­­­­­­Teacher Training Notes

Session 5

Slide 1 – Review

* Let’s have a refresh of last week’s lesson.

Slide 2 – Functions

* A function is a block of code which only runs when its called.
* You can pass it data, known as parameters into the function.
* A function can return data as a result.

Slide 3 - Functions — Create

* In Python, a function is defined using the def key word followed by round brackets, that can be left empty or have a parameter passed through it and then a colon.

Slide 4 - Functions — Call

* To use or call a function, you type the name of the function followed by the round brackets.
* So In this example, if you look at the code within the function, it will print “Hello World”, so if you call the function, at the bottom of the example, this will run the code within the function and print “Hello World”.

Slide 5 - Functions — Parameters

* As stated earlier, parameters are data we can pass the function.
* In this example, in the brackets we have defined a variable ‘name’, so anything we pass in the brackets when calling the function will be passed to the name variable within the code.
* So as the example shows, when calling the function and adding the different names, it will print “Hello Alice”, “Hello Bob” and “Hello Charlie” concurrently.

Slide 6- Functions — Parameters

* This example shows that you can pass multiple parameters to your function, this one will take a name and age and put that into the code.
* So when the last one is called, what will it print?
* Hello Charlie. I’m 17 years old. In 10 years time, I will be 27.

Slide 7 - Functions — Parameters

* This function will take three parameters, and multiply them together.
* It then casts the answer to a string so it can be concatenated with a string.
* So what will be printed in these two examples?
* The area is 144
* The area is 840

Slide 8 - Functions — Parameters

* Here is the syntax for including parameters to your function.

Slide 9 - Functions — Returning

* You can have your function return a value, unless stated this will not be printed.
* You can use the return data in your code to input into other variables in your code.

Slide 10 - Functions — Single Job

* Best practice for functions is that each function should have a single job.
* Here, we have split up the previous function into two different ones, one that accepts your name and age, the other purpose is to return a value that is then input back into the first function.
* Single jobs make it easier if you need to update the code, as it will be easier to pinpoint the area that needs updated.

Slide 11 - Functions — Recursion

* Recursion is always a bit of a tricky one but is where the function calls itself.

Slide 12 - Functions — Recursion

* So here the function calc\_factorial is called within itself to times itself with itself minus 1, until the answer is 1 and stops the function.
* So what will it print?
* ‘The factorial of 4 is 24’.

Slide 13 - Questions?

* Does anyone have any questions from last week’s content?

Slide 14 – Files

* File handling is an important part of any web application.
* Python has several functions for creating, reading, updating and deleting files.

Slide 15 - Files — Open

* One of the key function for working with files in Python is the open function.
* The open function takes two parameters, filename and mode.
* This example has a variable named f, with the open function, opening a file called my\_file.txt and the “r” at the end means that it will open read only. |

Slide 16 - Files — Read

* The open function returns a file object, which has a read method for reading the content of the file.
* This example, opens the my\_file.txt document and then reads the entire document.

Slide 17 - Files — Read

* This example again opens the my\_file.txt document but the for loop will allow you to read each line in the document.

Slide 18 - Coding Time - Section A

* Now, it’s your time to try some coding, please refer to Section A on your worksheets.
* You’ll have to about 30 minutes to work on these.

Slide 19 - Files — Handling

* There are four different methods for opening a file
* "r" is Read, this opens a file for reading, error if the file does not exist
* "a" is Append, this opens a file for appending, creates the file if it does not exist
* "w"is Write, this opens a file for writing, creates the file if it does not exist, this will overwrite any information already in the text document.
* "x" is Create, this creates the specified file, returns an error if the file exists

Slide 20 - Files — Write

* To write to an existing file, you must add a parameter to the open function.
* In this example, you are opening an example.txt document in the write mode and then using the write function you are writing “Hello World” to the example.txt and then close the file.
* If you now open the example.txt document it will have “Hello World” printed in it.

Slide 21 - Files — Write

* In this next example, you again open the example.txt document and write “Hello World” and close it.
* You then open the example.txt file again, this time in append mode and add with the write function, “It’s nice to be here”, this will appear in the example.txt underneath the Hello World statement.
* If you had done this in the write mode, it would have overwritten the Hello World with It’s nice to be here, not appeared underneath it.

Slide 22 - Files — Write

* You can use while loops when handling files.
* In this example, we open a document called names.txt in append mode.
* We have to open the file first before we can add to it.
* Then while name has a value, it will keep adding that name to the names.txt until the name value is no more and then the loop will break and the file will close.

Slide 23 - Coding Time - Section B

* Now, it’s your time to try some coding, please refer to Section B on your worksheets.
* You’ll have to the end of the session to work on these.