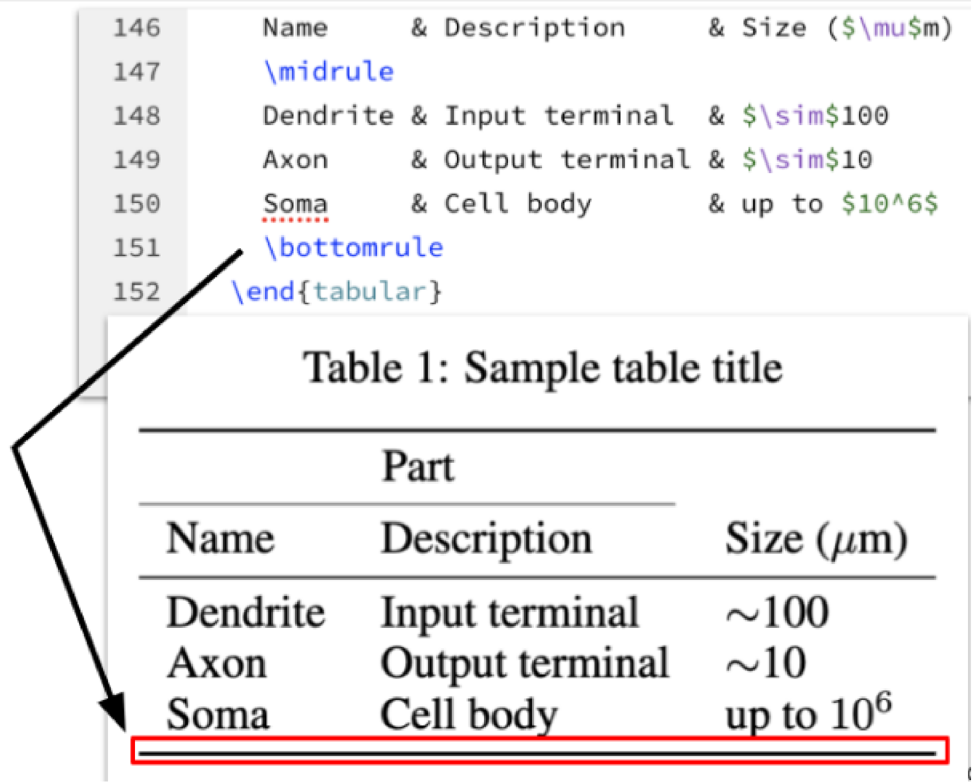


```
    Name     & Description     & Size ($\mu$m)
    \midrule
    Dendrite & Input terminal  & $\sim$100
    Axon     & Output terminal & $\sim$10
    Soma     & Cell body       & up to $10^6$
    \bottomrule
\end{tabular}
```

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

To add the line at the bottom, we need to use a command that is similar to `toprule{}` and `midrule{}` which is `bottomrule{}` .



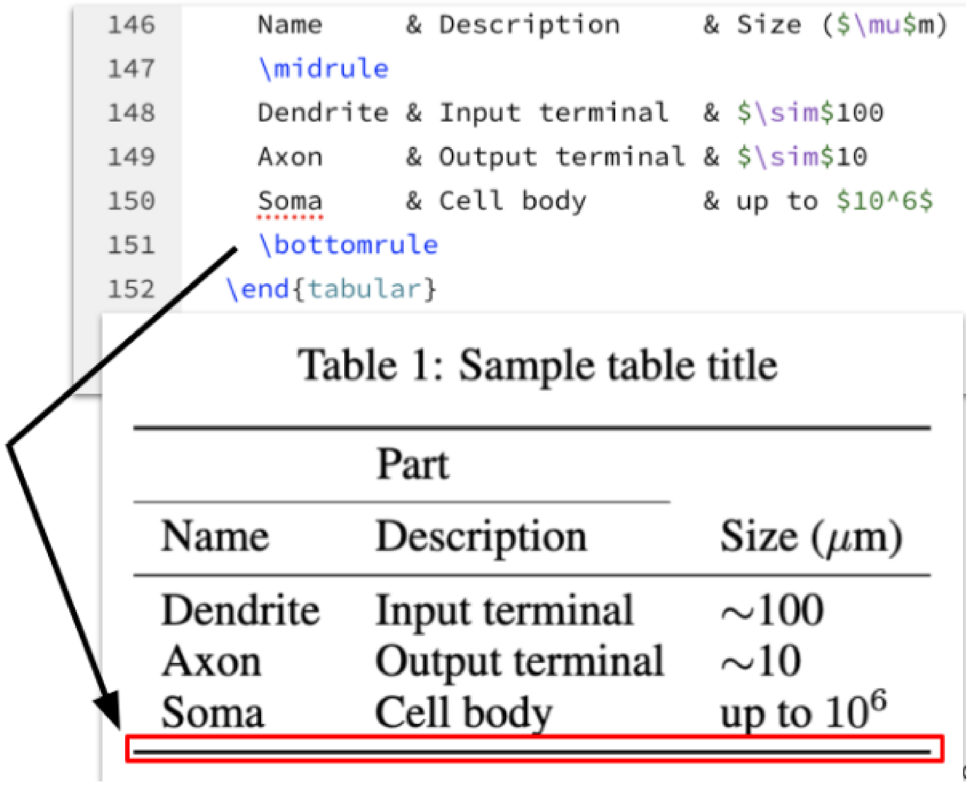
```
    Name     & Description      & Size ($\mu$m)
    \midrule
    Dendrite & Input terminal   & $\sim$100
    Axon     & Output terminal  & $\sim$10
    Soma     & Cell body        & up to $10^6$
    \bottomrule
\end{tabular}
```

