Programming Club Meeting
12 Slides

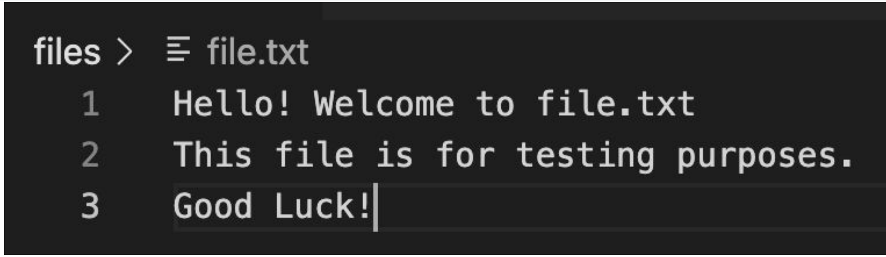
Files

Why Write Files

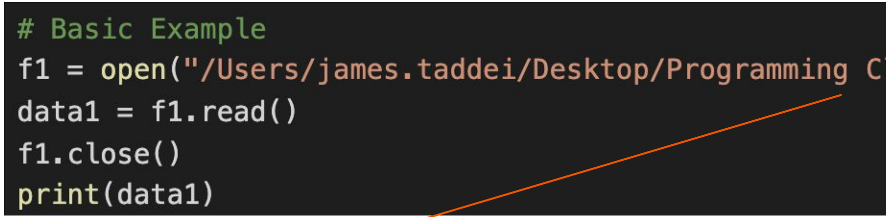
- Store data long term
    - Data in variables is stored in RAM which flushes when the program closes or the computer shuts down
- Hard drive can store much more data than RAM
- Files can be used on other computers or programs, data held in variables cannot

Opening a File

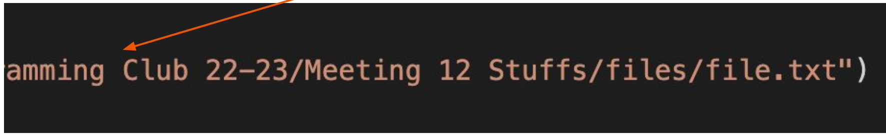
- Will need VS Code to use other files
- Open command (routes to random folder)
- Modes:
    - "r" - read, default, error if file doesn't exist
    - "a" - append, creates if file doesn't exist
    - "w" - write, creates if file doesn't exist, overwrites data
    - "x" - create, error if file exists
    - "+" - include with "r" or "w" for read and write permissions
        - "w+" and "r+" similar except "r+" doesn't delete contents and won't create a new one
- Should always close a file when you're done with it
- Can also delete files, not that important



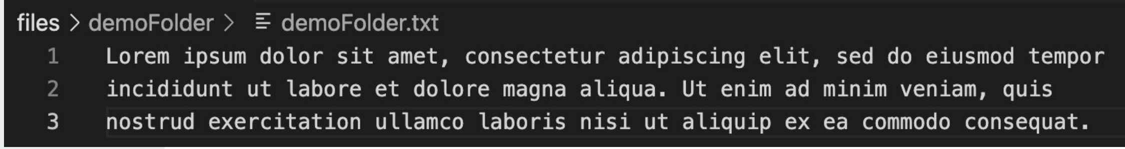
```
Hello! Welcome to file.txt
This file is for testing purposes.
Good Luck!
```



```
# Basic Example
f1 = open("/Users/james.taddei/Desktop/Programming C
data1 = f1.read()
f1.close()
print(data1)
```



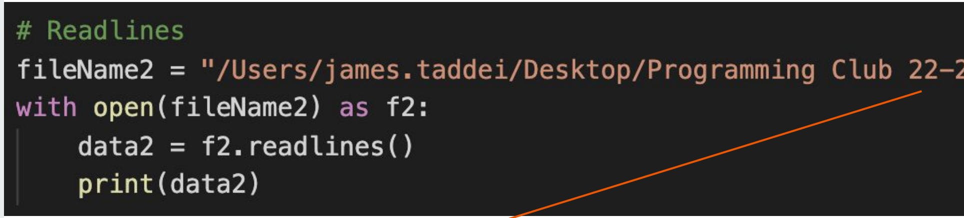
```
amming Club 22-23/Meeting 12 Stuffs/files/file.txt")
```



