How to Make Beautiful Technical Documents with LaTeX

PHYS 87

Benjamín Grinstein

UCSD

Spring 2021

LaTeX Install

You received an email asking you to install LaTeX:

- It's free!
- Interactive course, LaTeX necessary
- Use it for other courses or tasks
- Use it in the future

Course info

- So where was that web site? `http://physics.ucsd.edu/students/courses/spring2021/physics87/`
- Grading: attendance at least 80%
- Attendance: submit day's work by uploading to
    - Upload your work to `https://u.pcloud.com/#page=puplink&code=74N7ZfvOlSAmiuhjj39dohoFBMfR2zRmy` (Find link to this also on the course website, ABOUT>Grading)
    - Drag tex and pdf files to pCloud screen (or click "Browse" and navigate to selection)
    - Enter "Your name" as LastName, FirstName
- Justified absence may be considered, contact instructor!
- Office hours: by zoom or Skype. email instructor to schedule. Will also hang out a bit after lectures
- 8–10 sessions, normally one per week; check CANVAS and/or email
- If you are not having fun, something is off. Let me know!
- Chatter, questions and challenges welcome.

CANVAS

We will use Canvas in addition to the course web site:

- For Announcements
- To post video recording of lectures (all lectures are recorded!)
- To access Zoom lecture links:
  - "Zoom LTI Pro"
  - "Calendar"
- Chat/Discussion (possible but historically unpopular)

We will use Texworks

Worry about what LaTeX is later. For now, get it to work!
LaTeX and Texworks are installed in workstations. I will
explain what these are later

- The Dock is the bottom bar of icons
- On Dock, rightmost icon is "Finder"
- Clcik on Finder. Locate and click on "Applications" on
  sidebar of window that opened up.
- Scroll Down to Folder "TeX" and double click to open
- Double-click on Texworks
- If prompted say you do NOT want to install latest update

First Project

(If not already) Open Texworks. Type the following

```
\documentclass{article}
```

```
\begin{document}
```

```
Hello world!
```

```
\end{document}
```

Now, let's insert a mistake into the code and learn how to deal with it. Go back to your first project and type a line right below `Hello world!` as follows:

```
\documentclass{article}

\begin{document}

Hello world!
\parr

\end{document}
```

```
This is pdfTeX, Version 3.14159265-2.6-1.40.15 (TeX Live 2
 format=pdflatex)
 restricted \write18 enabled.
entering extended mode
(./first_project.tex
LaTeX2e <2014/05/01>
Babel <3.9l> and hyphenation patterns for 79 languages load
(/usr/local/texlive/2014/texmf-dist/tex/latex/base/article.
Document Class: article 2014/09/29 v1.4h Standard LaTeX doc
(/usr/local/texlive/2014/texmf-dist/tex/latex/base/size10.c
(./first_project.aux)
! Undefined control sequence.
l.6 \parr

?
```

In the text box type `?-ret`, that is, type a question mark and hit return. You should see

```
? ?
Type <return> to proceed, S to scroll future error messages
R to run without stopping, Q to run quietly,
I to insert something, E to edit your file,
1 or ... or 9 to ignore the next 1 to 9 tokens of input,
H for help, X to quit.
?
```

The first question mark was there before, the second one is the one you typed. The next few lines remind you of your options. The last question mark tells you that the program is waiting for your input again.

Option 2: Proceed

Now click in the text box and hit return without typing anything. The program runs through the end, and the output is produced. It looks exactly as before, with "Hello World!" near the top. The following is added to the console window

```
[1{/usr/local/texlive/2014/texmf-var/fonts/map/pdftex/updma
(./first_project.aux) )</usr/local/texlive/2014/texmf-dist/
msfonts/cm/cmr10.pfb>
Output written on first_project.pdf (1 page, 11915 bytes).
SyncTeX written on first_project.synctex.gz.
Transcript written on first_project.log.
```

There is useful information in this, but let's not get bogged down with all of it now. Notice only that an output file, `first_project.pdf`, was written. You can find it in the directory (folder) where you saved your project. So the output is not just on the output screen. It is in a file that you can share with others and print.

Sometimes you know that something has gone horribly wrong and you just want the compiler to stop immediately, aborting the compilation. To see how this works, hit the green Typeset button again and when compilation stops, type `x-ret` into the type-box. The console then adds below the question mark the lines:

```
No pages of output.
Transcript written on "first_project.log".
```

Since you have told the compiler to stop, there in no output.

Fixing it

Go back tou your project. Remove one r from \parr. Your input file reads:

```
\documentclass{article}

\begin{document}

Hello world!
\par
\end{document}
```

Typeset this. No error messages any more. Output is produced.

Syntax Basics

- Characters: A–Z, a–z, 0–9, some symbols (comma, period, etc) excluding reserved characters (see bellow)
- White space
- Empty lines
- Reserved characters: # $ % ^ & _ { } ~ \
- Latex Groups:

  ```
  {\bf This is bold.} This is no longer bold.
  ```

- LaTeX commands: start with \ followed by <u>letters only</u>, the `name` of the command.
  - Commands with arguments and options. Example

    `\hspace{1in}`

    The `\hspace` command takes as an argument a distance measure, in this case 1 inch. It leaves 1 inch of space in your text. Try this in your Hello World! document, forcing 1 inch of space between the two words:

    ```
    \documentclass{article}

    \begin{document}

    Hello \hspace{1in} world!

    \end{document}
    ```
  - Switches. In the example above, `\bf` is the boldface command. It is a switch. It changes the typesetting into boldface characters. The switch is limited to the group enclosed in curly brackets { and }

- Comments: anything following % in the current line is ignored. Add a comment to your first project now! Example:

```
\documentclass{article}

\begin{document}

Hello \hspace{1in} world! %This won't show

% neither will this
\end{document}
```

- LaTeX Environments:

```
\begin{environmentname}
text to be influenced
\end{environmentname}
```

Let's try this with the *center* environment:

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

Under the hood

- Open folder where you saved `first_project.tex`
- Find several other files, among them:

`first_project.pdf` PDF file $\Rightarrow$ Reader, Skim, Nitro, ...
`first_project.log` Plain text: log-file (warning & errors)
`first_project.aux` Plain text: keeps track of internals.

- Texworks is not LaTeX; rather:
  Texworks calls LaTeX (a separate "app").
- Texworks contains a PDF viewer:
  no need to use Reader separately.
- Texworks contains a rudimentary text editor:
  type and save your project.
- Techies:
  1. Create your project using advance editor of your choice
     (emacs? gEdit?)
  2. From Terminal(Mac)/Command(Windows) type
     `pdflatex` *filename*
  3. See *filename*.`pdf` with Reader (Skim? Preview?)

`\documentclass` takes two arguments:

`\documentclass[options]{class}`

There are other classes, such as `book` and `letter`. See the complete list in the LaTeX Wikibook. We will stick with `article` for now.

There are many options. You can enter them in a comma separated list. Let's try it. Start a new document with

```
\documentclass[letterpaper,notitlepage,10pt]{article}
\begin{document}
Hello world again!
\end{document}
```

Now change `12pt` for `10pt`.

Options are `10pt`, `11pt` and `12pt`. Default is `10pt`.

Class `extarticle` allows more.

Using packages

Packages enhance the capabilities of LaTeX. There is a huge number of packages available. We can see a listing of packages installed in our systems with TeXLive Utilities (Package Manager).
Example:

```
\documentclass[letterpaper,notitlepage,11pt]{article}

\usepackage{color}
\begin{document}
Hello {\color{red} world} again!
\end{document}
```

Packages can take options too:

```
\usepackage[options]{packagename}
```

You can call many packages:

```
\usepackage[options]{packagename1}
\usepackage[options]{packagename2}
\usepackage{packagename3,packagename4,packagename5}
```

Try this:

```
\documentclass[letterpaper,notitlepage,11pt]{article}
\usepackage[margin=2cm]{geometry}
\usepackage{color}
\begin{document}
Hello {\color{red} world} again!
\end{document}
```

My First Full Article

Now we add a title, author, date and abstract to our scientific breakthrough:

```
\documentclass[letterpaper,notitlepage,11pt]{article}
\usepackage[margin=2cm]{geometry}
\usepackage{color}
\begin{document}
\title{The Origin of Species}
\author{Charles Darwin}
\date{December 2014}
\maketitle
\begin{abstract}
  I copy-pasted this from wikipedia in
  http://en.wikipedia.org/wiki/On_the_Origin_of_Species
  (copy-paste what you want):
  On the Origin of Species, published on 24 November 1859, is a
  work of scientific literature by Charles Darwin which is considered
  to be the foundation of evolutionary biology. Its full title was On
  the Origin of Species by Means of Natural Selection, or the
  Preservation of Favoured Races in the Struggle for Life. For the
  sixth edition of 1872, the short title was changed to The Origin of
  Species. Darwin's book introduced the scientific theory that
  populations evolve over the course of generations through a process
  of natural selection.
\end{abstract}

Hello {\color{red} world} again!
\end{document}
```

Try this: change the date to `\date{\today}`

Copy the first three paragraphs, from Wiki's **On the Origin Of Species** from **On the Origin of Species,** *published ...* until *...become the unifying concept of the life sciences.* Paste in your document, and remove the *Hello world again!* line.

