

# Lab 8 – CSE 101 (Spring 2021)

## Objectives

The primary objective of this lab assignment is to get comfortable using dictionaries and file input in Python programs.

### 1. Practice with dictionaries (3 points)

Download [lab8.py](#) and do Part 1, uncommenting and trying out each example in the code one at a time before you move on to the next step.

Then proceed to do Part 2, do the tasks one at a time by writing the code as directed by the instructions in the comments. You do not need to modify or add any print functions, just uncomment the code for part 2 in the `main` function to test it.

For Part 3, you are writing a function to determine your upcoming trip destination. You want to travel to a city that is the farthest distance from Seoul, but you have a max distance you are willing to travel for a one-way trip.

In the code, complete the `destination` function that you will use to calculate your destination. This function takes the following parameters:

1. `max_distance`: an integer that represents how far you are willing to travel, one-way.
2. `places`: a dictionary that maps destinations (as strings) to distances (as integers). For example, if the key 'London' is mapped to the value 8000, this means that the destination London is 8000 kilometers away from where the person is right now.

Your function should scan through the dictionary, then find the farthest destination that is less than or equal to the `max_distance` for a one-way trip.

Note: If `max_distance` is a number that's zero or less, this means the distance is invalid, and your function should return the value `None` (NOT the string 'None'). If you cannot reach a city given the `max_distance`, return 'nowhere'. Do not worry about two destinations having the same distance.

In the code, you are given a dictionary of the distances from Seoul to other cities, as well as test cases with the expected output.

### 2. Practice using files (2 points)

For this problem add the following files to your VS Code directory.

- [prices1.txt](#)
- [prices2.txt](#)
- [prices3.txt](#)

Go to Part 4 in the lab8.py complete the function `shop` that takes the following parameters, in this order:

1. `filename`: an input file name. The file contains many lines of grocery items and their corresponding prices. In each line, the format is: `name of product,price per unit`. For example, a line of the file might look like this: `'Pringles,1.48'`.

**Note:** content like `'Pringles,1.48'` is a string, so you will need to convert `'1.48'` to a floating-point number when calculating the total price. For example, `float('1.48')` would return the floating-point number `1.48`. In practice your code will pass a variable to `float()` as an argument (not a literal string as shown here). Also, the quotation marks in the above example do not actually exist in the file - they are shown just to remind you that the file contains strings.

2. `shopping_list`: a string that represents a list of things that you want to buy during this shopping trip. The format is: `name of product 1,how much product 1 you want,name of product 2,how much product 2 you want, etc`. For example: `'Pringles,3,Butter,1'` means you are buying 3 units of Pringles and 1 unit of Butter.

The function computes and returns the total cost for the items in the `shopping_list` based on the prices given in the input file.

**Hint:** Build a dictionary based on the file and look up prices by product names.

**Note:** If the shopping list string is empty, return `0.0`. If there is an item in the shopping list that does not exist in the file, skip the item and continue processing the rest of the shopping list.

In the code, you are given test cases with the expected output.

## Submission

Submit the completed `lab8.py` program on Blackboard.