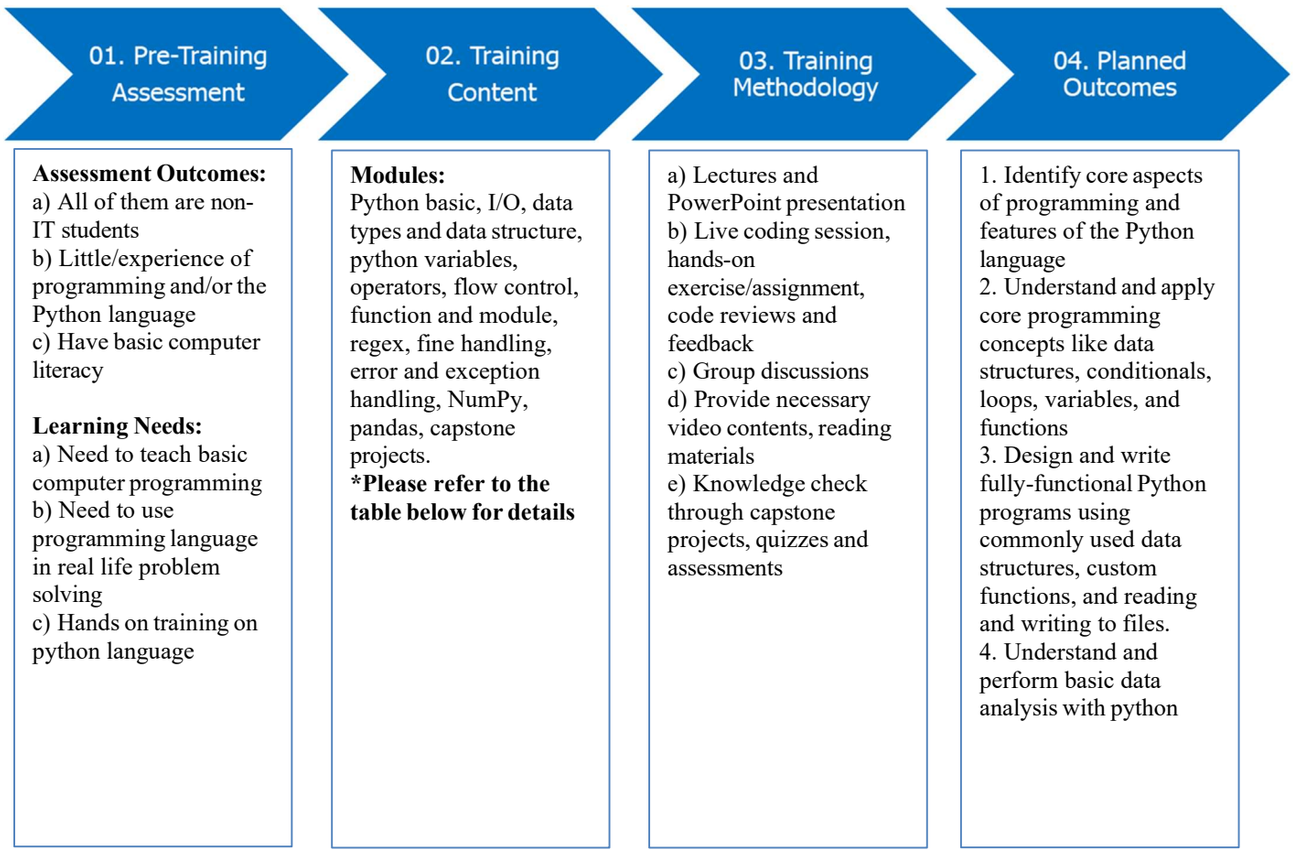
TRAINING APPROACH AND METHODOLOGY

Course Name: Introduction to Programming with Python



1. Pre-Training Assessment

Assessment Outcomes

a) All of them are non-IT students
b) Little or no experience of programming and/or the Python language
c) Have basic computer literacy

Learning Needs

a) Need to teach basic computer programming
b) Need to use programming language in real life problem solving
c) Hands on training on python language

2. Training Content

Please refer to the table below for details.

3. Training Methodology

a) Lectures and PowerPoint presentation
b) Live coding session, hands-on exercise/assignment, code reviews and feedback
c) Group discussions
d) Provide necessary video contents, reading materials
e) Knowledge check through capstone projects, quizzes and assessments

4. Planned Outcomes

1. Identify core aspects of programming and features of the Python language
2. Understand and apply core programming concepts like data structures, conditionals, loops, variables, and functions
3. Design and write fully-functional Python programs using commonly used data structures, custom functions, and reading and writing to files.
4. Understand and perform basic data analysis with python