Blank line = paragraph (play with this)

Line Breaks
Suppose you want to break a line, without starting a new paragraph.

Insert \\ after three words (make sure you leave at least one blank space after this).

Insert instead `\break` after three words (make sure you leave at least one blank space after this).

Insert instead `\hfil\break` after three words (make sure you leave at least one blank space after this). We'll talk about `\hfil` in more detail later, but obviously it fills up the blank space.

```
\documentclass[letterpaper,notitlepage,11pt]{article}
\usepackage[margin=2cm]{geometry}
\usepackage{color}
\begin{document}
\title{The Origin of Species}
\author{Charles Darwin}
\date{\today}
\maketitle
\begin{abstract}
  I copy-pasted this from wikipedia (copy-paste what you
  want): On the Origin of Species, published on 24 November 1859, is a
  work of ...  blah blah ... a process
  of natural selection.
\end{abstract}

\section{Introduction}
On the Origin blah blah blah

blah blah  life sciences.
\end{document}
```

You may also need sections of sections, or *subsections*. Type in

`\subsection{Generalities}`

right after the `\section{Introduction}` line and again
between the first and second paragraphs

`\subsection{More stuff}`

Play with leaving blank lines before and after these commands.
There are more sectioning commands. See the Wikibook for the
complete list.

Now insert between the second and third paragraphs the command

```
\newpage
```

You should stay away form using this. But sometimes you must.

Color help in editing LaTeX source

It helps finding stuff out if the document you are editing has the commands and environment delimiters in different color than normal text.

In TeXworks go to the pull down menu from the titlebar *Format>Syntax Coloring>LaTeX*. Your commands should appear in blue and the environments in dark green

Our third project:

\documentclass[11pt]{article}

\begin{document}
My fist equation is $F=ma$.
\end{document}

Now add to the last line:

My fist equation is $F=ma$. My second is \(a=F/m\).

Change the above as follows

```
My fist equation is $F=ma$. My second is \[ a=F/m\].
```

Let's add more text:

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

\begin{document}
My fist equation is $F=ma$. My second is \[ a=F/m\].
The period is in the wrong place. I should place it inside
displayed equation. I want to write enough that there will
lines after the displayed equation blah blah blah.
\end{document}
```

Place period in correct place.

Another version:

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

\begin{document}
My fist equation is $F=ma$. My second is
\begin{equation}
 a=F/m.
\end{equation}
The period is in the right place. I have place it inside the
displayed equation. I want to write enough that there will be a few
lines after the displayed equation blah blah blah.
\end{document}
```

Notice the equation number.
You can't leave a blank line in a displayed equation:

```
\begin{equation}
 a=F/m.


\end{equation}
```

gives an error message ! `Missing $ inserted` in the console
window.

Pairs: \label{*label*} and \ref{*label*}

```
\begin{equation}
\label{accel}
 a=F/m.
\end{equation}
The period is in the right place. I have placed it inside
the displayed equation \ref{accel}. I want to write enough that
there will be a few lines after the displayed equation
blah blah blah.
\end{document}
```

Typeset: in place of \ref{accel} get ??.

What went wrong? Force *Console Output* back by selecting
Window > Show Console Output:

```
LaTeX Warning:  There were undefined references.
LaTeX Warning:  Label(s) may have changed.  Rerun to
get cross-references right.
```

1. The clue to what must be done is in "Rerun to get cross-references right.." So hit the typeset button again. Now the typeset document will display the number 1 in place of \ref{accel}. The two "LaTeX warning" lines don't come up in the console anymore.

2. You can check more detail of the console output in a file created with .log extension, as in third_project.log

3. Two passes necessary: first one creates *auxiliary* file third_project.aux, second one reads it. It keeps track of references/labels. Look at file listing in folder.

Notice no parenthesis around "1." So try instead (\ref{accel}).

Now, to make the point of labeling obvious, insert *before* the current equation, another equation:

```
...
\begin{equation}
\label{eq:Newton}
F=ma
\end{equation}
is the same as
\begin{equation}
\label{accel}
 a=F/m.
\end{equation}
The period is in the right place. I have placed it inside
the displayed equation (\ref{accel}). I want to write enoug
that there will be a few lines after the displayed equation
blah blah blah. I added equation (\ref{eq:Newton}).
\end{document}
```

Let's Take a Break from Math!

But we will get back with much more

More math later...

You can use the pair `\ref{token}` – `\label{token}` to insert references to Sections, sub-Sections, etc
In your text:

```
\section{Method}
\label{my-methods}
Here we discuss the methods...
```

Elsewhere in the text you can write
```
In Sec.~\ref{my-methods} it is shown how to bake a cake,
```
for example.
You will have to process twice to get references right (each time a new `\ref` is introduced).

Here is how to make one:

```
\begin{itemize}
\item The first entry here
\item Then the second
\item etc
\end{itemize}
```

- The first entry here
- Then the second
- etc

To change the bullet into something else you have two options.
First, line by line:

```
\begin{itemize}
\item[-] The first entry here
\item[*] Then the second
\item[$>$] etc
\end{itemize}
```

If you want to change the item symbol for the whole document
insert the following before the first {\begin{itemize} (can go
in the preamble):

```
\renewcommand{\labelitemi}{\textgreater}
```

You can change this as many times as you please — but outside
the itemize environment.

Exercise: try typesetting this

> The first entry here
> Then the second
> etc

- The first entry here
- Then the second
- etc

Hint: Use `\textgreater` for ">" and `$\bullet$` for "•".

Here is how:

```
\renewcommand{\labelitemi}{\textgreater}

\begin{itemize}
\item The first entry here
\item Then the second
\item etc
\end{itemize}

\renewcommand{\labelitemi}{$\bullet$}

\begin{itemize}
\item The first entry here
\item Then the second
\item etc
\end{itemize}
```

You can nest lists:

```
\begin{itemize}
\item The first entry here
\item Then the second, which leads to
\begin{itemize}
\item The first sub-entry here
\item Then the second sub-entry
\item etc
\end{itemize}
\item Return to original list, etc
\end{itemize}
```

Exercise: make a tripple nested list

If you want to change the symbols of these lists do as before,
inserting before `\begin{itemize}` your choices:

```
\renewcommand{\labelitemi}{\textgreater}
\renewcommand{\labelitemii}{$\star$}
```

Exercise: How do you get this default:
  > First level
      ⋆ Second level
          • Third level
Check that it works by typesetting the tripple ensted list of the
pervious exercise.

Hint: Symbols used: `\textgreater`, `$\star$`, `$\bullet$`.

You can use

```
\begin{enumerate}
\item The first entry here
\item Then the second
\item etc
\end{enumerate}
```

1. The first entry here
2. Then the second
3. etc

You can also nest these:

```
\begin{enumerate}
\item The first entry here
\item Then the second, which leads to
\begin{enumerate}
\item The first sub-entry here
\item Then the second sub-entry
\item etc
\end{enumerate}
\item Return to original list, etc
\end{enumerate}
```

Question: what "numerals" are used in the second nested list?

For this we use the `enumerate` package.

```
\usepackage{enumerate}
...
\begin{document}
...
\begin{enumerate}[(i)]
\item The first entry here
\item Then the second
\item etc
\end{enumerate}
```

This gives:

(i) The first entry here

(ii) Then the second

(iii) etc

The option we used is one of `A, a I, i` and `1`. The parenthesis, and anything else, is treated as text. If you want to use any of `A, a I, i` and `1` in the literal text you should enclose it in curly brackets, as in {I}. So for example

```
\usepackage{enumerate}
...
\begin{document}
...
\begin{enumerate}[Exerc{i}se 1)]
\item The first entry here
\item Then the second
\item etc
\end{enumerate}
```

works, but try leaving out the curly brackets (really, try!).

