

## Agenda for Presentation

- Start with the demo, and why vim:
  - Just in the terminal, makes it easy and efficient to edit code without using a mouse (though you can use a mouse with vim)
  - Faster to type, refactor, see errors, change parts of code
  - Large community, lots of plugins, lots of helpful stack-overflow-like posts you can find with a quick google search.
  - Not required.
  - see also: emacs. We won't talk about this, but emacs is another cool editor with a steep learning curve and lots of community and plugins.
  - Demo itself:
    - Syntastic, with some OCaml or C code.
      - \* This one is especially useful, and we can show you how to download and install it at the end.
    - splitting
    - conquer-term
    - autocompile (tab, control-n, control-p)
    - nerd-tree
- Insert Mode
  - How to use i, etc.
  - Introduce the general concept of “mode,” or at least the word
- Normal Mode:
  - Essential commands
    - Saving
    - Quitting
- This is enough to do most of the basic editing operations most are accustomed to with other editors.
  - Include some basic source code examples
  - Moreover, it can run normally in the terminal!
  - But, it is no more powerful than other text editors like sublime text that someone might be using.
- Copying and Pasting: scaling up the commands, starting to talk about composition.
  - y, d and p. Copying, cutting, and pasting.
    - Example here is two large blocks of text, where you have to cut and paste one below the other.
  - Visual Mode
    - It turns out, most traditional text editors come prepackaged with a visual mode. For example, if you open up sublime, here is how you enter “visual mode:” (holds down mouse button, highlights some text).
    - Use this to highlight the entire paragraph, cut it, and paste it below.
  - Now, introduce the idea of composing operations with numbers and other operations, by changing `vjjjjjjj1111111111djjjjjp` to `d7j5jp`
- Cool things you can do from normal mode:

- **w** and **e** for navigating. Show how you can compose these with numbers, and **d**.
- This is how most of these modifiers in normal mode work. There are so many things that we aren't showing you, but they follow more or less the same principles.
- Search with **/**. The file can have the words "delete me" on line 1000 and they can search for it.
- Practice the same basic exercises, but this time using the new commands we've introduced.
- Setting up the `.vimrc`.
  - defaults in their home directory should be fine
- Pathogen:
  - install syntastic: 2.2.1–2.2.2 in <https://github.com/scrooloose/syntastic>
  - more practice
- Conclusion:
  - Composition/modes really are a great foundation for thinking about text editing.
  - The best way to improve is to concentrate on one or two features that you want to learn at a time and use those a lot until they're second nature.
  - Provide a list of plugins folks might enjoy.
  - Things you may be interested in:
    - Tabs
    - Marks
    - Registers
    - Folds
    - Different movement commands
    - Reading from shell
    - And So much more!