Objective: "This training will enable participants to use, program & customise python scripts. This is an advanced course taking the learning to a deeper level. Apart from taking a deep dive into core python concepts, this training will also enrich the participants with decent knowledge of using python for regular scripting, Automation, regular expressions, threads and data analysis for different datasets in excel, csv, json etc", testing

Prerequisite

basic Python knowledge

Lab Setup Hardware

PC/Laptop

>= 8gb Ram

>= 20gb free hard disk space

Internet Access (mandatory)

Software Windows machines

any python IDE

Some packages may be needed to install during the course, which will need an active internet connection and installation permissions.

**Table of Contents:**

The training cannot be strictly followed as per the session-wise break up, and trainer will have to introduce some advance concepts early on, as most topics are inter-related. Trainer will pace up & slow down based on speed of participants.

**Assumptions :**

**It is assumed the participants are already aware of the following:**

**Python Basics**

data

variables

operators

print

None & del

**Data structures**

list & tuples

slicing

dictionaries

structuring data

sets & misc.

**Flow control**

Boolean data type

if elif

for in range

for each

while

for else

**Functions**

definition, scope & return values

important inbuilt functions

function as objects

nesting of functions

return, global, local, nonlocal

**Table of Contents**

**Day 1**

quick refresher of basics

particular questions/discussions on basics

**modules & packages**

creating modules

\_\_init\_\_.py for packages

important libraries

os

sys

datetime

**File handling**

Files and File Paths

Absolute vs. Relative Paths

os.makedirs() & os.path

split, size, listdir etc

open close

read write

shelve module

**oops**

python class

oops implementation.

private data

doc string

\_\_init\_\_

class & instance variables

**Advanced oops**

inheritance

super

call & repr

decorators within class

@class @static

**Day 2**

**Regex**

match

search

search and replace

regular expression modifiers

option flags, patterns

character classes

grouping with parenthesis

repetition

**Concurrent execution**

threads

Thread-Local Data

Thread Objects

Lock Objects

mutli processing

**Exception handling**

try except

raise

nesting of try excepts

else clause

finally

**Advanced function concepts**

anonymous functions

lambdas

map & reduce

generators

yield

high order functions

decorators

**Day 3**

**Serialization**

pickle & unpickle

load

dump

json files

load & loads

dump

**Testing in Python**

Different frameworks

dependencies

**UnitTest**

Introduction

setup() teardown()

testsuite class

Assertion

Test Discovery

Skip Test

Exceptions Test

Time Test

Signal Handling

extras:

internals of DS

YAML

XML

as web framework