This is all that the package `enumerate` does. Just a simple way of handling the labels. The more sophisticated package `enumitem` gives you much more control. For example, you can modify the layout of the list. But it is more difficult to use. For example, the same output as above is obtained by using

```
\usepackage{enumitem}
...
\begin{document}
...
\begin{enumerate}[label=Exercise \arabic*)]
\item The first entry here
\item Then the second
\item etc
\end{enumerate}
```

So this is more complicated. But now we can do more.

For example,

```
\usepackage{enumitem}
...
\begin{document}
...
\begin{enumerate}[label=Exercise \arabic*), itemsep=1in]
\item The first entry here
\item Then the second
\item etc
\end{enumerate}
```

There are many such controls, `topsep`, `leftmargin`, `rightmargin` and `labelwidth` among others. You should play with these. Get the full power of this package from the `enumitem` package documentation. It allows for referencing entries in the item lists, making inline lists and more.

This is sometimes useful; it left justifies and boldfaces the descriptor (not as typeset in this slide!):

```
\begin{description}
\item[First] The first entry here
\item[Second] Then the second
\item[Last] Then the last
\end{description}
```

First The first entry here

Second Then the second

Last Then the last

The argument of `item` is optional, but why would you not use it, and the output would look peculiar.

Exercise: Typeset this:

First The first entry here

Second Then the second

Last Then the last

Hint: \usepackage{color}

Answer: Per the hint place `\usepackage{color}` in the preamble. Then

```
\begin{description}
\item[\color{red}First] The first entry here
\item[\color{blue}Second] Then the second
\item[\color{black}Last] Then the last
\end{description}
```

Bibliography

- A painful aspect of composing a technical or academic document
- Would like a simple way
  - of citing papers or books,
  - to list them accordingly at the back of our document.
- Sometimes we are asked to have the listing of references
  - alphabetized,
  - in the order in which they are used/cited in the document.
- The style of the references may change from publisher to publisher too.
- What a pain!

Fortunately LaTeX can take care of all this for you.

So we need to refer to, say,

On the Relation between the Expansion and the Mean Density of the Universe, A. Einstein, W. de Sitter, Proc.Nat.Acad.Sci. 18 (1932) 213-214

and

On Gravitational waves, Albert Einstein, N. Rosen, J.Franklin Inst. 223 (1937) 43-54

And the text should read something like

Einstein discussed the relation between the expansion and the mean density of the universe[1], and then he went on to say something about gravitational waves[2], whatever they are

To do this in LaTeX we first write text into our document:

```
Einstein discussed the relation between the expansion
and the mean  density of the universe\cite{albert1},
and then he went on to say something
about gravitational waves\cite{einsteinRosen},
whatever they are.
```

Note that the labels are completely arbitrary text tokens. You choose!

Now add before the \end{document} the following references:

```
\begin{thebibliography}{99}

\bibitem{albert1}
On the Relation between the Expansion and the Mean
Density of the Universe, A. Einstein, W. de Sitter,
Proc.Nat.Acad.Sci. 18 (1932) 213-214

\bibitem{einsteinRosen}
On Gravitational waves
Albert Einstein, N. Rosen, J. Franklin
Inst.~223~(1937)~43-54
\end{thebibliography}
```

The blank lines between bibitems are ignored by LaTeX: you can add more or have none at all. Blank lines within a bibitem produce unwanted line breaks.

As with other referencing in LaTeX the first time you run this you get question marks in place of citing references, as in *Einstein discussed . . . of the universe[?], and then he went on to say something about gravitational waves[?], whatever they are.* The Console output contains,

```
LaTeX Warning: Citation 'albert1' on page 1 undefined
on input line 15.

LaTeX Warning: Citation 'einsteinRosen' on page 1
undefined on input line 16.
```

and then lower down, near the end,

```
LaTeX Warning: There were undefined references.

LaTeX Warning: Label(s) may have changed. Rerun to
get cross-references right.
```

The last line tell you how to fix the problem. Just run LaTeX again. But the Warning messages are more generally useful for debugging problems, as we will discuss in lecture. So typeset again. Now the question marks are replaced by [1] and [2], and the Console output gives no Warning messages.

Exercise: Try this: List the two references in the bibliography in the reverse order. What happens? Make sure you look at the typeset output and the Console output after each time you run LaTeX now. (To open the console select *Show Console Output* from the *Window* pull down menu.)

Exercise: Now try this: add to the text a `\cite{peter}` somewhere. What is displayed in the text? Does it change if you run LaTeX again? Think about this: why does it (or not) change? Look at the "References" section. Notice any changes? Check the console output for warnings.

Exercise: Finally try this: add `\cite` commands to the two references you have several places in the text, in whatever order you want.

The mandatory argument of the `thebibliography` environment indicates the width of the field used for enumerating. It tells LaTeX the maximum number of digits expected in enumerating citations.

If the argument is

- A digit between 1 and 9 (say, 1 or 7 or 9) it means *one* digit which is useful for a list with no more than 9 entries.
- A number between 10 and 99 (say 34 or 87 or 99) it tells LaTeX you will use two digits for the list. That's 99 references maximum.

It is common to use as argument 9, or 99, or 999. It is never a bad idea to overestimate the number of references in your work.

Bibliography

BibTeX is more flexible, but more difficult to use. It has the great advantage that bibliography files are stored and can be reused in many documents. They are databases. Only the references you need are used.

We need to

1. Make one (or more) plain text bibliography files (databases), extension `.bib`, say `samplebib1.bib`
2. Include `\cite{bibitemlabel}` in text as before
3. Include
   `\bibliographystyle{plain}`
   `\bibliography{samplebib1}`
   before `\end{document}`
4. Typeset with *pdfLaTeX*. Then run *BibTeX*. Then typeset twice with *pdfLaTeX*.

To run *BibTeX* choose *BibTeX* from the drop-down menu in *TeXworks* (the one that normally says *pdfLaTeX*).

This is a lot to swallow at once. Let's go one step at a time

Bibliography
The `bib` file(s)

- The bibliography databases are plain text files
- You can create them with TeXworks (or any plain text editor):
    - *File > New*
    - Enter the bibliography information (bellow)
    - *File > Save as . . .*
    - In the *Save* dialogue box make sure
        1. to save in the same folder as you LaTeX document
        2. to choose *BibTeX databases (*.bib)* from the *Files of Type:* drop-down menu.
- Continue adding entries to your database, as needed (simply "Save" now)

Note: Don't do this in your computer quite yet!
We had before, in `thebibliography`:

```
\bibitem{albert1}
On the Relation between the Expansion and the Mean
Density of the Universe, A. Einstein, W. de Sitter,
Proc.Nat.Acad.Sci. 18 (1932) 213-214
```

Instead we write into the new file (I call `samplebib1.bib`)

```
@article{albert1,
  author = "Einstein, A. and  de Sitter, W.",
  year = "1932",
  title = "On the Relation between the
   Expansion and the Mean Density of the Universe",
  journal = "Proc.Nat.Acad.Sci.",
  volume = "18",
  pages = "213--214"
}
```

And for

```
\bibitem{einsteinRosen}
On Gravitational waves,
Albert Einstein, N. Rosen, J.Franklin
    Inst. 223 (1937) 43-54
```

we enter (best practice: leave a blank line after the previous)

```
@article{einsteinRosen,
  author = "Einstein, Albert and Rosen, N.",
  year = "1937",
  title = "On Gravitational waves",
  journal = "J.Franklin Inst.",
  volume = "223",
  pages = "43--54"
}
```

In practice we want to avoid all that typing. Do this:

1. In TeXworks: New Document
2. Browser (eg, Safari): Find an article in *Google scholar*
3. Under article abstract click *Cite*
4. In Cite window choose *BibTeX*
5. Copy and paste
6. "Save as ..." *BibTeX database (*.bib)*
   Make sure to save in same folder as your LaTeX document

I will call this new database `samplebib1.bib`

Next edit the label for the reference. In our previous example,

```
@article{einstein1937gravitational,
  title = "On gravitational waves",
 .....
```

we would change the first line to:

```
@article{einsteinRosen,
```

Of course, in your work you choose the reference and the label.

Next add one more entry to your database. In our example above we would look for *einstein de sitter On the Relation between the Expansion and the Mean Density of the Universe* and paste into our file

```
@article{einstein1932relation,
  title = "On the Relation between the Expansion and
           the Mean Density of the Universe",
  author= " Einstein, Albert and De Sitter, Willem",
...}
```

and edit the first line to read

```
@article{albert1,
```

But you choose an article and a label on your own.

