# CS150 - Working with Files

Note Title 3/6/2012

## Announcements

- HW5 is due Thursday
- Next HW up tomorrow, + due Wed.
after break

# File I/O

In Python, we can work with 'outside' files.

For a script (in same directory as a file called myfile.txt), need to create a local variable to interact with the file.

"real" file name

**Ex:** varname = file('myfile.txt')

Python won't ever use actual file name except for here - the rest is only through the variable.

Notes:

- x = file(' directions.txt') gives read-only acess to that file
- if no such file, raises IOError
- optional second parameter if you want to write:

write

outputfile = file('output.txt', 'w')

or

outputfile = file ('output.txt', 'a')

append

# Available Functions

- close() - closes the file & saves it
- flush() - save
- read() - reads entire file
- read(size) size in bytes
- readline() - reads up to next /n
- write(s) - writes a string s
- writelines(seq) - writes all strings in seq

list or tuple

no newlines added

## Example: wc

- wc is a unix tool that allow you to count the number of words in a file

(Think same utility in word)

Ex: wc file

output: #lines #words #chars

We'll code our own version of word count to demonstrate input from a file.

A few options:

How should we read the file?

read() , readline()

How to break it into lines?

How to count chars and words?

3rd way:
```
for line in file:
    len(line)
    split to get words
```

## Practice: 8.6 in textbook

Create a file people.txt which has the following format: person bithyear

Ex:

```
St. Edward the Confessor 1003
Matilda 1102
Edgar the Peaceable 942
Ethelred the Unready 968
```

Write a program that reads this file and reports the name of the **oldest** person.

For example, on above, program should print:

```
Edgar the Peaceable
```