

**SEMESTER-III**

Course Code	PCS21C07J	Course Name	PYTHON PROGRAMMING	Course Category	C	Professional Core			
						L	T	P	C
						4	0	2	5

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computer Science	Data Book / Codes/Standards			

Course Learning Rationale (CLR):		The purpose of learning this course is to:		
CLR-1 :	Describe the core syntax and semantics of Python programming language.			
CLR-2 :	Discover the need for working with the strings and functions.			
CLR-3 :	Illustrate the process of structuring the data using lists, dictionaries, tuples and sets.			
CLR-4 :	Indicate the use of regular expressions and built-in functions to navigate the file system.			
CLR-5 :	Infer the Object-oriented Programming concepts in Python.			
CLR-6 :	Understand Event Driven Programming			

Learning		
1	2	3
Level of Thinking (Bloom)	Expected Proficiency (%)	Expected Attainment (%)

Program Learning Outcomes (PLO)														
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Fundamental Knowledge	Application of Concepts	Link with Related Disciplines	Procedural Knowledge	Skills in Specialization	Ability to Utilize Knowledge	Skills in Modeling	Analyze, Interpret Data	Investigative Skills	Problem Solving Skills	Communication Skills	Analytical Skills	PSO 1	PSO 2	PSO 3
L	H	-	H	L	-	-	-							
M	H	L	M	L	-	-	-					-	-	-
M	H	M	H	L	-	-	-					-	-	-
M	H	M	H	L	-	-	-					-	-	-
H	H	M	H	L	-	-	-					-	-	-

Course Learning Outcomes (CLO):		At the end of this course, learners will be able to:		
CLO-1 :	Develop, document, and debug modular python programs to solve computational problems			
CLO-2 :	Select a suitable programming construct and data structure for a situation.			
CLO-3 :	Use built-in strings, lists, sets, tuples and dictionary in applications.			
CLO-4 :	Define classes and use them in applications			
CLO-5 :	Use files for I/O operations.			

Duration (Hour)	18	18	18	18	18
S-1	SLO-1	An introduction to python programming	Arrays, Array methods	Equality: Object Identity, Structural Equivalence	Errors and exceptions, Exception handling in Python
	SLO-2	Structure of a Python program	Strings ,	Advanced list processing, List comprehension	Exception handling methods, Illustrate exception handling in Python
S-2	SLO-1	understanding Python interpreter and Python Shell	String methods	Conversion of list to array, tuple, string	Introduction to modules,
	SLO-2	Datatypes	mutable strings,	Conversion of array, string, tuple, dictionary to list	Important modules in Python
S3	SLO-1	Example program using all data types	Immutable strings	Tuples	Creating modules
	SLO-2	Example program using variables	String module,	tuple operation	accessing modules
S4	SLO-1	String literals	Sum array of numbers	Tuple methods	Namespaces and its methods
	SLO-2	Escape Sequences	Funtions	Introduction to dictionary,	Locating modules, dir()
S5-6	SLO-1	Laboratory 1: Write a Python code to display system information using pywhois	Laboratory 4: Make a simple calculator	Laboratory 7: Program to Transpose a Matrix	Laboratory 10: Program using recursive function. Program to illustrate exception handling in Python
	SLO-2			Program for sorting using list Using a List to Find the Median of a Set of Numbers	
					Laboratory 13: Program using classes and methods



Duration (Hour)		18	18	18	18	18
S7	SLO-1	String Concatenation	Function arguments	Operations	PYTHONPATH	Rules for Defining a Simple Class
	SLO-2	Variables	Anonymous functions,	Methods,	Packages,	Rational Number
S8	SLO-1	assignment statement	Illustrate functions using python	Add, remove a key in dictionary	Creating packages	Arithmetic and Operator Overloading
	SLO-2	Program Comments	Set declaration	Accessing values	accessing packages	Comparison Methods,
S9	SLO-1	Doc Strings	Set operation	Replacing Values,	Default (Keyword) Arguments,	Equality and the eq Method
	SLO-2	Numerical Datatypes	Set methods	Traversing a dictionary	Functions as First-Class Data Objects	Input of Objects
S10	SLO-1	Character sets	Introduction to Lists	Introduction to file	Mapping	the try-except Statement
	SLO-2	Arithmetic expressions	List literals	file creation	Filtering	Inheritance
S 11-12	SLO-1	Laboratory 2: The Magic 8 Ball is a toy used for fortune-telling or seeking advice.	Laboratory 5: Arrays and strings	Laboratory 8: Program on dictionary operations. Program on dictionary methods	Laboratory 11: Write a python program to define a module and import a specific function in that module to another program	Laboratory 14: Python Program for Operator overloading
	SLO-2					
S13	SLO-1	Understanding error messages	Basic list operations	File operations	Reducing	Hierarchies
	SLO-2	Logical operators	Replacing an Element in a List	Format operators	Using lambda to Create Anonymous Functions	Modeling
S14	SLO-1	Definite iteration : For loop	List methods with illustration,	Directory functions,	Standard Libraries in Python	Polymorphic Methods
	SLO-2	Selection : if statement	Program to List Methods for Inserting Elements	File positions	Introduction to classes	Abstract Classes
S15	SLO-1	if else statement	Example program to Replace an Element in a List	Example program to access and manipulate files	Design with Classes	The Costs of object oriented programming
	SLO-2	Example program using if and if else	Sorting and searching a list	Example program to read and write text and numbers	Objects	Benefits of Object-Oriented Programming
S16	SLO-1	Conditional iteration :while loop	Aliasing	Recursive functions	Classes	Event-Driven Programming,
	SLO-2	Example program using while loop	mutator methods	Abstract functions	An example for class	Example for Event-Driven Programming
S 17-18	SLO-1	Laboratory 3: Check whether a number is prime or not, Python Program to Generate a Random Number	Laboratory 6: Program to illustrate set operations and its methods. Program to illustrate list operations and its methods. Program for list comprehension	Laboratory 9: Program to create and modify text file in Python Program for word count in text file.	Laboratory 12 : Programs to illustrate lambda functions with mapping, filtering, reducing and substituting	Laboratory 15: Program using polymorphism, abstract classes
	SLO-2					

<b>Learning Resources</b>	Kenneth A. Lambert, (2011), "The Fundamentals of Python: First Programs", Cengage Learning
---------------------------	--

Learning Assessment											
Bloom's Level of Thinking		Continuous Learning Assessment (50% weightage)								Final Examination (50% weightage)	
		CLA – 1 (10%)		CLA – 2 (10%)		CLA – 3 (20%)		CLA – 4 (10%)#			
		Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	20%	20%	15%	15%	15%	15%	15%	15%	15%	15%
	Understand										
Level 2	Apply	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%
	Analyze										
Level 3	Evaluate	10%	10%	15%	15%	15%	15%	15%	15%	15%	15%
	Create										
	Total	100 %		100 %		100 %		100 %		100%	



# CLA – 4 can be from any combination of these: Assignments, Seminars, Short Talks, Mini-Projects, Case-Studies, Self-Study, MOOCs, Certifications, Conf. Paper etc.,

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr. S. Karthik, Assistant Consultant, Tata Consultancy Services	Dr.S.Sasikala, Associate Professor and Head, Dept. of Computer Science, University of Madras	1. Mrs. E.Aarthi
		2. Dr. P. Muthulakshmi