SAVE. Don't forget to SAVE.

Edit your LaTeX document.

1. Remove `thebibliography`
2. Include in its stead:
   `\bibliographystyle{plain}`
   `\bibliography{samplebib1}`
   before `\end{document}`
3. Include `\cite{`*label*`}` in your text (with the *label*s you used in your bib-database
4. Typeset with *pdfLaTeX*. Then run *BibTeX*. Then typeset twice with *pdfLaTeX*.

To run *BibTeX* choose *BibTeX* from the drop-down menu in *TeXworks* (the one that normally says *pdfLaTeX*).

- The argument `plain` in `\bibliographystyle` is a BibTeX *style*.
- Change that for `alpha` and typeset again (four steps!).
- And again with `abbrv`.

A list of styles with samples can be found in
`https://www.overleaf.com/learn/latex/Bibtex_bibliography_styles`
A more comprehensive listing in
`http://www.reed.edu/cis/help/LaTeX/bibtexstyles.html`

You may use more than one database:

```
\bibliography{samplebib1,samplebib2}
```

If there are repeated entries in the databases, when you run BibTeX the Console Output gives:

```
Repeated entry---line 1 of file samplebib2.bib
 : @article{albert1
 :                        ,
I'm skipping whatever remains of this entry
Repeated entry---line 12 of file samplebib2.bib
 : @article{einsteinRosen
 :                            ,
I'm skipping whatever remains of this entry
(There were 2 error messages)
```

Clean up your databases to avoid this! BibTeX is smart: you can still typeset the document. It will use the first instance of the repeated references. Try it (by duplicating your database into a "new" one).

Beyond Google Scholar:

Major journals give you the full reference in BibTeX format. Example:

1. Go to `www.sciencemag.org`
2. Navigate to any particular journal (eg *Science*) and then a particular article
3. On right column: Article Tools, choose *Citation Tools*
4. Form list of Formats choose *BibTeX*
5. Gives plain text file to save to your computer. Open file. Copy and paste to your database

You are out of luck with Nature. But fine with many journals, including American Physical and Chemical Societies journals, IEEE journals, and more

Bibliography

There are more *fields* you can specify. Template:

```
@article{Xarticle,
    author    = "",
    title     = "",
    journal   = "",
    volume    = "",
    number    = "",
    pages     = "",
    year      = "XXXX",
    month     = "",
    note      = "",
}
```

- Author names are `Last, First`
- Multiple authors are separated by `and`.
- Year has to be four digits.
- Fields that are not used should not appear (do not use %
  to comment unwanted fields).

`article` is an *entry type*. There are many others. One that you may use frequently is

```
@book{Xbook,
    author    = "",
    title     = "",
    publisher = "",
    volume    = "",
    number    = "",
    series    = "",
    address   = "",
    edition   = "",
    year      = "XXXX",
    month     = "",
    note      = "",
}
```

Here is an example copied from an entry in Google Scholar:

```
@book{weinberg1993first,
  title={The first three minutes: a modern
    view of the origin of the universe},
  author={Weinberg, Steven},
  year={1993},
  publisher={Basic Books}
}
```

Note the syntax: $\{\cdots\}$ instead of "$\cdots$" is allowed.

A bit more math never hurts

To use the powerful *amsmath* package, insert, in the preamble:

`\usepackage{amsmath}`

(that is, after `documentclass`). Write and equation using the `equation` environment, and replace `equation*` for `equation` in the environment name.

The same can be accomplished using `\nonumber` after the equation in the `equation` environment (with or without the `amsmath` package).

Get a copy of the *amsmath user's guide* from *ftp.ams.org/pub/tex/doc/amsmath/amsldoc.pdf*. There are lots of goodies there, some of which we explore next.

Math

If an expression is too long to fit in one line:

```
\begin{multline}
a+b+c+d+e+f+g+h+i+j+k+l+m\\
+n+o+p+q+r+s+t+u+v+w+x+y+z
\end{multline}
```

$$a + b + c + d + e + f + g + h + i + j + k + l + m$$
$$+ n + o + p + q + r + s + t + u + v + w + x + y + z \quad (1)$$

Now try this with `equation` replacing `multline`. It ignores the line break. If you look at the console it says
`Overfull \hbox (0.15474pt too wide) detected at line 20.`
There is a no-equation-number version for this too:

```
\begin{multline*}
a+b+c+d+e+f+g+h+i+j+k+l\\
+m+n+o+p+q+r+s+t+u+v+w+x+y+z
\end{multline*}
```

You may want to display several equations at once. One on top of the next (as many as you want):

```
\begin{equation}
\begin{split}
a+b&=c\\
e+f&=g
\end{split}
\end{equation}
```

The ampersand before the equal sign, `&=`, sayas, "align at equal sign." The ampersand is used in many LaTeX constructions for alignment.

Notice this comes with one equation number for the whole block so we give it only one label.

```
\begin{equation}
\label{manyeqs}
\begin{split}
a+b&=c\\
e+f&=g
\end{split}
\end{equation}
The equations (\ref{manyeqs}) whatever.
```

For separately numbered aligned equations,

```
\begin{align}
\label{manyeqs1}
a+b&=c\\
\label{manyeqs2}
e+f&=g
\end{align}
The equations (\ref{manyeqs1}) and (\ref{manyeqs2}) are ...
```

Here is a more complicated situation. You want two equations per line, aligning the two sets of equations:

$$a + b = c \qquad\qquad l = h + n \qquad (2)$$
$$e + f = g \qquad\qquad o + p + q = r \qquad (3)$$

Here is how:

```
\begin{align}
\label{manyeqs1}
a+b&=c    &   l&=h+n \\
\label{manyeqs2}
e+f&=g    &    o+p+q&=r
\end{align}
The equations (\ref{manyeqs1}) and (\ref{manyeqs2}) ...
```

Note the additional & to separate the two equations in one line.

Exercise

Typeset this

$$a = b \qquad c = d \qquad e = f$$
$$g = b \qquad h = d \qquad k = f$$

```
\begin{align*}
a&=b & c&=d & e&=f \\
g&=b & h&=d & k&=f
\end{align*}
```

Note: the ∗ version gives no equation numbers.

You can still add a label to these equations:

$$a + b = c \qquad\qquad l = h + n \qquad (4)$$

$$e + f = g \qquad\qquad o + p + q = r \qquad (5)$$

The equations (4) and (5) . . .
Here is how:

```
\begin{align}
\label{manyeqs1}
a+b&=c & l & = h + n \\
\label{manyeqs2}
e+f&=g& o+p+q &=r
\end{align}
The equations (\ref{manyeqs1}) and (\ref{manyeqs2}) ...
```

There are a few other constructions. See the *amsmath user's guide.*

Getting tired of writing parenthesis around \ref{eqlabel}?
With amsmath you can use instead \eqref{eqlabel}, as in

```
The equations \eqref{manyeqs1} and \eqref{manyeqs2} ...
```

We need to raise to powers and so on:

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

$$E = mc^2 \tag{6}$$

Oopsie!:

```
\[ a=x^12 \]
```

$$a = x^1 2$$

Instead we need grouping:

```
\[ a=x^{12} \]
```

$$a = x^{12}$$

For subscripts,

```
\begin{equation}
a_1=x_{12}
\end{equation}
```

$$a_1 = x_{12} \qquad (7)$$

You can combine them:

```
\begin{equation}
a_1=x_{12}^5
\end{equation}
```

$$a_1 = x_{12}^5 \qquad (8)$$

Exercise: Typeset this:

$$a^2 = b^2 + c^2$$

Solution

```
\[
a^2=b^2+c^2
\]
```

More Math

We use a lot of greek in math. So we need:

```
\begin{equation}
\alpha + \beta = \gamma+\delta
\end{equation}
```

and

```
\begin{equation}
A + B = \Gamma+\Delta
\end{equation}
```

There is no `\Alpha` nor `\Beta` because they are the same symbols as `A` and `B`.

For some symbols there is a variant alternative:

```
\begin{equation}
\epsilon+\varepsilon = \theta+\vartheta=\phi+\varphi
\end{equation}
```

See the complete list of math symbols in the Wikibook.

Some symbols on the keyboard work in math mode!:

```
a+n!-b/c=[a*(bc)] < d > e'=|g|
```

But you may want greater-than-or-equal sign, as in $a \geq b$. Try

```
a \ge b \le c
```

There is an equivalent, sometimes easier to remember,

```
a \geq b \leq c
```

We also have

```
a \equiv b \ll c \gg d \sim g \neq h
h \propto k\approx z\times w \pm x \mp z
```

You will memorize many of these as you use them. No point in memorizing them all now. Consult the complete list of math symbols in the Wikibook.

