



Program	BACHELOR OF TECHNOLOGY (B.Tech)	Semester - 2
Type of Course	Engineering Science Course	
Prerequisite	Basic Programming Skills	
Rationale	Python is a modern language useful for writing codes for programming in the area of Server side Web development, Data Analytics, AI and scientific computing as well as production tools and game programming and many more. This course covers the basics of Python programming to harness its potential for modern computing requirements.	
Effective From A.Y.	2025-26	

Teaching Scheme (Contact Hours)				Examination Scheme				
Lecture	Tutorial	Lab	Credit	Theory Marks		Practical Marks		Total Marks
				SEE TH	IAT	SEE P	CCE	
2	-	2	3	70	-	30	-	180

SEE - Semester End Examination, IAT - Internal Assessment Test, CCE - Continues & Comprehensive Evaluation

Course Content		T - Teaching Hours W - Weightage	
Sr.	Topics	T	W
1	Unit-1 Introduction to Python Programming Features of Python Programming, Basic elements of Python: Keyword, Identifier, Comments, Variable. Lines and Indentation, Multiline Statement, Multiple Assignment, Inbuilt Data type, Using the Python interpreter, Execution of first program.	3	10
2	Unit-2 Data Types in Python List: Creation, Accessing list, list operations, Working with lists, Function and Methods. Tuple: Creation, indexing, Accessing tuples, Operations, Working, Functions and Methods, Set: Creation, modification, operations, Dictionary: Creation, key-value access, methods, iteration, Accessing values in dictionaries, Working with dictionaries, Augmented assignment operators, methods, example, string and tuples, references.	6	20
3	Unit-3 Operators and Control structures Conditional statements: If, If else, Nested if..else, If-elif-else Iterative statements, Loop: for, while, Loop control statements: break, continue, pass.	6	20
4	Unit-4 Functions and String Functions: Defining own functions with parameters, return values, recursion and return statements, the none value, keyword Arguments, input and output function, local and global scope of function, recursion, global variables, String and String operations: Creation, string methods, formatting, Concatenation, repetition, indexing, slicing, membership testing, length check, iteration, comparison.	6	20
5	Unit-5 Files handling and Debugging File Handling: Introduction to file handling in Python, files and directories, file reading and writing, additional file methods, working with files, working with directories, read and write to/from CSV File, Debugging: Raising exceptions, getting the traceback as a string, assertions, logging, IDLE's debugger.	5	20
6	Unit-6 Object Oriented Programming Concepts and GUI	4	10



Course Content		T - Teaching Hours W - Weightage	
Sr.	Topics	T	W
	Object Oriented programming concepts: The object-oriented paradigm, Class and Instance in python, Inheritance, Polymorphism and custom exceptional classes, Encapsulation and Information hiding, Multithreading, Writing GUI in python: Tkinter tool, example GUI, adding a button, widgets, Drawing using turtle, Other GUI tools.		
Total		30	100

Suggested Distribution Of Theory Marks Using Bloom's Taxonomy						
Level	Remembrance	Understanding	Application	Analyze	Evaluate	Create
Weightage	20	40	35	5	0	0

NOTE : This specification table shall be treated as a general guideline for the students and the teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Course Outcomes	
At the end of this course, students will be able to:	
CO1	Describe the basic elements of python programming to solve various problem.
CO2	Use Python built-in data types such as lists, tuples, sets, and dictionaries for data manipulation.
CO3	Analyze the use of operators and control structures to control program flow in Python.
CO4	Apply functions and string operations to develop modular Python programs.
CO5	Apply file handling and debugging techniques to manage data and errors in Python programs.
CO6	Apply object-oriented programming concepts and GUI tools to develop Python applications.

CO PO Mapping						
CO	CO - 1	CO - 2	CO - 3	CO - 4	CO - 5	CO - 6
PO - 1						
PO - 2						
PO - 3						
PO - 4						
PO - 5						
PO - 6						
PO - 7						
PO - 8						
PO - 9						
PO - 10						
PO - 11						

**Reference Books**

1.	Python: The Complete Reference (TextBook) By Martin C. Brown McGraw Hill Education
2.	Think Python : How to Think Like a Computer Scientist (TextBook) By Allen B. Downey Green tea press 2nd
3.	Introduction to Computation and Programming Using Python By John V Guttag MIT Press
4.	Python Programming Fundamentals - A Beginner's Handbook By Nischay kumar Hegde
5.	Fundamentals of Python – First Programs By Kenneth A. Lambert CENGAGE Publication
6.	Introduction to Python for Engineers and Scientists By Sandeep Nagar Apress

List of Practical

1.	Demonstrates various IDEs having a python environment to perform Python programs.
2.	Write a program to demonstrate the use of list, dictionary, tuple and set.
3.	Write a python program that demonstrates the use of looping and branching statements.
4.	Write a program to implement functions scoping and recursion.
5.	Write a program to implement string functions.
6.	Write a python program to demonstrate the use of class and object in python.
7.	Write a program to understand the working of exception handling and assertions.
8.	Write a python program to export CSV files with read and write operations.
9.	Draw graphics using Turtle tool.
10.	Develop applications to implement GUI programming using Tkinter.