Brainware Computer Academy Syllabus

Python (Module I) (Core Python)

Total 72 Classes / 144 Hours

Module I		
Python Core:	}	27 Classes
Theory Examination		1 Class

Total: **28 Classes**

Class logistics:
3 Classes a week of 2 hours each.

Study Materials:

Let Us Python By Yashavant Kanetkar, BPB Publications

Python: The Complete Reference by Martin C. Brown,Tata McGraw Hill Publication

Syllabus Core Python (Module I)

Week 1.1: Class 1: Algorithms and Flow Chart Introduction

- Algorithms and Flow Chart
- Develop logic using Flow Chart
- Different type of Programming Languages
- History & need of Python
- Practical: Practice

Week 1.2: Class 2: Python Introduction

- Develop Logic using Flow Chart
- Python Programming Methodology
- Application of Python
- Advantages of Python
- Installing Python
- Practical: Practice

Week 1.3: Class 3: Fundamental of Python Interactive mode

- Program structure
- Interactive Shell
- Translator
- Script files.
- Using IDE
- Identifiers, Literals
- Variables
- Practical: Practice

Week 2.1: Class 4: Fundamental of Python Script mode

- Working with Script mode
- Python Indentation
- Variables
- Arithmetic Operators
- Assignment Operator
- Input and Output statement
- Comments in Python
- Practical: Practice

Week 2.2: Class 5: Python Operators

- Relational Operators
- Logical Operators

Membership Operators
Identity Operators
Operators Precedence
Practical: Practice

Week 2.3: Class 6: Expression and Structure

Evaluating Expression
Type Casting
Conditional Statements
The if Statement
The if-else Statement
The if-elif Statement
Nested if Statements
Practical: Practice

Week 3.1: Class 7: Python Loop Structure

Looping and Iteration
The For Loop
The While Loop
Loop else Statement
Nested Loops
Break and Continue
range() function
Practical: Practice

Week 3.2: Class 8: Function

Pass statement
Functions
Using Function
Practical: Practice

Week 3.3: Class 9: Function

In-built functions
Built-in-Function call with parameter/s
Ceil(x)
Floor(x)
Fabs(x)
Exp(x)
Pow(x,y)
Sqrt(x)
Practical: Practice

Week 4.1: Class 10: Function

Abs(x)
Max(x,y,z, ……)
Min(x,y,z, ……..)
Cmp(x,y)
Len(s)
Range(start, stop, step)
Round(x[,n])
Practical: Practice

Week 4.2: Class 11: String Handling

Declaring a string in python
Using subscript
Initialisation of string
\n – using a new line for a string
String are immutable
Len(), capitalize(), find(), isalnum(), isalpha(), isdigit(), lower(), islower(), isupper(), upper()
Practical: Practice

Week 4.3: Class 12: String Handling

lstrip(), rstrip(), isspace(), istitle(), replace(old, new), join(), swapcase(), partition(sep)
split([sep[, maxsplit]]),
Practical: Practice

Week 5.1: Class 13: Function

Invoking UDF
Flow of Execution
Default Arguments, Named Arguments
Practical: Practice

Week 5.2: Class 14: Recursive Function and Lambda Function

Recursion Function
Use of recursion function
Lambda function
Practical: Practice

Week 5.3: Class 15: String handling

Introduction to String
Accessing Individual Elements
String Operators
Practical: Practice

Week 6.1: Class 16: Object Oriented Concept

Object Oriented Programming
Class and Object in Python
Inheritance
Practical: Practice

Week 6.2: Class 17: OOPs and its features

Polymorphism like
Function overloading
Function overriding
Practical: Practice

Week 6.3: Class 18: Collection List

Introduction to List
Creating List
Accessing List
Joining List
Mutable and Immutable
Practical: Practice

Week 7.1: Class 19: List

Replicating List
List Slicing
Liner List Manipulation
Stacks & Queues in list
Practical: Practice

Week 7.2: Class 20: Tuple

Introduction to Tuple
Creating Tuples
Accessing Tuples
Joining Tuples
Replicating Tuples
Practical: Practice

Week 7.3: Class 21: Dictionary

Dictionary
Introduction to Dictionary
Accessing values in dictionaries
Working with dictionaries
Properties
Practical: Practice

Week 8.1: Class 22: Collection Set

Introduction to Set
Accessing and Joining
Replicating and Slicing
Frozenset
Practical: Practice

Week 8.2: Class 23: Modules

Built-in Modules platform.system(), platform.dir(platform)
os.chdir(), os.mkdir(), os.getcwd(), os.rmdir(), os.listdir()
sys.exit(), sys.path(), sys.version()
math.pi, math.pow(), math.sqrt(), math.ceil()
Practical: Practice

Week 8.3: Class 24: Modules

statistics.mean(), statistics.median(),statistics.mode()
random.random(), random.random(1,100)
Importing Modules in Python Programs
Working with Modules
Practical: Practice

Week 9.1: Class 25: Exception Handling

Default Exception and Errors
Catching Exceptions
Raise an exception
Try.... except statement
Raise, finally blocks
Practical: Practice

Week 9.2: Class 26: User Define Exception

User Defined Exception
Raise User Define Exception
Practical: Practice

Week 9.3: Class 27: Theory Examination

Syllabus Advance Python (Module-II)
To be taught with Core Python

Week 10.1: Class 28: File Input / Output

- Text and Bytes files
- Opening a file
- Reading and Writing Files
- Practical: Practice