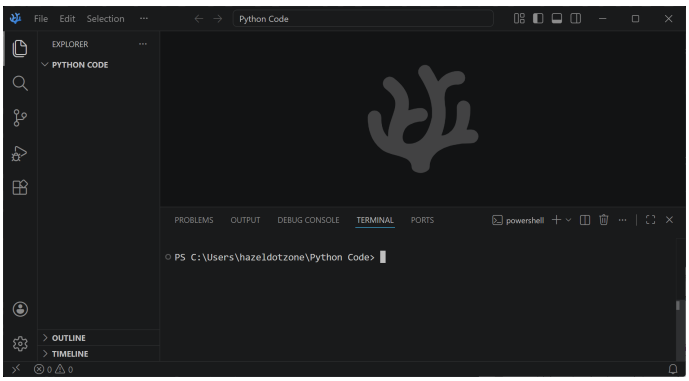
Exercises for Deck 002

Python — Source Files

Dr. Hazel “twitch.tv/hazeldotzone” Campbell

1) In the screenshot, label the explorer pane, the terminal, and the code editor pane.



2) In the screenshot, there is light yellow, dark yellow, blue, and white text. Which color is...

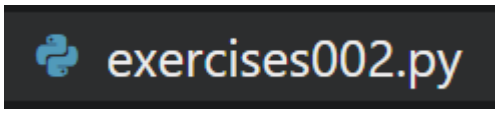
   a) The identifiers?

   b) The delimiters?

   c) The literals?

   d) The operator?

```
1    print("hello"+"world")
```



3) In the screenshot to the left, what does the blue icon represent?

4) In your editor, create a new file called “exercise002.py”.

   a) Verify that your editor recognizes the new file as a python source code file. How did you do that?

   b) Locate the editor in your operating system’s file browser/explorer/finder. What is the path to the file?

5) Make sure your “exercise002.py” is empty and saved. Then open a terminal in your editor. Make sure you’re not on the problems/output/debug/ports tab.

   a) Verify what command-line shell you are giving input to by recognizing its prompt string. Copy the prompt string from the terminal you just opened:

   b) Based on the prompt string, what is waiting for input?

6) Using the terminal, run your “exercise002.py” ...
    a) What command did you use to run it?
    b) Locate the output in the terminal window.
    c) How many lines of output did your python program “exercise002.py” write?
7) Now we will add some lines of code to exercise002.py.
    a) Add a line that prints “fear is the mindkiller”
    b) Add a second line that prints “fear is the little-death”
    c) Save and run “exercise002.py”
    d) Locate the written output of the first line.
    e) Locate the written output of the second line.
    f) Where in your terminal is the written output? What was the line in the terminal before “fear is the mindkiller”?
        1) Why does that line appear in the terminal before your first print call’s written output?
    g) Where in your terminal is the written output? What was the line in the terminal after “fear is the little-death”?
        1) Why does that line show up after your second print call’s written output?
8) Add a third line to exercise002.py that calls `pirnt()` instead of print. Pay attention to the misspelling.
    a) Double check using the editor that the misspelled `pirnt()` call is in fact on line 3. Where did you check that?
    b) Save and run the exercise002.py that calls the misspelled `pirnt` function.
    c) Verify that you got an error.
        1) Locate the error. What is the error?
        2) Locate the location of the fault in the traceback. What line does the traceback indicate that the fault is on?
    d) Does the lines from the litany against fear still appear?
9) Repair your program by correcting the spelling of `pirnt` to `print()`.
    a) Save and run your program.
    b) Inspect the terminal for any errors and verify the fault is repaired.
    c) How many lines of output does your program write in total?

10) Add a fourth line to your program that prints the sum of 33 and 55 using the addition operator and integer literals.
    a) Save and run your program.
    b) Verify that your program still outputs the two lines from the litany against fear, a blank line, and then the sum (88).
11) Add a fifth line to your program that prints the concatenation of the strings “33” and “55” using the addition operator.
    a) Save and run your program.
    b) Verify that your program still outputs everything it did before and the concatenation 3355.
12) Draw a syntax diagram of the fourth line of your program, and the fifth line of your program. Put the two diagrams next to each other like this:

Fourth line:	Fifth line:

a) Label the function calls, operators, and the types of literals in your diagrams.

b) What is different between the two diagrams?

c) What is the same in the two diagrams?

13) In the sixth line of your program, write the sum of two integer literals 33 and 55.

a) Save and run your program.

b) Did adding the sixth line change the output?

1) Why or why not?

c) Would the sixth line have shown something in the Python REPL?

1) Why or why not?

14) In the seventh line of your program, attempt to print the sum of the integer literal 33 and the string literal “55”.

a) Save and run it.

b) Verify that you receive an error.

c) Locate the error in the terminal. What is the error?

d) Locate the fault location in the terminal. On what line is the location?

15) Without fixing the fault in the seventh line of your program, add an eighth line to print the string literal “I will face my fear.”

a) Save and run your program.

b) Verify you still receive an error as you did in question 14.

c) Does your program write the output “I will face my fear.” to the terminal? (Hint: it should not.)

d) Form a hypothesis about why your program did not write the output “I will face my fear.”

1) Write down your hypothesis.

e) Plan a change to your program that would confirm your hypothesis by changing your program’s behaviour.

1) Describe the change you plan to make.

A) What is the code you plan to change?

B) What is the behavior you plan to change?

C) What is the output you expect from the **old** behavior?

D) What is the output you expect from the **new** behavior?

f) Implement the change to your program, save, and run it.

g) Write down the change in output you observed from the new version when compared to the old version.

h) Does your answer to (g) match your expectation in (e)?

i) Have you confirmed your hypothesis?