Exercise 15: Reading Files

MAIN
PLAY VIDEO
PREVIOUS
NEXT
HELP
Follow

You know how to get input from a user with `raw_input` or `argv`. Now you will learn about reading from a file. You may have to play with this exercise the most to understand what's going on, so do the exercise carefully and remember your checks. Working with files is an easy way to *erase your work* if you are not careful.

This exercise involves writing two files. One is the usual `ex15.py` file that you will run, but the *other* is named `ex15_sample.txt`. This second file isn't a script but a plain text file we'll be reading in our script. Here are the contents of that file:

```
This is stuff I typed into a file.
It is really cool stuff.
Lots and lots of fun to have in here.
```

What we want to do is "open" that file in our script and print it out. However, we do not want to just "hard code" the name `ex15_sample.txt` into our script. "Hard coding" means putting some bit of information that should come from the user as a string directly in our source code. That's bad because we want it to load other files later. The solution is to use `argv` or `raw_input` to ask the user what file to open instead of "hard coding" the file's name.

```
 1  from sys import argv
 2
 3  script, filename = argv
 4
 5  txt = open(filename)
 6
 7  print "Here's your file %r:" % filename
 8  print txt.read()
 9
10  print "Type the filename again:"
11  file_again = raw_input("> ")
12
13  txt_again = open(file_again)
14
15  print txt_again.read()
```

A few fancy things are going on in this file, so let's break it down real quick:

Lines 1-3 uses `argv` to get a filename. Next we have line 5 where we use a new command `open`. Right now, run `pydoc open` and read the instructions. Notice how like your own scripts and `raw_input`, it takes a parameter and returns a value you can set to your own variable. You just opened a file.

Line 7 prints a little message, but on line 8 we have something very new and exciting. We call a function on `txt` named `read`. What you get back from `open` is a `file`, and it also has commands you can give it. You give a file a command by using the `.` (dot or period), the name of the command, and parameters. Just like with `open` and `raw_input`. The difference is that `txt.read()` says, "Hey `txt`! Do your read command with no parameters!"

The remainder of the file is more of the same, but we'll leave the analysis to you in the Study Drills.

What You Should See

I made a file called `ex15_sample.txt` and ran my script.

```
$ python ex15.py ex15_sample.txt
Here's your file 'ex15_sample.txt':
This is stuff I typed into a file.
It is really cool stuff.
Lots and lots of fun to have in here.


Type the filename again:
>  ex15_sample.txt
This is stuff I typed into a file.
It is really cool stuff.
Lots and lots of fun to have in here.
```

Study Drills

This is a big jump so be sure you do this Study Drill as best you can before moving on.

1. Above each line, write out in English what that line does.
2. If you are not sure ask someone for help or search online. Many times searching for "python THING" will find answers to what that THING does in Python. Try searching for "python open."
3. I used the word "commands" here, but commands are also called "functions" and "methods." You will learn about functions and methods later in the book.
4. Get rid of the lines 10-15 where you use `raw_input` and run the script again.
5. Use only `raw_input` and try the script that way. Why is one way of getting the filename better than another?
6. Start `python` to start the `python` shell, and use `open` from the prompt just like in this program. Notice how you can open files and run `read` on them from within `python`?
7. Have your script also call `close()` on the `txt` and `txt_again` variables. It's important to close files when you are done with them.

Common Student Questions

Q: Does `txt = open(filename)` return the contents of the file?

> No, it doesn't. It actually makes something called a "file object." You can think of a file like an old tape drive that you saw on mainframe computers in the 1950s, or even like a DVD player from today. You can move around inside them, and then "read" them, but the DVD player is not the DVD the same way the file object is not the file's contents.

Q: I can't type code into my Terminal/PowerShell like you say in Study Drill 7.

> First thing, from the command line just type `python` and press Enter. Now you are in `python` as we've done a few other times. Then you can type in code and Python will run it in little pieces.

> Play with that. To get out of it type `quit()` and hit Enter.

Q: Why is there no error when we open the file twice?

> Python will not restrict you from opening a file more than once and sometimes this is necessary.

Q: What does `from sys import argv` mean?

> For now just understand that `sys` is a package, and this phrase just says to get the `argv` feature from that package. You'll learn more about these later.

Q: I put the name of the file in as `script, ex15_sample.txt = argv` but it doesn't work.

> No, that's not how you do it. Make the code exactly like mine, then run it from the command line the exact same way I do. You don't put the names of files in, you let Python put the name in.

Video

Purchase The Videos For $29.59

For just $29.59 you can get access to all the videos for Learn Python The Hard Way, **plus** a PDF of the book and no more popups all in this one location. For $29.59 you get:

- All 52 videos, 1 per exercise, almost 2G of video.
- A PDF of the book.
- Email help from the author.
- See a list of everything you get before you buy.

When you buy the videos they will immediately show up **right here** without any hassles.

Already Paid? Reactivate Your Purchase Right Now!

Buying Is Easy

Buying is easy. Just fill out the form below and we'll get started.

Full Name

Full Name

Email Address

Email Address

● Pay With Credit Card (by Stripe™)

○ Use your PayPal™ account.

Buy Learn Python The Hard Way, 3rd Edition

Zed Shaw PDF + Videos + Updates $29.95	**Amazon** Paper + DVD $22.74	**Amazon** Kindle $17.27	**B&N** Paper + DVD $22.96	**B&N** Nook (No Video) $17.27
	InformIT eBook + Paper $43.19	**Interested In Ruby?** Ruby is also a great language. Learn Ruby The Hard Way		