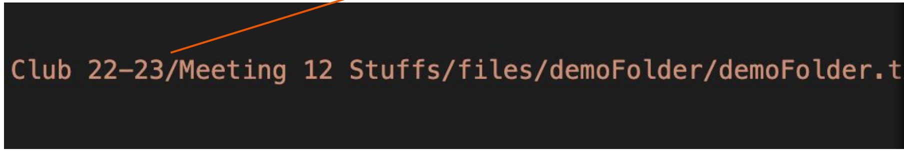
```
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis
nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
```

Better Method

- Don’t need to remember to close
  - Especially easy to forget as the close may need to be many lines later
- Better way to manage things like files
- Creates a sub-area that clearly shows where the file is open

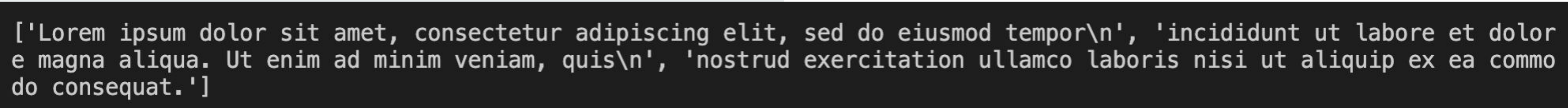


```python
# Readlines
fileName2 = "/Users/james.taddei/Desktop/Programming Club 22-2
with open(fileName2) as f2:
    data2 = f2.readlines()
    print(data2)
```



```
Club 22-23/Meeting 12 Stuffs/files/demoFolder/demoFolder.t
```

Output:



```
['Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor\n', 'incididunt ut labore et dolor
e magna aliqua. Ut enim ad minim veniam, quis\n', 'nostrud exercitation ullamco laboris nisi ut aliquip ex ea commo
do consequat.']
```

Files Methods

- Read
- Readline
  - If you want to do this continuously over a file, you can use a for loop just like a list
- Readlines
- Seek
- Write

Practice

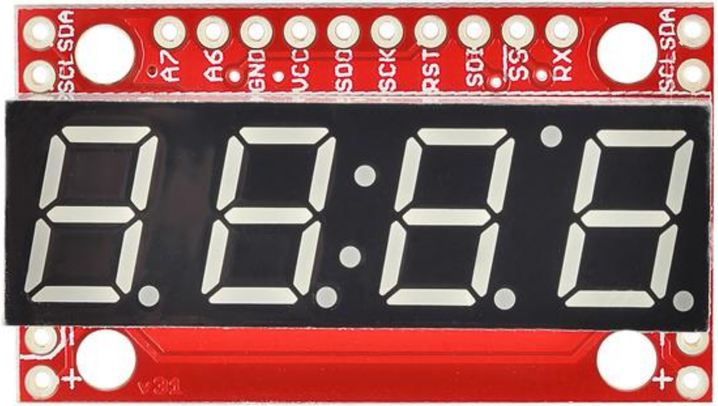
Practice Problem 1:

Store Random Values

- Write a Python program that will store 10 of each of the 3 random values from the Random Values problem we did 2 weeks ago.
- Relevant Information:
  - To generate the random values, you can look at the code in the “randomValues” folder on the GitHub
  - The file should be 10 random RGB values followed by the random strings and then the multiples of 7
  - When writing to files, the value must be a string
  - Each value should be on a newline

Practice Problem 2: Longest 7-Segment Word

- Goal: Write a Python program that will determine the longest word that a seven-segment screen can display.
- Relevant Information:
  - Assume that a 7-segment display can't show G, K, M, Q, V, W, X, and Z but can make all other letters
  - You will need a file containing every English word, one is in the GitHub folder named "longest7SegWord"



Practice Problem 3:

Word Frequency

- Src: https://www.w3resource.com/python-exercises/file/python-io-exercise-10.php
- Goal: Write a Python program that will determine the number of times that each word in the file occurs.
- Relevant Information:
  - You can assume that there are only words, spaces, and newlines in the file
  - Words should be counted as the same even if the capitalization is different

Next Meeting: Review