CSA4103 Python Programming
Credits: 4C

Learning Outcomes	Suggested Pedagogical Processes
Basic understanding of concept of variable, operators, keyword control structure etc.	Presentation method will be applied along with hand on method is preferred
Acquire the knowledge of functions.	Discuss the concepts of functions and their usage with syntax and examples.
Get familiarity with the concepts of strings.	Classroom discussion about concepts of strings and the operators with examples.
Learn the concepts of Data structures using Python.	Explain the concepts and operations of Data structures.
Understand the concepts of Modules.	Explain the concepts of various modules.
Gain the knowledge about functionalities of error and exception handling.	Power point presentation will be used to show the concepts of Error and Exceptional handling. Demo method to show the functionality of error and exceptional handling.
Get familiarity with Standard Library.	Discuss about concepts of Standard Library.
Get familiarity with the concepts of GUI development.	Power point presentation will be used to show the concepts of GUI development. Demo method will be used to show effective way of developing a good GUI.

Unit No.	Title of Unit and Contents
I	**Introduction to Python** 1.1 History of Python 1.2 Need of Python Programming 1.3 Applications of Python Programming 1.4 Values 1.5 Variables and Keywords 1.6 Operators in Python 1.7 Operator Precedence 1.8 Expressions and Statements 1.9 Accepting Input and Displaying Output 1.10 Putting Comments
II	**Conditional Constructs and Looping** 2.1 if, if..else statement 2.2 while, for (range function) 2.3 break, continue, else, pass 2.4 Nested Loops 2.5 Use of Compound expression in conditional constructs and looping
III	**Functions : Importing Modules** 3.1 Invoking built-in functions 3.2 Functions from math module 3.3 Using random() and randint() functions of random module to generate random numbers 3.4 Composition 3.5 Invoking User-defined functions

	3.6 Passing Parameters (Default parameter values, keyword arguments) 3.7 Scope of Variables 3.8 Void functions and function returning values 3.9 Flow of execution
IV	**Strings** 4.1 Creating, Initializing and Accessing Elements 4.2 String Operators: +,*, in, not in, range, slice [n:m] 4.3 Comparing strings using relational operators 4.4 String functions and methods 4.5 Pattern matching
V	**Data Structures** 5.1 Concepts of Mutable lists: Creating, Initializing and Accessing elements in lists, Traversing, Updating and Deleting elements 5.2 List Operations: Joining, List slices, List functions and methods 5.3 Dictionaries: Concept of key-value pair, Creating, Initializing and Accessing elements in a Dictionary, Traversing, Updating and Deleting elements, Dictionary functions and methods 5.4 Tuples: Immutable Concept, Creating, Initializing and Accessing elements in a Tuple, Tuple functions
VI	**Modules** 6.1 Executing modules as scripts 6.2 The Module Search Path 6.3 "Compiled" Python files 6.4 Standard Modules 6.5 The dir() function 6.6 Packages: Importing * from a Package, Intra-Package References, Packages in Multiple Directories
VII	**Input and Output** 7.1 Output Formatting 7.2 Reading and Writing Files
VIII	**Errors and Exceptions** 8.1 Syntax Errors 8.2 Exceptions: Handling Exceptions, Raising Exceptions User-defined Exceptions 8.3 Defining Clean-Up Actions 8.4 Predefined Clean-Up Actions
IX	**Standard Library** 9.1 String Pattern Matching 9.2 Mathematics 9.3 Date and Time 9.4 Output Formatting
X	**GUI Development** 10.1 The simple GUI program in Python 10.2 Event-driven programming 10.3 Changing the layout 10.4 Getting input from the user 10.5 Examples on GUI: Designing a GUI

Learning Resources

1. Bruce J. Maclennan, Functional Programming: Practice and Theory, 1990
2. Greg Michaelson, An Introduction to Functional Programming Through Lambda Calculus (Dover Books on Mathematics) Paperback, 2011
3. Kenneth C. Louden, Programming Languages: Principles and Practice, 3rd Edition, 2013
4. Michael Dawson, Python Programming for the Absolute Beginner, 3rd Edition, Cengage Learning, 2011
5. David Beazley, Python Essential Reference, Third Edition, 2006
6. E-Books: python_tutorial. pdf, python_book_01.pdf
7. Mark Lutz, Learning Python, O'Reilly, 2009
8. https://docs.python.org
9. https://docs.python.org/3/tutorial/index.html