Many (many!) symbols that cannot be found in the basic set in can be found elsewhere. A good place to start is the amssymb package. Load it:

`\usepackage{amssymb}`

and try the following (in math mode!):

`a \gtrsim b \lesssim c`

Consult the AMS math list of symbols in http://authortools.aas.org/aastex/aassymbols.pdf for more symbols. Really, go there and try some. Here are a few
`\aleph` $\aleph$ `\beth` $\beth$, `\gimel`, $\gimel$
`\otimes` $\otimes$, `\times` $\times$, `\div` $\div$, `\cap` $\cap$, `\cup` $\cup$
`\infty` ∞, `\forall` $\forall$, `\partial` ∂, `\hbar` $\hbar$

But if you cannot find it in amssymb, then what. Almost anything you can dream of is available. Consult *and play with* the **Comprehensive LaTeX Symbol List** in mirrors.ctan.org/info/symbols/comprehensive/symbols-letter.pdf. You will find all sorts of weird stuff, and sometimes several packages that do similar things. For example, the packages `wasysym`, `marvosym` and `mathabx` all supply astronomical symbols. Let's try it:

```
...
\usepackage{wasysym}
...
\mercury, \earth
...
```

These work both in text and in math mode. Try both.

Alert student Zaid Mansuri points to another solution: Detexify is a web site (at `http://detexify.kirelabs.org/classify.html`) that allows you to scribble a symbol, then attempts at recognizing it and offers several possible LaTeX versions.

Try it! Typeset two of these: ♀, ♂, ♉, ⊟ ə

The message is that other than the common symbols that you will memorize from repeated use, other symbols are easy to find in all these resources.

Fractions

Both in inline and in display modes, but with different results:

`\frac{a}{b}` $\dfrac{a}{b}$

You can also have one in another

`\frac{\frac{a}{b}}{\frac{c+d}{d+e}}` $\dfrac{\frac{a}{b}}{\frac{c+d}{d+e}}$

Grouping is not required

`\frac12=\frac24=\frac{a}b=\frac{a}{b}`

But the following will give an error message

`\fracab=\fraca{b}`

Typeset these equations:

$$F = G_N \frac{m_1 m_2}{r^2}$$

$$n_\pm(E,T) = \frac{1}{e^{\frac{E}{k_B T}} \pm 1} = \frac{1}{e^{\hbar\omega/k_B T} \pm 1}$$

$$F_{\mu\nu} = [D_\mu, D_\nu] = \partial_\mu A_\nu - \partial_\nu A_\mu = \partial_{[\mu} A_{\nu]}$$

Solutions:

```
\[
F = G_N\frac{m_1m_2}{r^2}
\]
\[
n_{\pm}(E,T)=\frac1{e^{\frac{E}{k_BT}}\pm1}
           =\frac1{e^{{\hbar\omega}/{k_BT}}\pm1}
\]
\[
F_{\mu\nu} =  [D_\mu ,  D_\nu]
   =\partial_\mu A_\nu-\partial_\nu A_\mu
 =\partial_{[\mu} A_{\nu]}
\]
```

Big and small: note the different typesetting of fraction in inline vs display modes: \(\tfrac12\) vs \[\tfrac12\].

Sometimes we want a small faction in display mode. Use tiny fraction, \tfrac. Compare

\[\frac12a+b\]

with

\[\tfrac12a+b\]

Derivatives are just fractions:

`\frac{df}{dt}`

For partial derivatives we need a curly "d"

`\frac{\partial f}{\partial t}`

Integrals, sums and products:

`\int f(x) dx, \sum x_n, \prod \omega_k`

These also display differently in inline/display modes.
Ranges displayed:

`\int_0^1 f(x)dx, \sum_{n=0}^7 x_n,  \prod_1^{10} \omega_k`

Notice the different placement of the limits in display *vs* inline modes.

Typeset: "Taylor expansion $e^x = \sum_{n=0}^{\infty} \frac{1}{n!} x^n$."

$$\int_0^1 \frac{df}{dx} dx = f(1) - f(0)$$

$$e^{\zeta(s)} = \prod_{n=1}^{\infty} e^{1/n^s}$$

(This uses the greek letter zeta)

Solutions:

``Taylor expansion $e^x=\sum_{n=0}^\infty \frac1{n!}x^n$.''

\[\int_{0}^1 \frac{df}{dx}dx= f(1)-f(0)\]

\[e^{\zeta(s)}=\prod_{n=1}^\infty e^{1/n^s}\]

Bad:

`(\frac12 a+\frac{x+y}{z+d})` $(\dfrac{1}{2}a + \dfrac{x+y}{z+d})$

Better:

`\left(\frac12 a+\frac{x+y}{z+d}\right)`

Other delimiters:

`( a ), [ b ], \{ c \}, | d |, \| e \|, \langle f \rangle`

$$(a), [b], \{c\}, |d|, \|e\|, \langle f \rangle$$

The last one is very common to denote averages. They all get resized and don't have to match:

`\left\langle\frac12 a+\frac{x+y}{z+d}\right]`

Even more math

You may want

$$\frac{df}{dt}\bigg|_{t=0}$$

`\left. \frac{df}{dt} \right|_{t=0}`

Exercise: typeset

$$2\left[3\frac{a}{z} + 2\left(\frac{a}{d} + 7\right)\right]$$

$$x^2\left(\sum_n A_n + 3\left(b + \frac{1}{c}\right)\right)\bigg]_0$$

Solution:

```
\[ 2\left[3\frac{a}{z}+
      2\left(\frac{a}{d}+7\right)\right] \]
\[ \left.x^2\left(\sum_nA_n
      +3\left(b+\frac1c\right)\right)\right]_0 \]
```

If you try

```
\begin{multline}
\left(\frac12 a\\
+\frac{x+y}{z+d}\right)
\end{multline}
```

you get the error message

```
! Missing \right. inserted.
<inserted text>
                \right .
l.34 \end{multline}
```

Instead we need

```
\begin{multline}
\left(\frac12 a \right. \\
\left.+\frac{x+y}{z+d}\right)
\end{multline}
```

This works in any environment that goes over many lines. For example:

```
\begin{align*}
\left(\frac12  a \right. & \\
&\left. +\frac{x+y}{z+d}\right)
\end{align*}
```

Note that the alignment symbol cannot be inside the delimiters. The following gives an error message:

```
\begin{align*}
\left(\frac12 & a \right. \\
&\left. +\frac{x+y}{z+d}\right)
\end{align*}
```

Exercise: Typeset this, using `multline*`

$$2\left(1 + \frac{1}{2} + \frac{1}{2^2} + \frac{1}{2^3} + \frac{1}{2^4} + \frac{1}{2^5} + \frac{1}{2^6} + \frac{1}{2^7} + \frac{1}{2^8} + \frac{1}{2^9}\right.$$
$$\left. + \frac{1}{2^{10}} + \frac{1}{2^{11}}\right) = \frac{4095}{1024}$$

Again, LaTeX generally makes the right decision on how to resize delimiters, but sometimes you want to override it:

`( \big( \Big( \bigg( \Bigg(`

Now that you have seen it, where would you use it? Here is one example:

`(a+f(x)),\quad \left(a+f(x)\right), \quad \big(a+f(x)\big)`

gives

$$(a + f(x)), \quad (a + f(x)), \quad \big(a + f(x)\big)$$

Exercise: we previously had

```
\[ 2\left[3\frac{a}{z}+
    2\left(\frac{a}{d}+7\right)\right] \]
```

giving

$$2\left[3\frac{a}{z}+2\left(\frac{a}{d}+7\right)\right]$$

Make it look like this:

$$2\left[3\frac{a}{z}+2\left(\frac{a}{d}+7\right)\right]$$

Solution:

```
\[
   2\Bigg[3\frac{a}{z}+
      2\bigg(\frac{a}{d}+7\bigg)\Bigg]
\]
```

The alignment character is &.

```
\documentclass[11pt]{article}
\usepackage{amsmath}

\begin{document}
\[
\begin{matrix}
  a & b & c \\
  d & e & f \\
  g & h & i
 \end{matrix}
\]
\end{document}
```

gives a matrix without delimiters.

