**The Python Program**

**Duration: 5 Days**

**Pre-Requisites:** All Participants attending this training should have good Programming experience.

**Course Contents:**

**DAY - I**

**Setting up and running Python**

* Setting up the editor ( any one ) Sublime text , Pycharm , Visual Studio Code
* Python program structure
* Hello World
* String formatting using different methods
* No braces only indentation
* understanding the differences between python 2.7 and 3.x

**Operators**

* arithmetic
* relational
* logical
* assignment
* bitwise
* membership
* identity operators
* is operator and == operator

**Control Structures**

* Conditionals
* If
* If-else
* If-elif
* Looping statements
* while
* for
* while else block and for else block
* break and continue
* range class
* Strings - Operations and String methods
* List - Operations and Methods
* **Exercises & Problems**

**DAY - II**

**Container Types**

* tuples
* sets - operations like union , intersection etc
* frozensets
* dictionaries
* Operations and Methods
* Mutable and Immutable data types
* List , Set and Dictionary comprehensions
* Understanding copy , deepcopy functions

**Functions**

* def keyword
* Syntax for defining the function
* Positional arguments
* Keyword arguments
* Default Arguments
* call-by-reference and call-by-value methods
* variable number positional and keyword arguments ( \*args and \*\*kwargs )
* global and local scope
* global and nonlocal keywords
* importing functions from the module.
* import and from statements
* Built-in functions - random , sys , math

**Iterators , Generators , coroutines and lambda**

* lambda syntax
* iter and next functions
* Generator Functions - yield as a last statement
* Generator expressions
* Coroutines - using yield as an expression

**Miscellaneous**

* zip , map , filter programming tools
* enumerate
* itertools
* Packing and Unpacking in python 3
* **Exercises & Problems**

**DAY - III**

**Modules and Namespaces**

* import statement
* if \_\_name\_\_ == '\_\_main\_\_’
* python files
* Package vs Folder
* \_\_init\_\_.py and \_\_all\_\_

**File Handling**

* Files and directories
* Text files
* Binary files
* Text file processing - methods read , readline and readlines
* write , writelines and append
* Context managers - with statement
* Understanding the \_\_enter\_\_ and \_\_exit\_\_ methods

**Useful modules**

* datetime
* time
* subprocess
* os

**Regular Expressions**

* Implementation of RE in python
* understanding metacharacters and metasymbols
* useful functions in re module
* match()
* search()
* findall()
* sub()
* compile()
* split()
* regular expression constants

**Logging**

* Logging setup
* Log Levels
* Console Logger
* File Logger
* **Exercises & Problems**

**DAY - IV**

**Object Orientation**

* Abstraction
* Encapsulation
* Data binding
* Data hiding
* Inheritance
* Polymorphism
* Method overloading in python
* Understanding the special methods: \_\_str\_\_ , \_\_repr\_\_ etc.,
* Operator overloading
* Understanding the decorator concepts
* passing and returning functions
* function decorators
* Class Decorators - class method , static method and property decorators
* Understanding the dunders

**Exception handling**

* try - except
* try - except and else
* try - except - finally
* try - except - else - finally
* writing custom exceptions

**Serialization**

* Working the different file formats
* CSV and EXCEL – csv module
* JSON Parsing –json and simple json modules
* XML file parsing, YAML file parsing.
* **Exercises & Problems**

**DAY - V**

**GUI programming using tkinter ( standard module )**

* importing the Tk class
* creating window object
* functions related to window object
* creating label objects
* methods related to label object
* creating button objects
* methods related to button object
* creating entry widget objects
* methods related to entry widget
* Simple GUI application

**Multi-threading**

* Multi-processing
* Multi-threading
* Creating thread
* Joining thread
* Daemon threads
* Cancelling threads
* Critical section and locking

**Database connectivity**

* creating database connection object – MySQL or Oracle ( default sqlite )
* Understanding modules used by various Databases.
* Installation of modules
* Querying and cursors - CRUD operation
* closing the cursor and database connection

**Networking**

* TCP/IP Basics - Introduction to basic networking terminolgies, layers
* Socket programming
* socket methods - Server methods and Client Methods
* Demo on Setting up a Simple TCP Client – Server
* understanding of sending a mail.

**Creating Virtual Environments**

* creating virtual environment - virtualenv and installing packages using - pip
* activating and deactivating the virtual environments
* pip command line options

**Unit Testing using pytest module**

* why unit test
* Testing frameworks
* pytest command and its options to test the functions
* pytest marks
* pytest - parametrize decorator
* **Exercises & Problems**

**Case Study:**

A Mini Project / Case Study will be provided by the trainer on the completion of training, the participants will complete the same send the project to the trainer within 1 week of completion of training. The trainer will evaluate and provide his comments / inputs to the participants.