Chapter 11

Dr. Raza Ul Mustafa Khokhar
American University
Computer Science Department - CSC-148

Working with Data Files

- In Python, we must **open** files before we can use them and **close** them when we are done with them. As you might expect, once a file is opened it becomes a Python object just like all other data

Method Name	Use	Explanation
`open`	`open(filename,'r')`	Open a file called filename and use it for reading. This will return a reference to a file object.
`open`	`open(filename,'w')`	Open a file called filename and use it for writing. This will also return a reference to a file object.
`close`	`filevariable.close()`	File use is complete.

Finding file in your disk

- The way that files are located on disk is by their **path** You can think of the filename as the short name for a file, and the path as the full name.
- For example on a Mac if you save the file hello.txt in your home directory the path to that file is /Users/yourname/hello.txt
- On a Windows machine the path looks a bit different but the same principles are in use. For example on windows the path might be C:\Users\yourname\My Documents\hello.txt

Absolute path

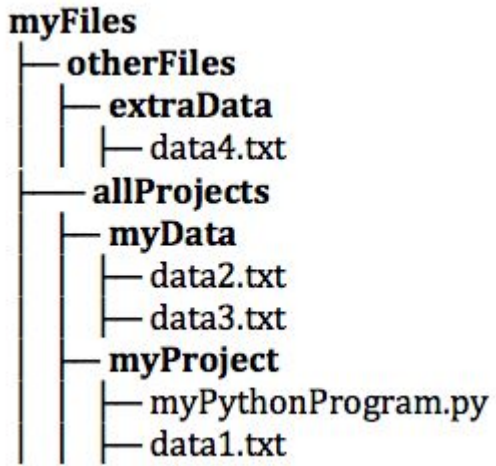
if you had a file called hello.py in a folder called CS150 that is inside a folder called PyCharmProjects under your home directory, then the full name for the file hello.py is /Users/yourname/PyCharmProjects/CS150/hello.py. This is called an *absolute file path*.

A *relative file path* starts from the folder that contains your python program and follows a computer's file hierarchy. A file hierarchy contains folders which contains files and other sub-folders.

Example of relative paths

Using the example file hierarchy above, the program, myPythonProgram.py could access each of the data files using the following *relative file paths*:

- data1.txt
- ../myData/data2.txt
- ../myData/data3.txt
- ../../otherFiles/extraData/data4.txt



Reading a file

- Suppose we have a text file called ccdata.txt that contains the following data representing statistics about climate change

To open this file, we would call the open function. The variable, fileref, now holds a reference to the file object returned by open

```
fileref = open("ccdata.txt", "r")
fileref.close()
```

Text file content, download from online material

Data file: *ccdata1.txt*

1850	-0.37	2.24E-7
1860	-0.34	3.94E-7
1870	-0.28	6.6E-7
1880	-0.24	1.1
1890	-0.42	1.72
1900	-0.2	2.38
1910	-0.49	3.34
1920	-0.25	4.01
1930	-0.14	4.53
1940	0.01	5.5
1950	-0.17	6.63
1960	-0.05	10.5
1970	-0.03	16
1980	0.09	20.3
1990	0.3	22.6
2000	0.29	24.9
2010	0.56	32.7
2019	0.74	33.3

Read text file line by line

```
ccfile = open("ccdata.txt", "r")

for aline in ccfile:
    values = aline.split()
    print('In', values[0], 'the average temp. was', values[1], '°C and CO2
emmisions were', values[2], 'gigatons.')

ccfile.close()
```

Readfile using while loop?

```
infile = open("ccdata.txt", "r")
line = infile.readline()
while line:
    values = line.split()
    print('In', values[0], 'the average temp. was', values[1], '°C and CO2 emmisions
were', values[2], 'gigatons.')
    line = infile.readline()

infile.close()
```

Few other methods - What do you think they will do?

```
infile = open("ccdata.txt", "r")
aline = infile.readline()
aline

infile = open("ccdata.txt", "r")
linelist = infile.readlines()
print(len(linelist))

print(linelist[0:4])

infile = open("ccdata.txt", "r")
filestring = infile.read()
print(len(filestring))
```

Explanation of few methods

In addition to the for loop, Python provides three methods to read data from the input file.

- The readline method reads one line from the file and returns it as a string. The string returned by readline will contain the newline character at the end. This method returns the empty string when it reaches the end of the file.
- The readlines method returns the contents of the entire file as a list of strings, where each item in the list represents one line of the file.
- It is also possible to read the entire file into a single string with read.

Writing text file

- The only difference between opening a file for writing and opening a file for reading is the use of the 'w' flag instead of the 'r' flag as the second parameter.

```
infile = open("ccdata.txt", "r")
outfile = open("emissiondata.txt", "w")

aline = infile.readline()
outfile.write("Year \tEmmision\n")
while aline:
    items = aline.split()
    dataline = items[0] + '\t' + items[2]
    outfile.write(dataline + '\n')
    aline = infile.readline()

infile.close()
outfile.close()
```

Only taking two columns of the original txt file and then writing it to new file.

Slides & Material

razaulmustafa.us/cs148/