For standard delimiters:

```
\begin{pmatrix}
  a & b & c \\
  d & e & f \\
  g & h & i
 \end{pmatrix}
```

You can have several and do stuff with them, eg

```
2A+3\times\begin{pmatrix}
  a & b & c \\
  d & e & f \\
  g & h & i
 \end{pmatrix}
=0
```

For a determinant

```
\begin{vmatrix}
  a & b & c \\
  d & e & f \\
  g & h & i
 \end{vmatrix}
```

You can use other delimiters by use of the \left and \right commands:

```
\left(\begin{matrix}
  a & b & c \\
  d & e & f \\
  g & h & i
 \end{matrix}\right]
```

Exercise: typeset

The Pauli matrices are:

$$\sigma^1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma^2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \text{and} \quad \sigma^3 = \begin{pmatrix} 1 & \\ 0 & -1 \end{pmatrix}$$

Note for those of you that may know about Pauli matrices: There is a deliberate typo inserted in the last matrix, a blank instead of a zero in the second element of the first row — this is not an oversight but intended to challenge you to typeset this.

Use of \cdots (horizontal), \vdots (vertical)
and \ddots (diagonal):

```
A =
 \begin{pmatrix}
  a_{11} & a_{12} & \cdots & a_{1n} \\
  a_{21} & a_{22} & \cdots & a_{2n} \\
  \vdots  & \vdots  & \ddots & \vdots  \\
  a_{m1} & a_{m2} & \cdots & a_{mn}
 \end{pmatrix}
```

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

Alignment
Extra space

Sometimes LaTeX does not calculate space correctly:

```
M = \begin{bmatrix}
      \frac{5}{6} & \frac{1}{6} & 0           \\
      \frac{5}{6} & 0           & \frac{1}{6} \\
      0           & \frac{5}{6} & \frac{1}{6}
    \end{bmatrix}
```

$$M = \begin{bmatrix} \frac{5}{6} & \frac{1}{6} & 0 \\ \frac{5}{6} & 0 & \frac{1}{6} \\ 0 & \frac{5}{6} & \frac{1}{6} \end{bmatrix}$$

To fix this add extra space:

```
M = \begin{bmatrix}
      \frac{5}{6} & \frac{1}{6} & 0           \\[0.3em]
      \frac{5}{6} & 0           & \frac{1}{6} \\[0.3em]
      0           & \frac{5}{6} & \frac{1}{6}
    \end{bmatrix}
```

Here `em` is a distance measure, roughly equal to the width of the letter `m`. So the space we have added is `0.3em`, that is 30% of the width of `m`.

$$M = \begin{bmatrix} \frac{5}{6} & \frac{1}{6} & 0 \\ \frac{5}{6} & 0 & \frac{1}{6} \\ 0 & \frac{5}{6} & \frac{1}{6} \end{bmatrix}$$

The construction \\[*distance*] can be used also in line breaks.

Type the following in text mode:

```
Here is a short line\\[1in]
and the new line of text continues one inch below.
```

Alignment

Sometimes we need finer alignement or finer spacing control or insertion of guide lines in a matrix. For this there is

```
\begin{array}{ccc}
  -a & -b & c \\
  d & e & f \\
  g & h & i
 \end{array}
```

The command takes an argument, in this case `ccc` telling LaTeX to center justify the columns. Try instead

```
\begin{array}{rrr}
  -a & -b & c \\
  d & e & f \\
  g & h & i
 \end{array}
```

for right justification of the columns. You can guess `lll` left justifies.

You can mix these:

```
\begin{array}{rlc}
  -a & -b & c \\
  d & e & f \\
  g & h & i
 \end{array}
```

Placing a vertical separator line is easy:

```
\begin{array}{r|rr}
  -a & -b & c \\
  d & e & f \\
  g & h & i
 \end{array}
```

You can place several vertical separator lines. Change the above to {|r|r|r|}, or to {|r||r||r|}, etc.

For horizontal lines use `\hline`:

```
\begin{array}{r|rr}
  -a & -b & c \\
\hline
  d & e & f \\
\hline
  g & h & i
 \end{array}
```

The spacing is not quite right. A crutch around this is to add an extra blank line with a line break followed by negative space. We adjust the amount of negative space by trial an error:

```
\begin{array}{r|rr}
  -a & -b & c \\
\hline\\[-1em]
  d & e & f \\
\hline\\[-1em]
  g & h & i
 \end{array}
```

Typeset this:

$$\left\|\begin{array}{c|c} a \times b & c + d \\ \alpha & \gamma \\ \hline 3 & 1.1 \end{array}\right\|$$

Tables look like matrices without delimiters, with text entries. Basically the same as `array` but with text entries (and in text mode). In the last example environment→`tabular`, add few vertical/horizontal lines, remove displayed equation:

```
\documentclass[11pt]{article}
\begin{document}
Here is a table:
\begin{tabular}{r||r|r}
  -a & -b & c \\
\hline\hline
  d & e & f \\
\hline
  g & h & i
\end{tabular}
\end{document}
```

A more interesting table has actual text in it and occupies its own paragraph:

```
\documentclass[11pt]{article}
\begin{document}
Here is a table:

\begin{tabular}{c||l|l}
  header 1 & header 2 & header 3 \\
\hline\hline
  entry 1-1 & entry 1-2 & entry 1-3 \\
\hline
  entry 2-1 & entry 2-2 & entry 2-3 \\
\hline
  entry 3-1 & entry 3-2 & entry 3-3
\end{tabular}
\end{document}
```

Here is what the output of the above code looks like:

Here is a table:

header 1	header 2	header 3
entry 1-1	entry 1-2	entry 1-3
entry 2-1	entry 2-2	entry 2-3
entry 3-1	entry 3-2	entry 3-3

A more interesting table has actual text.

Exercise. Typeset this:

Jersey	First Name	Last Name
10	Cristiano	Ronaldo
11	Didier	Drogba

To get it centered across the page use the `center` environment, *i.e.*, \begin{center} ... \end{center}

Answer.

```
\begin{center}
\begin{tabular}{c||l|l}
  Jersey & First Name & Last Name \\
\hline\hline
  10 & Cristiano & Ronaldo \\
\hline
  11 & Didier & Drogba
\end{tabular}
\end{center}
```

To see that column widths adjust automatically:

Exercise. Add to the previous, to typeset this:

Jersey	First Name	Last Name
10	Cristiano	Ronaldo
10	Edson	Arantes do Nascimento (Pele)
11	Didier	Drogba

Answer.

```
\begin{center}
\begin{tabular}{c||l|l}
  Jersey & First Name & Last Name \\
\hline\hline
  10 & Cristiano & Ronaldo \\
  \hline
10 & Edson & Arantes do Nascimento (Pele)\\
\hline
  11 & Didier & Drogba
\end{tabular}
\end{center}
```

Replacing the alignment character by p{*dimension*} gives a left-right justified column of fixed width of the given dimension:

```
\begin{tabular}{c||l|p{1in}}
  Jersey & First Name & Last Name \\
\hline\hline
  10 & Cristiano ...
       ... Drogba
\end{tabular}
```

Jersey ‖	First Name	Last Name
10	Cristiano	Ronaldo
10	Edson	Arantes do Nascimento (Pele)
11	Didier	Drogba

To add a line break within a cell you use `\newline`
(obviously `\\` won't work):

```
\begin{center}
\begin{tabular}{c||l|p{1in}}
  Jersey & First Name & Last Name \\
\hline\hline
  ...
10 & Edson & Arantes\newline do Nascimento  (Pele)
  ...
\end{tabular}
\end{center}
```

Jersey	First Name	Last Name
10	Cristiano	Ronaldo
10	Edson	Arantes do Nascimento (Pele)
11	Didier	Drogba

Sometimes you want lines across some columns but not all:

```
\begin{tabular}{c||l|p{1in}}
  Jersey & First Name & Last Name \\
\hline\hline
  10 & Cristiano & Ronaldo \\
\cline{2-3}
 & Edson & Arantes\newline do Nascimento (Pele)\\
\hline
  11 & Didier & Drogba
\end{tabular}
```

Jersey	First Name	Last Name
10	Cristiano	Ronaldo
	Edson	Arantes do Nascimento (Pele)
11	Didier	Drogba

.

Sometimes you want lines across some columns but not all:

```
\begin{tabular}{c||l|p{1in}}
  Jersey & First Name & Last Name \\
\hline\hline
  10 & Cristiano & Ronaldo \\
\cline{2-3}
 & Edson & Arantes\newline do Nascimento (Pele)\\
\hline
  11 & Didier & Drogba
\end{tabular}
```

\cline's argument: range of columns for horizontal line.
Same start and end of range gives only one column line.
E.g., \cline{2-2} instead of \cline{2-3} column 2 only.