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

Now we just need to finish off our table.

First we need to get out of the tabular mode, so we will use our trusty `\end{}` function. In this case, we end with `\end{tabular}`. To end the table overall, we use `\end{table}`. We will discuss what the `\label{}` function does soon.

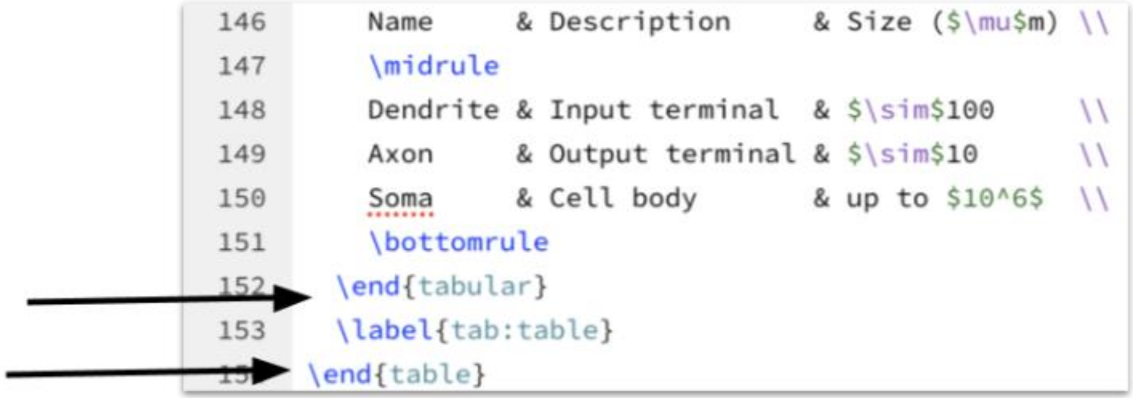


```
    Name     & Description     & Size ($\mu$m)
    \midrule
    Dendrite & Input terminal  & $\sim$100
    Axon     & Output terminal & $\sim$10
    Soma     & Cell body       & up to $10^6$
    \bottomrule
\end{tabular}
```

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

To add the line at the bottom, we again use `\bottomrule{}` .



```
    Name     & Description      & Size ($\mu$m) \\
    \midrule
    Dendrite & Input terminal   & $\sim$100     \\
    Axon     & Output terminal  & $\sim$10      \\
    Soma     & Cell body        & up to $10^6$  \\
    \bottomrule
  \end{tabular}
  \label{tab:table}
\end{table}
```

Creating internal links

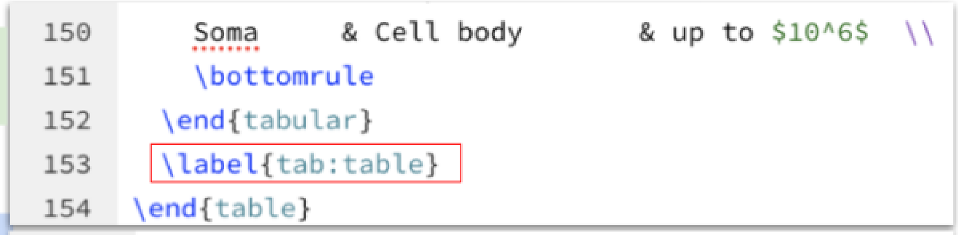
You may have noticed a command `\label{}` when looking through the template.

