EAS CPD Unit 2: Week 3

Exercises – Functions

Functions allow us to define code for later reuse by simply calling the name of the function. This can be useful for breaking down complex programmes into simpler segments, or for having a single line method call for code that is often repeated within a programme. Both can make code easier to read and write.

Example:

```
def say_hello():                    -Define the function name and arguments
     print("Hello, world!")         -Function code

say_hello()                         -Calling the function
```

1 "Simple" Functions

Get used to writing functions with these simple(ish) exercises. You may test each exercise by calling your functions at the bottom of your python files - you do not need to implement user input (though it may make a good challenge!)

Exercise 1.1
Make a simple calculator app that can add, subtract, multiply or divide two numbers.

HINT: You will want to define a different function for each type of calculation.

Exercise 1.2
Extend your calculator app so that it can calculate the mean of a list of numbers.

Exercise 1.3
Write a program with functions for converting temperature between Celsius and Fahrenheit.

Exercise 1.4
Write a function that counts the number of words in a string.

HINT: There are multiple ways to solve this but you may find the .split() method useful

Challenge 1
Add functionality to your calculator app so that if any non-numbers are used it does not attempt a calculation and instead provides a suitable message.

Challenge 2
Write a function that will check if a provided string is a palindrome (reads the same forwards as backwards).

HINT: You will need to compare each character to its equivalent string location in reverse.

For maximum compatibility you may expand your function so that it works regardless of any spaces and upper/lower case.

2 Phonebook Program

Pull together what we have learned about functions, data structures and read/write by creating a single program with a variety of complex functionality.

Exercise 2.1
Create a new folder on your computer. This will store everything for our program so give it a suitable name.

Download the "phonebook.csv" file from the week 3 "DataFiles" folder (https://tc1.me/CSCPD2425) and save it in your new folder.

Create a new python file with the following code:

```
desired_phonebook = "phonebook.csv"
phonebook = {}
```

save it in your new folder with a suitable name.

Exercise 2.2
Add the following functions to your program:

`open_phonebook()` – reads the csv file and fills the "phonebook" dictionary. Some processing will be needed to make the data suitable. Names should be used as the key.

`update_phonebook()` – writes the contents of the "phonebook" dictionary back to the csv file. You will need to make the data suitable.

`add_contact(name, number)`- adds a new contact to the "phonebook" dictionary

`get_contact(name)` – returns the phone number for the provided name. Returns an appropriate message of no such contact exists in the "phonebook" dictionary

`delete_contact(name)` – removes a contact from the "phonebook" dictionary