10.3. Alternative File Reading Methods

Once you have a file "object", the thing returned by the open function, Python provides three methods to read data from that object. The `read()` method returns the entire contents of the file as a single string (or just some characters if you provide a number as an input parameter. The `readlines` method returns the entire contents of the entire file as a list of strings, where each item in the list is one line of the file. The `readline` method reads one line from the file and returns it as a string. The strings returned by `readlines` or `readline` will contain the newline character at the end. Table 2 summarizes these methods and the following session shows them in action.

Method Name	Use	Explanation
`write`	`filevar.write(astring)`	Add a string to the end of the file. `filevar` must refer to a file that has been opened for writing.
`read(n)`	`filevar.read()`	Read and return a string of `n` characters, or the entire file as a single string if `n` is not provided.
`readline(n)`	`filevar.readline()`	Read and return the next line of the file with all text up to and including the newline character. If `n` is provided as a parameter, then only `n` characters will be returned if the line is longer than `n`. **Note** the parameter `n` is not supported in the browser version of Python, and in fact is rarely used in practice, you can safely ignore it.
`readlines(n)`	`filevar.readlines()`	Returns a list of strings, each representing a single line of the file. If `n` is not provided then all lines of the file are returned. If `n` is provided then `n` characters are read but `n` is rounded up so that an entire line is returned. **Note** Like `readline` `readlines` ignores the parameter `n` in the browser.

In this course, we will generally either iterate through the lines returned by `readlines()` with a for loop, or use `read()` to get all of the contents as a single string.

In other programming languages, where they don't have the convenient for loop method of going through the lines of the file one by one, they use a different pattern which requires a different kind of loop, the `while` loop. Fortunately, you don't need to learn this other pattern, and we will put off consideration of `while` loops until later in this course. We don't need them for handling data from files.

Note

A common error that novice programmers make is not realizing that all these ways of reading the file contents, **use up the file**. After you call readlines(), if you call it again you'll get an empty list.

Check your Understanding

```
Writing essays for school can be difficult but
many students find that by researching their topic that they
have more to say and are better informed. Here are the university
we require many undergraduate students to take a first year writing requirement
so that they can
have a solid foundation for their writing skills. This comes
in handy for many students.
Different schools have different requirements, but everyone uses
writing at some point in their academic career, be it essays, research papers,
technical write ups, or scripts.
```

1. Using the file `school_prompt2.txt`, find the number of characters in the file and assign that value to the variable `num_char`.

Save & Run 5/14/2021, 5:37:44 AM - 2 of 2

```
1 file=open("school_prompt2.txt","r")
2 st=file.read()
3 num_char = 0
4 for ch in st:
5     num_char+=len(ch)
6 file.close()
7
```

Activity: 1 -- ActiveCode (ac9_4_1)

Result	Actual Value	Expected Value	Notes
Pass	537	537	Testing that num_char has the correct value.

You passed: 100.0% of the tests

```
This summer I will be travelling.
I will go to...
Italy: Rome
Greece: Athens
England: London, Manchester
```

```
France: Paris, Nice, Lyon
Spain: Madrid, Barcelona, Granada
Austria: Vienna
I will probably not even want to come back!
However, I wonder how I will get by with all the different languages.
I only know English!
```

2. Find the number of lines in the file, `travel_plans2.txt`, and assign it to the variable `num_lines`.

Save & Run 5/14/2021, 5:38:10 AM - 2 of 2

```
file=open("travel_plans2.txt","r")
st=file.read()
print(st)
lst=st.split("\n")
num_lines=len(lst)-1

file.close()

```

```
This summer I will be travelling.
I will go to...
Italy: Rome
Greece: Athens
England: London, Manchester
France: Paris, Nice, Lyon
Spain: Madrid, Barcelona, Granada
Austria: Vienna
I will probably not even want to come back!
However, I wonder how I will get by with all the different languages.
I only know English!
```

Activity: 2 -- ActiveCode (ac9_4_2)

Result	Actual Value	Expected Value	Notes
Pass	11	11	Testing that num_lines is assigned to correct value.

You passed: 100.0% of the tests

```
Sad upset blue down melancholy somber bitter troubled
Angry mad enraged irate irritable wrathful outraged infuriated
Happy cheerful content elated joyous delighted lively glad
Confused disoriented puzzled perplexed dazed befuddled
Excited eager thrilled delighted
Scared afraid fearful panicked terrified petrified startled
Nervous anxious jittery jumpy tense uneasy apprehensive
```

3. Create a string called `first_forty` that is comprised of the first 40 characters of `emotion_words2.txt`.

Save & Run 5/14/2021, 5:38:31 AM - 2 of 2

```
file=open("emotion_words2.txt","r")
first_forty=file.read(40)
file.close()

```

Activity: 3 -- ActiveCode (ac9_4_3)

Result	Actual Value	Expected Value	Notes
Pass	'Sad u...er bi'	'Sad u...er bi'	Testing that first_forty was created correctly.

Expand Differences

You passed: 100.0% of the tests

Data file: `travel_plans2.txt`
Data file: `school_prompt2.txt`
Data file: `emotion_words2.txt`

You have attempted 4 of 3 activities on this page

✔ Completed. Well Done!

10.4. Iterating over lines in a file">Next Section - 10.4. Iterating over lines in a file

| Back to top