Breather: Good coding practice

Without change to the output, you can code the above as

```
\begin{center}\begin{tabular}{c||l|p{1in}}Jersey&First
Name&Last Name\\\hline\hline10&Cristiano&Ronaldo\\
\cline{2-3}&Edson&Arantes\newline do Nascimento
(Pele)\\\hline 11&Didier&Drogba\end{tabular}\end{center}
```

You decide which is clearer. Think possible errors and debugging!

Exercise: Typeset

Shape	Area	Perimeter
Disk of radius R	πR^2	$2\pi R$
Rectangle of sides L_1 and L_2	$L_1 L_2$	$2(L_1 + L_2)$
Square of side $L_1 = L_2$		
Right triangle, base b and height h	$\frac{1}{2}bh$	$b + h + \sqrt{b^2 + h^2}$

Solution:

```
\begin{center}
\begin{tabular}{|p{2in}|c|c|}
Shape&Area&Perimeter\\
\hline\hline
Disk of radius $R$ &$\pi R^2$ & $2\pi R$\\
\hline
Rectangle of sides $L_1$ and $L_2$ & $L_1L_2$&$2(L_1+L_2)$\\
\cline{1-1}
Square of side $L_1=L_2$ & & \\
\hline
Right triangle, base $b$ and height $h$ & $\frac12bh$&$b+h-
\end{tabular}
\end{center}
```

Opposite: Vertical line that does not span all rows

We want, *e.g.*, to include a single "title" line, like this:

Dream Team		
Jersey	First Name	Last Name
10	Cristiano	Ronaldo
	Edson	Arantes
		do Nascimento
		(Pele)
11	Didier	Drogba

For this we use `\multicolumn`

`\multicolumn` has three arguments:

1. number of columns
2. alignment, `l`, `c` or `r`, with possible vertical bars
3. the text

```
\begin{center}
\begin{tabular}{|c||l|p{1in}|}
\hline
\multicolumn{3}{|c|}{Dream Team}\\
\hline
  Jersey & First Name & Last Name \\
\hline\hline
  10 & Cristiano & Ronaldo \\
\cline{2-3}
 & Edson & Arantes\newline do Nascimento  (Pele)\\
\hline
  11 & Didier & Drogba\\
\hline
\end{tabular}
\end{center}
```

As with `array` we have an issue of separation between horizontal lines and text.

We can use our previous trick of adding a blank line plus negative space: use `\\[-1em]` after `\hline` or `\cline`, *e.g.*,

```
\begin{tabular}{|c||l|p{1in}|}
\hline\\[-1em]
\multicolumn{3}{|c|}{Dream Team}\\
\hline\\[-1em]
  Jersey & First Name & Last Name \\
\hline\hline\\[-1em]
  10 & Cristiano & Ronaldo \\
\cline{2-3}\\[-1em]
...
\end{tabular}
```

Alignment

1. Within each cell in a table you can use any environment.
2. The column containing the environment *must* be justified by p{*dimension*}
3. Useful, for example, in typing tables with formulae

```
\begin{center}
\begin{tabular}{c|p{2in}}
  Newton & \begin{equation*} \vec F=m\vec a\end{equation*} \\
\hline
  Einstein & \begin{equation} E=mc^2\end{equation}
\end{tabular}
\end{center}
```

Newton	$\vec{F} = m\vec{a}$
Einstein	$E = mc^2$ (9)

This is not pretty, but works. For simple equations it is easier to use the inline-text equation. The following looks better:

```
\begin{center}
\begin{tabular}{c|c}
  Newton & $ \vec F=m\vec a$ \\
\hline\\[-1em]
  Einstein &$E=mc^2$
\end{tabular}
\end{center}
```

Newton	$\vec{F} = m\vec{a}$
Einstein	$E = mc^2$

Alignment
at decimal point, and other separators

It does not look pretty:

```
\begin{center}
\begin{tabular}{c}
2.30\\                              2.30
100.23\\                          100.23
10.12                             10.12
\end{tabular}
\end{center}
```

Instead use:

```
\begin{center}
\begin{tabular}{r@{.}l}
2&30\\                             2.30
100&23\\                         100.23
10&12                            10.12
\end{tabular}
\end{center}
```

Here is what's going on. We used two columns, one right justified and the next left justified. The separator is not a vertical line but whatever goes in @{*separator symbol*}, in this case the separator symbol being a period.

Exercise: Typeset this (note the alignment at equal sign)

a	$x^2 + y = 30$
b	$100 = \sin(\theta) + \cos\varphi$
c	$q \cup p = q \cap p$

Exercise: Typeset this (note the alignment at equal sign)

a	$x^2 + y = 30$
b	$100 = \sin(\theta) + \cos\varphi$
c	$q \cup p = q \cap p$

Solution:

```
\begin{center}
\begin{tabular}{|l|r@{~$=$~}l|}
\hline
a&$x^2+y$&30\\\hline
b&100&$\sin(\theta)+\cos\varphi$\\\hline
c&$q\cup p$&$q \cap p$\\\hline
\end{tabular}
\end{center}
```

Many more packages `array`, `tabularx`, `tabulary`, ...
You know where to find more details. Example using using the
switch `\rowcolors` from the package `xcolor`, option `table`:

```
\documentclass[11pt]{article}
\usepackage[table]{xcolor}
...
\begin{document}
...

{ \rowcolors{1}{yellow}{pink}
\begin{tabular}{|c||l|p{1in}|}
\hline
\multicolumn{3}{|c|}{Dream Team}\\
\hline
  Jersey & First Name & Last Name \\
\hline\hline
  10 & Cristiano & Ronaldo \\
\cline{2-3}
 & Edson & Arantes\newline do Nascimento  (Pele)\\
\hline
  11 & Didier & Drogba\\
\hline
\end{tabular} }
...
```

- Think of a *float* as a frame into which you can put something
- Put what? A table, a picture (an image), ...
- It *floats*: it finds a place to land in your document.
- LaTeX decides where
- You can nudge it (even insist) into placing it where you want.
- LaTeX is designed to make a good decision: give it a chance

Floods and Graphics
The table environment

Use this to

1. automatically place the table
2. automatic table number referencing
3. include a caption.

```
Here is some text. In table~\ref{mytable}
we show our Dream Team
\begin{table}[b]
\begin{center}
\begin{tabular}{|c||l|p{1in}|}
...
\end{tabular}
\end{center}
\caption{ \label{mytable}
This table shows blah blah}
\end{table}
```

```
Here is some text. In table~\ref{mytable}
we show our Dream Team
\begin{table}[b]
...
\caption{ \label{mytable} This table ...}
\end{table}
```

- `table` takes one *positioning* argument
- `[b]` *place this table at the bottom of a page, close to here*
- Also: `h` for *here*, and `t` for *at the top of the page.*
- `t` is the most commonly used.

- Argument of `\caption` is text.
- You can insert a label and refer to it (just as for equations)
- The counters for numbering tables and equations are independent

Experiments:

1. Paste a lot of text into your document, enough for a couple of pages of typeset material, at least 6 good paragraphs. (*Hint*: Find one good paragraph, copy it into the buffer, and paste it many times into your document).
Then insert your *Dream Team Table* between paragraphs 2 and 3. Include a caption with a `\label{dreamteam}` (you provide the text). Insert `\ref{dreamteam}` somewhere in the text before and again after where you inserted the table. Typeset once with each of positioning `b`, `t` and `h`.

2. Copy the table and caption and paste into the space between paragraphs 4 and 5. Typeset. Check console (warning on repeated labels).
Change label of second table: `\label{dreamteam2}`. Insert a few `\ref{dreamteam2}` somewhere in the text before and again after where you inserted the table.

