

**CE 291: PYTHON PROGRAMMING SKILLS**  
**B.TECH 4<sup>TH</sup> SEMESTER (CIVIL)**  
**PROGRAMME ELECTIVE - I**

---

**Credits and Hours:**

Teaching Scheme	Theory	Practical/Tutorial	Total	Credit
Hours/week	2	4	6	4
Marks	100	100	200	

**A. Objective of the Course:**

The objectives of the course are:

- To describe the core syntax and semantics of Python programming language.
- To illustrate the process of structuring the data using lists, dictionaries, tuples and sets.
- To introduce function - focussed programming paradigm through python.
- To infer the Object-oriented Programming concepts in Python.
- To indicate the use of regular expressions and built-in functions to navigate the file system.
- To train in the development of solutions using modular concepts.

**B. Outline of the course:**

Sr. No.	Title of the unit	Minimum number of hours
1.	Basics of Python	01
2.	Variables and Data Types	04
3.	Loops and Conditional Control	06
4.	Advanced Data Structures	04
5.	Functions	02
6.	Object Oriented Programming in python	08
7.	Reading and Writing Files	03
8.	Packages and Modular Programming	02

Total Hours (Theory): 30

Total Hours (Lab): 60

Total Hours: 90

### C. Detailed Syllabus:

<b>1. Basics of Python</b>	<b>01 Hour</b>	<b>03%</b>
1.1 Introduction to Python		
1.2 Installation and Environment Setup		
1.3 Simple Python Program Practice		
<b>2. Variables and Data Types</b>	<b>04 Hours</b>	<b>13%</b>
2.1 Variables, Operators, Precedence and Associativity		
2.2 Data Types , Indentation, Comments, Reading Input, Print Output		
2.3 Memory Optimization		
2.4 Type Conversion		
<b>3. Loops and Conditional Control</b>	<b>06 Hours</b>	<b>20%</b>
3.1 The if... Decision Control Flow Statement, The if...else Decision Control Flow Statement, The if...elif...else Decision Control Statement		
3.2 Nested if Statement, The while Loop, The for Loop,		
<b>4. Advanced Data Structures</b>	<b>04 Hours</b>	<b>13%</b>
4.1 Lists- Basic List Operators, Replacing, Inserting, Removing an Element, Searching and Sorting Lists, Tuples.		
4.2 Dictionaries- Dictionary Literals, Adding and Removing Keys, Accessing and Replacing Values; Traversing Dictionaries.		
4.3 Sets, Strings, Files and their Libraries		
<b>5. Functions</b>	<b>02 Hour</b>	<b>07%</b>
5.1 Defining a Function, Calling a Function		
5.2 Pass by Reference vs Value, Function Arguments, Required and Default Arguments, Keyword Arguments and Variable-Length Arguments		
5.3 The Anonymous Functions, The Return Statement		
5.4 Scope of Variables		
<b>6. Object Oriented Programming in python</b>	<b>08 Hours</b>	<b>27%</b>
6.1 Object-Orientation Introduction, Class Definition, Class Objects, Instance Objects, Method Objects		
6.2 Class and Instance Variables, Inheritance, Private Variables, The Polymorphism		

7.	Reading and Writing Files	03 Hour	10%
7.1	Reading Files, Parsing and Processing Files		
7.2	File storage, File operations (delete, copy)		
8.	Packages and Modular Programming	02 Hour	07%
8.1	Executing modules as Scripts, Import and Use of Built-in Modules, Accessing Third Party Packages and Modules		
8.2	Creating Custom Modules and Packages, Importing and Using Custom Modules		

#### D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board, etc.
- Attendance is compulsory in lectures and laboratory.
- Internal exams will be conducted as per pedagogy as a part of internal theory evaluation.
- Assignments based on course content will be given to the students at the end of each unit/topic and will be evaluated at regular interval.
- Surprise tests/Quizzes/Seminar will be conducted.
- The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.
- Practicals/Tutorials related to course content will be carried out in the laboratory.

#### E. Student Learning Outcomes:

At the end of the course the students will be able to:

- Interpret the fundamental python syntax, semantics and fluent in the use of python control flow statements.
- Determine the methods to create and manipulate python programs by utilizing the data structures like lists, dictionaries, tuples and sets.
- Express proficiency in the handling of strings and functions.
- Articulate the Object-Oriented Programming concepts such as inheritance and polymorphism as used in Python.
- Identify the commonly used operations involving file systems, and regular expressions.
- Execute modules as a script.

## F. Recommended Study Material:

### ❖ Text Books:

1. Nagar, Sandeep, "Introduction to Python for Engineers and Scientists - Open Source Solutions for Numerical Computation", Apress, 2018.
2. Kenneth Lambert, "Fundamentals of Python: First Programs", ISBN-13: 978-1337560092, Cengage learning publishers, first edition, 2012.
3. Magnus Lie Hetland, "Beginning Python from Novice to Professional", Third Edition, Apress, 2017
4. Reema Thareja, "Python Programming using Problem Solving Approach", ISBN-13:978-0-19-948017-3, Oxford University Press, 2017.

### ❖ Reference Books:

1. David Beazley, Brian K. Jones, "Python Cookbook", 3rd edition, O'REILLY, 2016
2. Brett Slatkin, "Effective Python: 59 Specific Ways to Write Better Python", Novatec, 2016
3. Allen Downey, "Think Python: How to Think Like a Computer Scientist", Green Tea Press, 2015
4. Mark Lutz "Learning Python", 4th Edition, O'REILLY, 2016
5. Vamsi kurama, "Python programming: A modern approach", ISBN-978-93-325-8752- 6, Pearson, 2018.
6. W. Chun, "Core python programming", ISBN-13: 978-0132269933, Pearson, 2nd edition, 2016.

### ❖ Web Materials:

1. <https://www.python.org/>
2. <http://www.diveintopython3.net/>
3. <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django>
4. <https://www.fullstackpython.com/django.html>
5. <https://codelabs.developers.google.com/>
6. <https://www.tutorialspoint.com/python/index.htm>

### ❖ Software:

1. Python IDLE
2. Anaconda Python
3. PyCharm

## LIST OF PRACTICALS

Practical No.	Name of Practical
1	Setting Transition curve & vertical curve using python
2	Determination of capacity of reservoir and volume of borrow pits
3	Determining areas from plans, trapezoidal rule- Simpson's rule
4	Estimation of discharge through notches and weirs
5	Estimation of time of emptying a reservoir or a tank
6	Estimation of discharge through Venturimeter
7	Developing the programme to calculate the bending moment and shear force & plot the diagrams
8	Calculation of deflections of determinate structures & plot the diagrams
9	Development of programme to obtain moment of inertia of regular shapes