This is a very helpful command that creates a tag that allows you to refer back to a figure or a section header. There is also a nifty command `\ref{}` that creates references for these tags. However, it is a little tricky, so we will walk through a couple of examples.

You need the `\ref{}` command to match up with exactly what you have listed for the `\label{}` command for each tagged figure, table, or section.

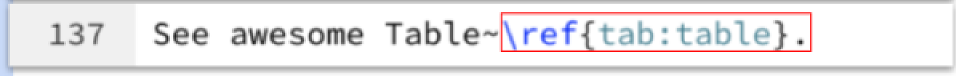
You also need to have the same notation for each type: - `tab:` for tables - `fig:` for figures - `sec:` for section headers

End of the code for the table:



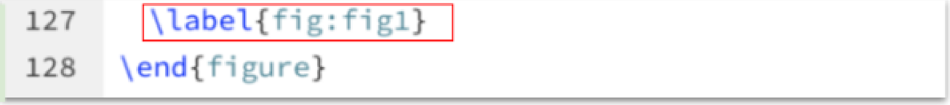
```
    Soma      & Cell body        & up to $10^6$  \\
    \bottomrule
  \end{tabular}
  \label{tab:table}
\end{table}
```

Code referring to the table:



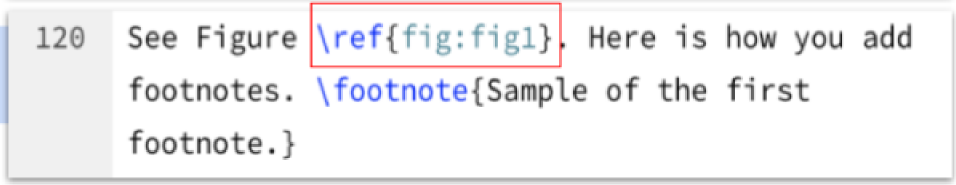
```
See awesome Table~\ref{tab:table}.
```

End of the code for the other figure:



```
  \label{fig:fig1}
\end{figure}
```

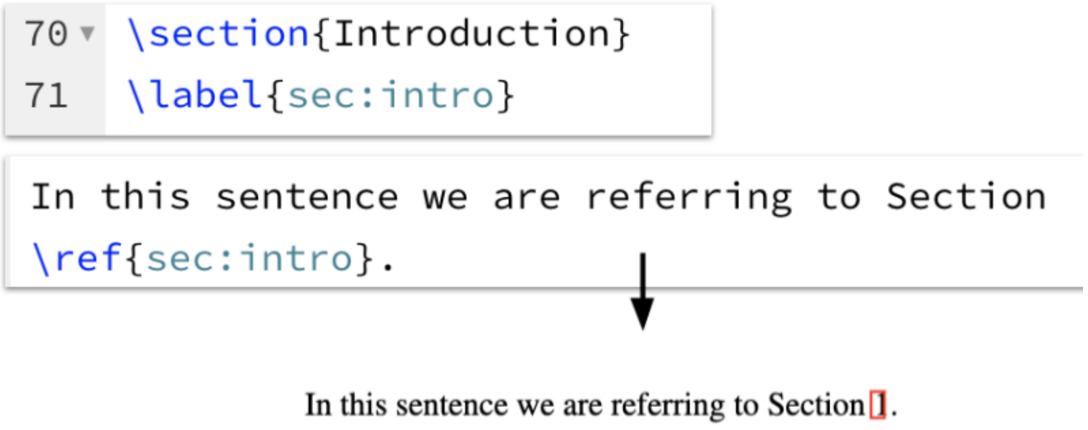
Code referring to the figure:



```
See Figure \ref{fig:fig1}. Here is how you add footnotes. \footnote{Sample of the first footnote.}
```

Here we will create a new link for the introduction. We first need to add a label to the introduction using the `label{}` function. We will call it `intro` and we need to specify that this is a section header with `sec` , like so: `label{sec:intro}` .