Sometimes in a very long document you want a list of tables showing where to find each table. Just add `\listoftables`. Add this now to the previous example, just before your end doc:

```
\newpage
\listoftables
\end{document}
```

Floats and Graphics

To insert graphic files we need a package
Use this `\usepackage{graphicx}` in the preamble.
Command for inserting a graphic file (a picture)

`\includegraphics[`*attributes*`]{`*filename*`}`

1. Go to course web page:
   courses.physics.ucsd.edu/2021/Spring/physics87/
2. Notes > Exercises > UCSD Tritons image gl-5-triton.png
3. On toolbar (Safari): File > Save as . . .
4. Select folder with your LaTeX work, then Save; retain file name
5. Notice: `*.png`
6. In your LaTeX document insert:
   1. `\usepackage{graphicx}` in the preamble.
   2. `\includegraphics{gl-5-triton.png}` between paragraphs.
7. Typeset

Floats and Graphics

After we put `\usepackage{graphicx}` in the preamble and typeset our document with the figure we get this:

Oops! Too large. Next try instead

`\includegraphics[width=2in]{gl-5-triton.png}`

1. The attribute `width=`*dimen* sets the width of the image to the desired dimension.
2. This and `height=`*dimen* are the most common. There are a few others, like:
   - `scale=`*factor* amplifies the image by *factor* (reduces it if factor is smaller than 1)
   - `trim=l b r t` used for cropping by lengths l, b, r and t, from the left, bottom right and top, respectively.

For `trim` to work you have to add the attribute `clip` (or rather set `clip=true` but adding just `clip` works):

`\includegraphics[trim= 8cm 0cm 0cm 0cm,clip ]{gl-5-triton.png}`

See the complete list of attributes in the WikiBook.

Exercise: Resize and crop the Triton image to get this:

Exercise: Resize and crop the Triton image to get this:

Solution:
`\includegraphics[width=4cm,trim= 7cm 6cm 8cm 1cm,clip]{gl-5-triton.png}`

Try this:

`\includegraphics[width=2in, height=4in]{gl-5-triton}`

The extension is not necessary: LaTeX knows to look for `.png` files as graphic files. In fact, if you are typesetting with *pdfLaTeX*, and we are, and are using the *graphicx* package, and we are, then LaTeX looks for files with extensions `.jpg`, `.png` and `pdf`, and assumes that if you did not write an extension that it will find the full file with extension in the current directory (and looks for all three extensions).

Floats and Graphics
Graphics folder

1. Convenient to have a folder (ie, a directory or subdirectory) containing your graphics files

2. Create a folder `images` in the folder with your `filename.tex`

3. Move the file `gl-5-triton.png` to that folder.

4. Then typeset with

   `\includegraphics{images/gl-5-triton}`

   (or, linux gurus, you can also use this
   `\includegraphics{./images/gl-5-triton}`)

5. There are subtleties arising in including blank spaces in the name of the folder or the document, so better don't!

The `figure` environment is analogous to the table environment. It is also a float. Without further ado:

```
\begin{document}
Some text here.

\begin{figure}[t]
\begin{center}
\includegraphics[width=2in]{images/gl-5-triton}
\end{center}
\caption{\label{tritonLogo} This is a UCSD logo that displa
  mascot, a Triton.}
\end{figure}

In figure~\ref{tritonLogo} one finds a Triton. This
is the symbol of \dots
\end{document}
```

Just as with tables:

`\listoffigures`

You can put this anywhere in your document, but it is often found eiher at the very beginning together with the table of contents, or at the end.

More: We will not do this in the course but you may want to play with putting boxes around images, text wrapping, inserting text on top of an image, including full pdf pages, and more. Consult the Wikibook. Play with it! There is also a lot of information there about converting among different formats abd editing graphics.

If we want to separate two expression in display mode, use
`\quad` or `\qquad`

`a=g, \quad c=z, \qquad d=0`

This is particularly useful with a little text, but this looks bad:

`a=g\quad then implies \quad g=2`

which gives $a = g \quad thenimplies \quad g = 2$. We need instead the
`\text` command:

`a=g\quad \text{then implies} \quad g=2`

Best to let LaTeX decide. But if you must, in *math mode* we have a full space `~`, a small space `\;`, a smaller space `\:`, and an even smaller space `\,`:

```
a~a\;a\:a\,a a~~~a    ~    a
```

Note that white spaces are ignored!
You can also subtract a bit of space, using `\!`.
You can use it repeatedly:

```
a\!a~a\!\!a
```

You can see that normally you want to stay away from this.
Going back to the example with integrals we re-write:

```
\int_0^1 \!\!f(x)\, dx,\qquad \sum_{n=0}^7 x_n,
                \qquad  \prod_1^{10} \omega_k
```

Typset the following (note the spacing between text and math expressions and the spacing in the exponent!):

We like this: $\left(\dfrac{1}{2}\right)^2$ better than this: $\left(\dfrac{1}{2}\right)^2$

In text mode, the special character `~` leaves a blank space.
`this~word` is interpreted as one word
LaTeX avoids a line break in the middle of any one word.
Useful in avoiding start of line with, for example, a number.

Example: Suppose that when you type
`... blah blah in equation (7) we see blah blah.`
out comes: *... blah blah in equation*
(7) we see blah blah.

To avoid this you type
`blah blah in equation~(7) we see blah blah.`

It is good practice to do this as you type, so that you don't
have to go back and edit your document every time this occurs.

Even more math
Trigonometry and such

Bad:

`cos(\theta)=sin(\tfrac{\pi}2-\theta)`

Good:

`\cos(\theta)=\sin(\tfrac{\pi}2-\theta)`

We also have, among others,

`\exp(x)=e^x, \ln x=\log x, \sinh(z), \theta=\arcsin(x)`

By now you should know where to find a more complete list!

```
\frac{df}{dt}=
\lim_{\epsilon \to 0}\frac{f(t+\epsilon)-f(t)}{\epsilon}
```

and

```
\lim_{x \to \infty}\frac{1}{x}=0
```

Exercise: typeset

$$\int_0^\infty f(x)\,dx \equiv \lim_{t\to\infty}\int_0^t f(x)\,dx$$

You can write

`x^{\frac12}=\sqrt{x}`

and

`x^{\frac13}=\sqrt[3]{x}`

Note how it automatically resizes:

`\sqrt{\frac{y}{x}}=\sqrt{a+b+c+d}`

Exercise: typeset

$$\sqrt[n]{x^{1/n}} = \left(\sqrt[n]{x}\right)^{\frac{1}{n}} = x^{1/n^2}$$

Newton's second law is actually a vector equation:

`\vec F = m \vec a`

Operations between vectors

`\vec a\cdot \vec b, \quad \vec a\times\vec b,`

You can combine things freely, as in

`\vec a\cdot (\vec b+\vec c_1), \quad \vec a\times\vec b`

Exercise: typeset

$$|\vec a + \vec b|^2 = \vec a \cdot \vec a + 2\vec a \cdot \vec b + \vec b \cdot \vec b$$

Even more math

For a list you need *lower* dots $x_1, \ldots, x_n$

```
x_1,\ldots,x_n
```

but for a product or a sum you need plain dots
$$x_1 + \cdots + x_n, \qquad y_1 y_2 \cdots y_n$$

```
x_1+\dots+x_n,\qquad y_1y_2\cdots y_n
```

In text dots (called ellipsis) are \dots or \ldots. You should not use ... Compare (in text mode):

```
from here \ldots to there ... and there
```

By the way, there is no space between \ldots and to in this example. To add space you can use

```
from here \ldots\ to there ... and there
```

or

```
from here \ldots~to there ... and there
```

Backup slides

A bit more information about stuff

The following instructions from Chapter 1 of the course notes (that have live links):

- Go to `http://tug.org/mactex/`
- Scroll down to "To obtain the distribution, click the link below" and click on `MacTeX.pkg`
- Optional: Click the `Donate` link on the title bar. Giving them $10.00 helps a lot and it is certainly much cheaper than buying a textbook or comercial software.
- Editor: Texworks. Get it from `http://www.tug.org/texworks/#Getting_TeXworks`

Necessary Pain: Installation
More painful: Windows

The following instructions from Chapter 1 of the course notes (that have live links):

- Go to `http://miktex.org/`
- Click on *Download* on the title bar and in that page find the *Recommended Download*. This does not download MikTeX but rather an installer.
- Run the installer (if needed, read instructions under *read the tutorial* close to the top.
- When Wizard asks where to install MiKTeX make sure you know where (the default should be `C:\Program Files (x86)\MiKTeX 2.9`.
- Wizard will ask: "Preferred paper" (default `A4`, change to *letter*) and "Install missing packages on the fly" (default `Ask me first` is OK).
- Optional: Click the `Donate` link on the title bar. Giving them $10.00 helps a lot and it is certainly much cheaper than buying a textbook or comercial software.
- Editor: Texworks. Get it from `http://www.tug.org/texworks/#Getting_TeXworks`

This is the second slide

A bit more information about this

This is the second slide

A bit more information about this

This is the second slide

A bit more information about this

This is the second slide

A bit more information about this

This is the second slide

A bit more information about this

This is the second slide

A bit more information about this