

Functions

Functions. They make programmers look good. Instead of having a single `main()` function with 1,200 lines of code in it, you can now break up your code into nice, reusable pieces of code.

Your Assignment

In this assignment you will revisit reading data from a file, and use that data as arguments (parameters) for a number functions you will write.

You will need to:

1. Write and test a function `square_each(nums)`
 - Where `nums` is a (Python) list of numbers. It *modifies* the list `nums` by squaring each entry and replacing its original value. You must modify the parameter, **return will not be allowed!**
2. Write and test a function `sum_list(nums)`
 - Where `nums` is a (Python) list of numbers. It **returns the sum** of the numbers in passed-in the list.
3. Write and test a function `to_numbers(str_list)`
 - Where `str_list` is a list of strings, each of which represents a number. Modify each entry in the list by **converting it to a number**. You can choose to **return the list**, or just **modify `str_list` in-place, similar to #1**, above.
4. Write and test a function `main()`

- Take the previous three functions and construct a `main()` function that will call these functions properly. The program **computes the sum of the squares of numbers read from a file**. Your program should prompt for a file name and print out the sum of the squares of the values in the file. *Hint:* **use `readlines()`**, or **`readline()`** (be aware of how each behaves when returning data from your file).

Your program must read in a file containing a list of numbers. Use the files [data.txt](#) and [data2.txt](#) to test.

Output

See the following examples on `data.txt` and `data2.txt`. Note these are **two** separate examples. The output of each run of your program should only be two lines.

Example 1

```
Please enter the file name: data.txt
The sum of the squares of the numbers in the file is 385
```

Example 2

```
Please enter the file name: data2.txt
The sum of the squares of the numbers in the file is 338350
```

Notes

All printed strings **must** match the above output. To otherwise deviate will cause the Gradescope autograder to deduct points.

- The data files will only include integers, each on a line of their own.
- The only input the program takes is the name of the file.
- Each of the three functions (`square_each`, `sum_list`, and `to_numbers`) **must** be used (and function correctly).
- You must use a main function.

- You should include a docstring with at least your name.

Submission Requirements

The title for this lab is **hw5**. Your submitted file is **required** to be named `hw5.py`.

Grading

The rubric for this assignment is available through Gradescope.