Week 1: Introduction to Python

Introduction

What you will do in this worksheet:

1. Get Python up and running
2. Start to build up your own 'Python Cheat Sheet'
3. Create some Python functions in comparison to Java

If you are comfortable using Python then feel free to jump ahead to the 'Tasks' section!

Getting Started

In the zip file you will find two other documents along with this worksheet:

1. pythonSyntax.doc
2. sheet1.py

Cheat Sheet

The first of the files is a 'build-your-own' cheat sheet - fill it with Python syntax and notes that you find helpful! As you work through the exercises (and through later worksheets too), feel free to add examples and explanations of your own. This is to help you while learning Python so you don't have to keep googling the same questions! Also will come in handy when you inevitably forget all Python and are required to re-learn it :)

Python File

Steps to get the Python file running (using VSCode):

1. Open 'sheet1.py' within VSCode
2. Go to Terminal tab at the top of your screen and open a 'New Terminal'
3. Change directories to the zip folder. If you are unsure how to do this, see below.
4. To run your Python code, enter the following command inside the terminal:

```
    python3 sheet1.py
```

Note:
If your code doesn't run (you may get an error saying Python cannot find the 'random' package), then you will need to run the following code in terminal:

```
pip3 install random
```

Changing directories in terminal:
You can see which directory your terminal is currently in by using the following command:

```
pwd
```

You can list which files are currently in your directory using the following:

```
ls
```

To change directories to a new folder inside your current directory you can use (insert name into the command):

```
cd ./FolderName
```

To change directories to a completely new path, then just use (insert name into the command):

```
cd FolderPath
```

If you need to stop the python executor at any time throughout these exercises then you can press 'CTRL+c'.

Tasks

Task 1: ListToString

Complete the 'ListToString' function. A template of the function has been provided for you - please fill in the content. To test your function, go to the bottom of the 'sheet1.py' file and uncomment the first line and run the program.

INPUT: A list of strings.
OUTPUT: A string.
FUNCTION: Concatenate each of the strings in the list (in order) to produce a singular output string.

An example input and output would look like:

```
['A','B','C'] -> 'ABC'
```

You will have to complete research to figure out the following things - you are welcome to do this separately or split the tasks as a group:

- The syntax for inputting and outputting variables in a function
- How to find the length of a list
- How to produce a 'for loop' in Python to iterate through the list
- How to initialise a string
- How to concatenate strings

Task 2: GetEmptyCells

Complete the 'GetEmptyCells' function. You can test this function by uncommenting the second line in the testing section and running your program.

INPUT: A list of strings.
OUTPUT: A list of integers.
FUNCTION: Compute the indices of which elements in the list contain a singular space only.

An example input and output would look like:

```
[' ','B','C',' '] -> [0,3]
```

You will need to do more research to find out the following:

- How to initialise a list
- How to add items to the end of a list
- How to format 'if' statements

Task 3: GetRandomElement

Complete the 'GetRandomElement' function. This can be tested by uncommenting the third line in the testing section and running your code.

INPUT: A list (containing any data type).
OUTPUT: A singular element.
FUNCTION: Take a list and return a random element from the list. It should not matter what the list contains.

You will need to do more research to find out the following:

- How to use separate Python packages
- How to generate a random number

On line 1 of the code you will see that a library called 'random' has been imported. This is a separate library of functions that can be accessed by typing statements like 'random.FunctionName'.

Task 4: MaxIndices

Look at the following Java code and try figure out what the function does. Then, implement the same function in Python within the 'MaxIndices' function.

```
1  public static ArrayList<Integer> MaxIndices(ArrayList<Double>
       InputList){
2
3          ArrayList<Integer> OutputList = new ArrayList<Integer>();
4          Double Max = Collections.max(InputList);
5
6          for (int i = 0 ; i < InputList.size() ; i++){
7              if (InputList.get(i).equals(Max)){
8                  OutputList.add(i);
9              }
10         }
11         return OutputList;
12     }
```

Discussion

In what ways does Python differ from the languages you have learnt so far (Haskell, C, Java)? Which do you prefer?

Look at the following code in Java:

```
1  public class Test{
2
3      public static void main(String []args){
4         Boolean Run = false;
```

```
        if (Run) {
            N = 5 / 0;
        }
    }
}
```

Why won't this code run?

Try making the equivalent in a Python code - does it run?

What is the difference between dynamically typed and statically typed programming languages?

Next week

Next week we will use the functions that we have coded today to build our own Tic-TacToe game player.

Topic Suggestions

If you have any topics which you would like covered in these sessions then feel free to let me know by clicking here.
This may be content that isn't covered within your units which you would find useful or a particular topic within your current units that you feel you would benefit from more help with!
