New York University
School of Continuing and Professional Studies
Division of Programs in Information Technology

Introduction to Python

Homework Discussion, Session 4

4.1 Calculate the average Mkt-RF value for a given year using a compiled list.

The purpose of the assignment is to compile a list of values from a data source, then use that list to learn things about the data collected, like len, sum and average (sum / len). The benefit of using a list rather than a sum total and a count (last week's approach) is that we can perform other sorts of calculations as well -- median (middle value), for example. Building data into containers is one of the primary things we do as programmers. We cull data from a data source and use the container suited to our analysis (a set for unique values or membership checks, a dictionary for lookups, etc.)

We want to think of our program in terms of discrete steps. Here are the steps I can see for this solution, followed by an outline:

1. Ask the user for a 4-digit year and check to see that it is 4 digits only.
2. Open the file and read the data into a list, slicing the list to obtain the data lines.
3. Initialize an empty list. Loop through the file data and slice out the year. If the year is the user's chosen year, split out the Mkt-RF value and add to the list.
4. Check to see if the list has any elements (it won't if the year entered doesn't appear in the data). If there are no elements, exit with an error message.
5. Use len() and sum() to calculate the average.
6. Report the results.

Pseudocode:

```
use a while True loop
    take user input for a 4-digit year
    if the year is not all digits and len 4,
        print error message
        'continue' to loop back to the start of the loop
    'break' out of the loop (because the input is correct)


open file
use readlines() to produce a list of lines
close the filehandle
use a slice to get the data lines (without the header and footer)


initialize an empty list
looping through the file lines,
    slice out the year
    if the user's year equals the sliced out year,
        split the line into columns
        convert the Mkt-RF element to a float
        append the float to the list


if there is no data in the list (if not yearlist:),
    exit() with an error message


calculate the sum of values in the list with sum()
calculate the len of values in the list with len()
calculate the average of values using the above two


report the len and average
```

4.2 Spell checker.

The purpose of the assignment is to acquaint you with set membership tests. This program is a practical solution implementing functionality familiar to all of us; it can be used on any file. Testing membership is another core analysis technique.

Here are the steps I took in this solution followed by a pseudocode outline:

1. set 2 string variables specifying the name and location of the sawyer.txt and words.txt files.
2. Initialize an empty set. Open the dictionary words file, loop through each line (there is one word per line), strip the word of whitespace and punctuation, and lowercase the word. Add the word to the set. Close the filehandle.
3. Open the test words file (sawyer.txt), read the file into a string, and split the string into a list of individual words using str.split(). Looping through the list of words, strip the word of whitespace and punctuation, and lowercase the word. Then test each word against the spelling words set using the in operator. If a word isn't found in the set, report the missing word.

Pseudocode:

```
initialize an empty set
open the spelling words file
loop through the file line by line
    rstrip the word of whitespace
    rstrip the word of punctuation (';,:.')
    lowercase the word
    add the prepared word to the set
close the file


open the test file (sawyer.txt)
read the file into a string with read()
split the file into a list of words (on whitespace) with split()
close the file
loop through the list of words
    rstrip the word of whitespace
    rstrip the word of punctuation (';,:.')
    lowercase the word
    if the prepared word is not found in the set compiled earlier (use in test):
        report the misspelled word
```