# CS199
# Git, Latex, and the Cloud

Bhuvan Venkatesh

# So what's Latex?

- Latex is a markup language. You specify a bunch of elements and they get rendered according to settings. (Think of it like how HTML describes elements and CSS is how they are rendered on the web page).
- Pretty simple to make simple documents but gets hard whenever you get a markup error, but we will show you tools for that!

# Give me the hello world of latex.

```
\documentclass{article}
\title{Hello World Document Title}
\author{Bhuvan Venkatesh}
\date{\today}
\begin{document}
   \maketitle
   Hello world!
\end{document}
```

# Line By Line - Document Class

```
\documentclass{article}
```

- Defines the document to be article, you can style documents based on the type of the document

# Line By Line - Metadata

```
\title{Hello World Document Title}
\author{Bhuvan Venkatesh}
\date{\today}
```

- Defines the document to be article, you can style documents based on the type of the document

# Line By Line - Document

```
\begin{document}
   \maketitle
   Hello world!
\end{document}
```

- This is the body of the document!

# Other Common Commands

- `\usepackage{document}` - Use a package
- `\begin{itemize} \item \end{itemize}` - Make a list
- `\begin{listings} \item \end{listings}` - Code Snippets
- `\textbf{text},\textit{text}` - Bolding/Italicizing
- Pick them up as you go!

# Advanced Stuff - Math Mode

```
\[
    A &= 2

    b &= A+2
\]
```

- We really won't be using much of this, but know that there are different modes
- There are plenty of other ways to make your latex fancy, and all of that is just a stackoverflow/google away!

# Technical Reports - Templates

- We won't expect you to do *all* of the styling. In fact, most of our staff has done the styling for you. All you will need to do is put in the content and see if it renders nicely
- We will give you a sample technical report that of which you can base off.
- https://www.sharelatex.com/
- Sharelatex will give you better error reporting with latex

# So what's Git?

- Git is a version control system. It is much like SVN.
- What is a version control system? I'm glad you asked.
- Version Control Systems keep track of different versions of a folder and all of its subfolders through an idea of a commit.
- Git starts history with an empty repository (folder). You add a file or another folder to the repository and then you can track the changes.
- You can make any edits, add the changes, and then re-roll back down

# Tutorial

- Demo!

# Git is distributed

- Every git copy is as valid as every other copy of git!
- We have something called a remote that we usually attach to github, bitbucket, or some git hosting service.
- That remote is usually the authority that we push to that view on github
- But you can do things like `git push origin master -f`
- That will **overwrite the entire repository with your version of the repository**, basically only do it if you are absolutely sure that you should be using it.

# Why does this matter to me?

- Two Words: Merge Conflict
- Demo! (Pretty much the only way to do this lol)