We then need to refer to this in the same way somewhere else using the `ref{}` function, like so: `ref{sec:intro}` . This will create a link to that section.



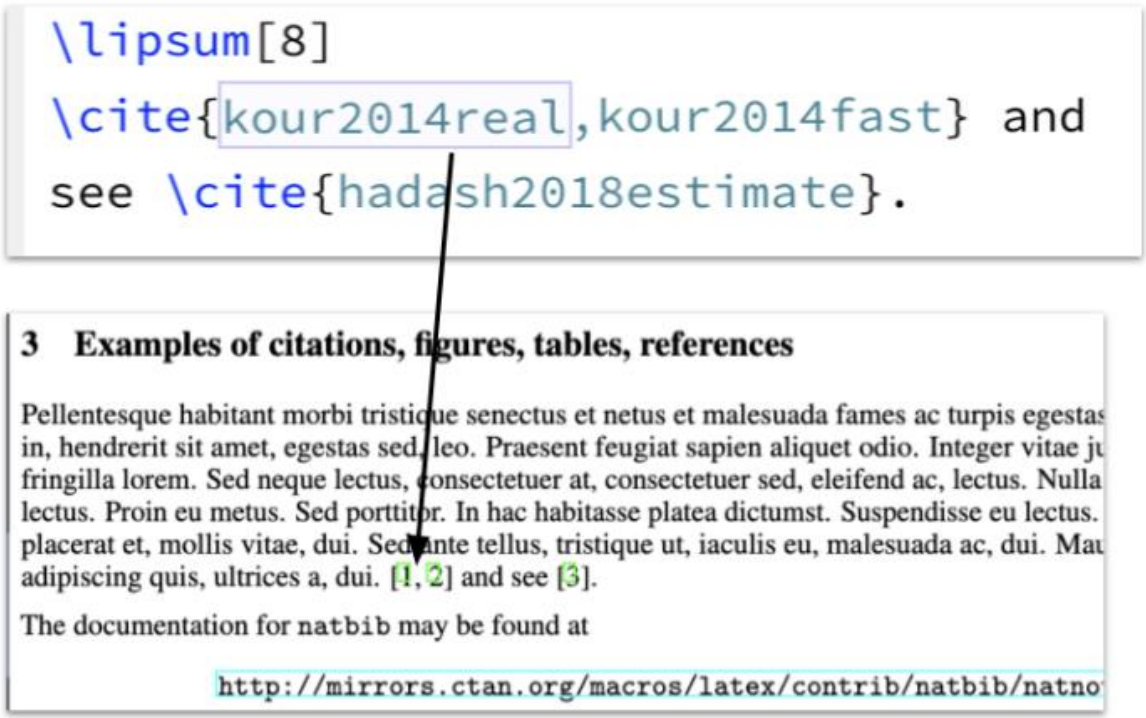
```
\section{Introduction}
\label{sec:intro}
```

```
In this sentence we are referring to Section
\ref{sec:intro}.
```

In this sentence we are referring to Section 1.

References

- Almost all scientific articles need references. To add these we can add to the **references.bib** file, which is found on the left menu.
- There are many ways to get the bib version of a reference. One easy way is to use Zotero, which is a free tool for writing bibliographies that has a chrome extension.
- The cool thing about the chrome extension is if you are viewing an article or a website online, you can often right click to add the file to Zotero. Then you can find the file in Zotero and right click to export the item to BibTeX format. This is a bibliography format that is compatible with TeX. We can then copy and paste this into the references.bib file, being careful to make sure that the brackets are closed.



```
\lipsum[8]
\cite{kour2014real,kour2014fast} and
see \cite{hadash2018estimate}.
```