Training Contents & Training Methodology & Planned Outcomes

Module	Session No. (3 hours each)	Topics	Training Methodology	Planned Outcomes
Python Basic	1	• What is Programming? • Python Language advantages and applications • History of Python Programming with important difference between python 2.x and python 3.x • Downloading & installing Python • Downloading & Installing Jupyter Notebook • Get Started with Python Coding: "Hello World" • Keywords in Python • Namespaces and Scope in Python • Statement, Indentation and Comment in Python • Structuring Python Programs • Assign values to variables in Python	a, b, c, d	1
Python Input/output	2	• Quiz/Assignment • Taking input in Python • Taking input from console in Python • Taking multiple inputs from user in Python • Python | Output using print() function • How to print without newline in Python? • Python | end parameter in print() • Python | sep parameter in print() • Python | Output Formatting	a, b, c, d, e	1
Python Data Types and Data Structure	3	• Introduction to DataTypes • Python Strings • Python List • Python Tuples	a, b, c, d, e	2, 3
	4	• Quiz/Assignment • Python Sets • Python Dictionary • Python Arrays	a, b, c, d, e	2, 3
Python Variables	5	• Variables, expression, condition and function • Global and local variables in python • Packing and unpacking arguments in python • Type conversion in python • Print single and multiple variable • Swap variable • Private variables • Name (A Special variable) in Python	a, b, c, d, e	2, 3

Python Operators	6	• Basic operator in python • Logical and bitwise not operator on boolean • Ternary operator • Division operator in python • Operator Overloading in Python	a, b, c, d, e	2, 3
	7	• Quiz/Assignment • Inplace and standard operators in python • Operator function in python • Inplace operator • Logic Gates in Python • Difference between == and is operator in Python • Python Membership and Identity Operators | in, not in, is, is not	a, b, c, d, e	2, 3
Python Control Flow	8	• Python Loops • Loops and Control Statements (continue, break and pass) in Python • Looping technique in python • range vs xrange in python • Chaining comparison in python	a, b, c, d, e	2, 3
	Midterm Examination and Evaluation			
	9	• else with for • switch function • Using iteration in python effectively • Python Itertools • Generators in python • Generators expression in python	a, b, c, d, e	2, 3
Python Function and Modules	10	• Functions in Python • class method vs static method in Python • Write an empty function in Python – pass statement • Return Multiple Values • Precision Handling • *args and **kwargs	a, b, c, d, e	2, 3
	11	• Quiz/Assignment • Python closures • Function Decorators • Decorators in Python • Decorators with parameters in Python • Memoization using decorators in Python • Help function in Python • Python | __import__() function • Python | range() does not return an iterator • Coroutine in Python • Built in useful Modules	a, b, c, d, e	2, 3
Python RegEx	12	• Quiz/Assignment • Python RegEx • Regular Expression (RegEx) in Python with Examples	a, b, c, d, e	2, 3, 4
Python File Handling	13	• File Handling • Write to File • Read File	a, b, c, d, e	2, 3, 4

		• Renaming and Deleting Files • Accessing Directories • File Methods • OS File/Directory Methods		
Python Errors and Exception Handling	14	• Quiz/Assignment • Exception handling • User defined Exception • Built-in Exception • clean up action • try and except in Python	a, b, c, d, e	2, 3, 4
Python Numpy	15	• Quiz/Assignment • Python Numpy • Numpy | ndarray • Numpy | Array Creation • Numpy | Data Type Objects • Data type Object (dtype) in NumPy • Numpy | Indexing • Numpy | Basic Slicing and Advanced Indexing • Numpy | Iterating Over Array • Numpy | Binary Operations • Numpy | Linear Algebra • Numpy | Sorting, Searching and Counting	a, b, c, d, e	2, 3, 4
Python Pandas	16	• Quiz/Assignment • Python | Pandas DataFrame • Creating a Pandas DataFrame • Dealing with Rows and Columns in Pandas DataFrame • Indexing and Selecting Data with Pandas • Boolean Indexing in Pandas • Conversion Functions in Pandas DataFrame • Iterating over rows and columns in Pandas DataFrame • Working with Missing Data in Pandas • Python | Pandas Series • Data analysis using Pandas • Read csv using pandas.read_csv()	a, b, c, d, e	2, 3, 4
Project Work (Four Projects)	17	• Project-1: Data Visualization using **Bokeh/ Matplotlib/ Seaborn etc.** • Project-2: Exploratory Data analysis and Visualization with Python • Assignment (Simple Data Analysis with Python)	a, b, c, d, e	2, 3, 4
	18	• Project-3: Exploratory Data analysis and Visualization with Python • Assignment (Simple Data Analysis with Python)	a, b, c, d, e	2, 3, 4
Project Work	19	• Project-4: Exploratory Data analysis and Visualization with Python • Assignment (Simple Data Analysis with Python)	a, b, c, d, e	2, 3, 4

Students Project Assessment	20	• Students Project Assessment • Question-Answer Session • Final Remark on the Introduction to Python Programming Course	a, b, c, d, e	2, 3, 4
Final Examination and Evaluation				
Total	**20 Training Sessions (60 hours) + 1 Midterm Examination + 1 Final Examination**			

Course Completion Criteria:
IICT, SUST will conduct midterm and final assessments. Upon successful completion of the program, and based on the final evaluation (**with at least an 60% score and 80% attendance**), each participant will be jointly awarded a certificate by the Public Department/Institute/Center/Universities and EDGE

Grading based on:

- Class attendance: 10%
- Quiz and Assignment(s): 20%
- Mid-term assessment: 20%
- Project: 25%
- Final Evaluation: 25%