**PYTHON WEEK 2 SUMMARY**

**Day 1 – Control Flows**

**Control flow statements –** A program’s control flow is the order in which the program’s code executes. The control flow of a Python program is regulated by conditional statements, loops, and function calls. Python has 3 types of control structures:

* Sequential – default mode
* Selection – used for decisions and branching.
* Repetition – used for looping, i.e., repeating a piece of code multiple times.

**Day 2 – Python Functions**

**Using functions –** You create functions by providing three pieces of information. The name of the function, a list of zero or more parameters, and, optionally, a block of code which provides the return value (a function can return nothing). You normally define functions in script files for the simple reason that you don’t want to type them more than once, you just want to edit a function (if necessary).

We must make a firm distinction between an argument and a parameter:

* Argument – The argument is the object used in an application of a function; it may be referenced by other variables or objects.
* Parameter – The parameter is a variable name that is part of the function and is a local variable within the function body.

There are three types of functions in Python:

* Ordinary functions – functions that follow mathematical procedures. They will receive an argument, perform a specific calculation with the argument, and return a result.
* Procedure functions – normally don’t return a result; they are called to execute a procedure. For example, a function can be created to set up a connection to a database.
* Factory functions - don’t take parameters. The function generates values. Some factory functions work by accessing an object encapsulated in a module. For example, you will access the random number generator encapsulated in the random module.

**Day 3 – Introduction to Modules**

**Introduction to Modules –** Modules are the pre-defined files that contain the python codes which depict the basic functionalities of class, methods, variables, etc. It consists of different functions, classes in a group of files inside a directory. Modules can also be termed as Libraries. These are basically the pre-defined methods that can be used to make the code more efficient and reduce redundancy.

**Mechanism of python modules –** the moment a module is imported through a program, Python Interpreter fetches the module from either of the following locations:

* The directory in the PYTHONPATH variable
* Default directory

The list of available modules in Python can be found out by executing the following command in the command prompt (interpreter shell): >>> help(“module”)

**Day 4 – Regular Expressions**

Regular expressions are a compact way of representing a collection of strings. Regular expressions (also known as ‘regexes’) are defined by using a language other than Python (a mini-language). Regexes are used for five main reasons:

* Parsing – Identifying and extracting pieces of text that match certain criteria.
* Searching – Locating substrings that can have more than one form like: ‘pet.png’, ‘pet.gif’, ‘pet.mpg’ while avoiding ‘carpet.gif’.
* Searching and replacing - Find substrings and replace specified words within the matched string or strings, e.g. replacing 071 234 5678 with +2771 234 5678.
* Splitting strings - Splitting a string where a certain character occurs, for example, split comma delimits strings every time a ‘,’ is found.
* Validation – Checking whether a string meets criteria, for example, to check whether an email address is in the standard format.