3 Examples of citations, figures, tables, references

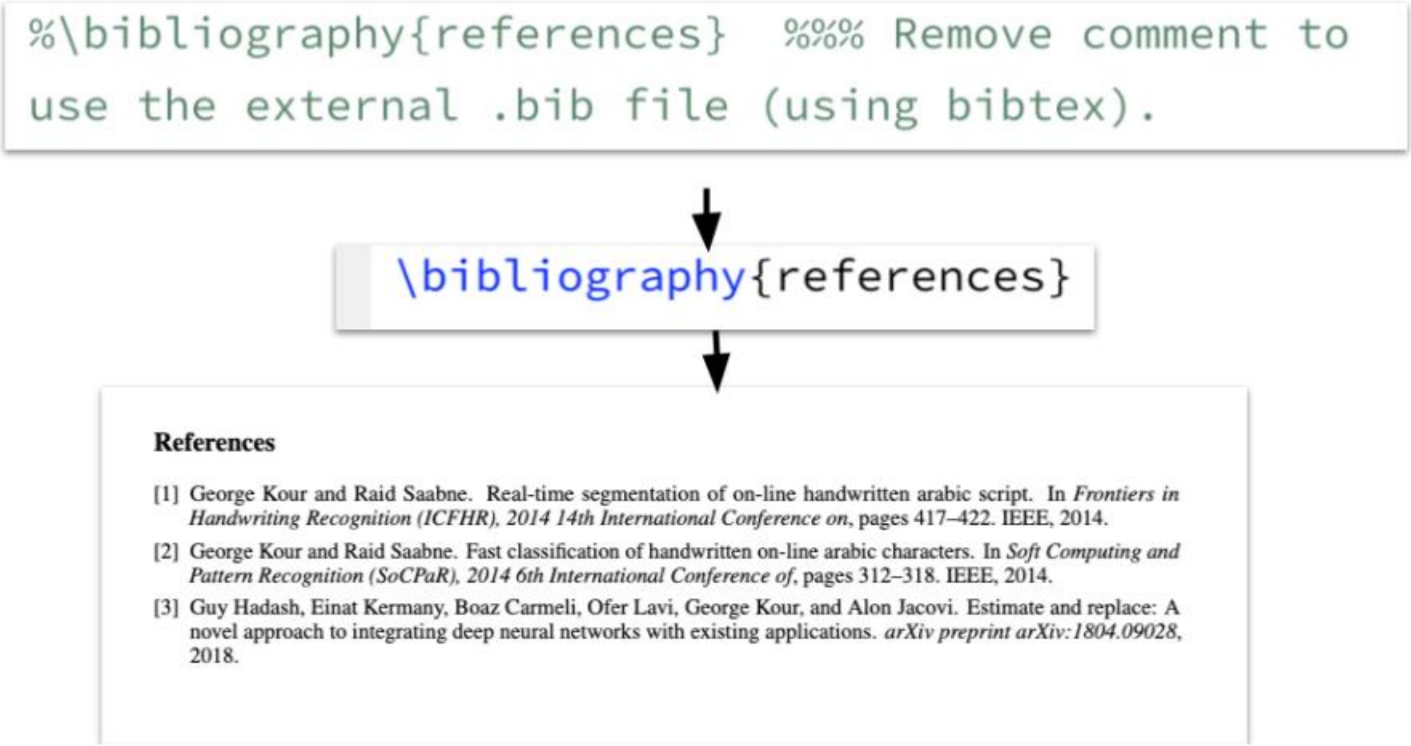
Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas
in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae ju
fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla
lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus.
placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mau
adipiscing quis, ultrices a, dui. [1, 2] and see [3].

The documentation for natbib may be found at

http://mirrors.ctan.org/macros/latex/contrib/natbib/natno

- The first part of the bib item will indicate what to refer to it in the text to create a citation to the reference. The first item that starts with an @ in the template references.bib file shows kour2014real in the brackets. We can see that line 100 uses the \cite{} function to cite this article, as well as another article.
- This results in a citation number and a link to the reference.

To add a bibliography, we can just undo the comment in front of the bibliography command. This is sufficient to create the bibliography. The code after this in the template to create each reference individually is not needed.



```
%\bibliography{references}  %%% Remove comment to
use the external .bib file (using bibtex).
```

```
\bibliography{references}
```

References

[1] George Kour and Raid Saabne. Real-time segmentation of on-line handwritten arabic script. In *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*, pages 417–422. IEEE, 2014.

[2] George Kour and Raid Saabne. Fast classification of handwritten on-line arabic characters. In *Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference of*, pages 312–318. IEEE, 2014.

[3] Guy Hadash, Einat Kermany, Boaz Carmeli, Ofer Lavi, George Kour, and Alon Jacovi. Estimate and replace: A novel approach to integrating deep neural networks with existing applications. *arXiv preprint arXiv:1804.09028*, 2018.