

Branch Name:	Computer Engineering / Information Technology / Information and Communication Technology/ Artificial Intelligence and Data Science / Artificial Intelligence and Machine Learning / Electronics and Communication Engineering
Program Code:	ET103 / ET105 / ET108 / ET107 / ET114 / ET116
Course Title:	Python Programming Practical
Course Code:	2ET1000110P
Pre-requisite Course:	Fundamentals of OOP

Course Objective:

The objectives of this course are:

- To provide students with contemporary knowledge in PYTHON PROGRAMMING.
- To explore real-world software development challenges with python programming.

Teaching and Examination Scheme:

Credit	Teaching Scheme Hours per week				Evaluation Scheme (Marks)								
					Theory (Marks)				Practical (Marks)				Total (Marks)
	L	T	P	Total	SEE		CCA		SEE		CCA		
					Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	
1	-	-	2	2	-	-	-	-	10	25	10	25	50

LAB/Practicals:

#	List of Experiments	Contact Hours
1	Create a program that asks the user to enter their name and their age. Print out a message addressed to them that tells them the year that they will turn 100 years old.	1
2	Ask the user for a number. Depending on whether the number is even or odd, print out an appropriate message to the user.	1
3	Write a Python program to convert GPAs to letter grades according to the following rules: GPAs : Grades $\Rightarrow$ 4.0 : A+, 3.7 : A, 3.4 : A-, 3.0 : B+, 2.7 : B, 2.4 : B-, 2.0 : C+, 1.7 : C, 1.4 : C-, below : F	1
4	Create a program that asks the user for a number and then prints out a list of all the divisors of that number.	1
5	Write a program that takes list and print only the even elements of the list .	1
6	Write a Python program to find a given string is happy or unhappy. A string is happy if every three consecutive characters are distinct.	1
7	Write a Python program to find the no of vowels in a given string.	1
8	Write a password generator in Python. Be creative with how you generate passwords - strong passwords have a mix of lowercase letters, uppercase letters, numbers, and symbols. The passwords should be random, generating a new password every time the user asks for a new password.	1
9	Ask the user for a string and print out whether this string is a palindrome or not.	1
10	Write a function that takes a string (asks the user for a long string containing multiple words) and Print back to the user the same string, except with the words in backwards order.	1
11	Implement a function that takes as input three variables, and returns the largest of the three.	1
12	Write a function to generate N fibonacci numbers. Make sure to ask the user to enter the value of N.	1
13	Write a function that takes an ordered list of numbers and another number. The function decides whether or not the given number is inside the list using binary search.	1
14	Write a Python program to sort a list of tuples using Lambda.	1

15	Develop a program to Sort Python Dictionaries by Key or Value using Lambda.	1
16	Write a program to keep track of friend's birthdays, and be able to find that information based on their name. Create a dictionary of names and birthdays. When you run your program it should ask the user to enter a name, and return the birthday of that person back to them.	1
17	Write a program to read user name of students from console and save in a file(.txt).	1
18	Given file name of .txt file that has a list of a bunch of names, count how many of each name there are in the file, and print out the results to the screen.	1
19	Write a program to building Hangman game. In the game of Hangman, a clue word is given by the program that the player has to guess, letter by letter. The player guesses one letter at a time until the entire word has been guessed. (In the actual game, the player can only guess 6 letters incorrectly before losing).	1
20	Write a Python class BankAccount with attributes like account_number, balance, date_of_opening and customer_name, and methods like deposit, withdraw, and check_balance.	1
21	Write a Python program that creates two threads to find and print even and odd numbers.	1
22	Write a Python program that executes division and handles an ArithmeticError exception if there is an arithmetic error.	1
23	Write a Python program to print a triangle using Turtle.	1
24	Write a Python program that implements a temperature converter application using Tkinter, allowing users to convert between Celsius and Fahrenheit.	2
25	Write a Python GUI program to create calculator application using radio buttons widgets of tkinter module.	2

List of References:

1. Introduction to Computation and Programming Using Python by John V Guttag (PHI)
2. Core Python Programming by R. Nageswara Rao , Dreamtech
3. Core Python Programming Second Edition by Wesley J. Chun.(PHI)
4. Professional Python by Luke Sneeringer, Wrox
5. A Practical Introduction to Python Programming by Brian Heinold

E-Resource / Web Links:

- <https://www.python.org/>
- <https://docs.python.org/2/library/turtle.html>
- <https://nptel.ac.in/courses/117106113/34>

Course Learning Outcomes (CLO):

Course Outcomes	Description	Bloom's Taxonomy Level
COL1	To code in Python	Understanding (2), Applying (3)
COL2	To create applications in Python	Create (2), Evaluate (3)
COL3	To design real world problem in Python	Analyze (2), Create (2), Evaluate (1)

Mapping of CLOs with POs & PSOs:

Course Learning Outcomes	Program Outcomes (POs)												Program Specific Outcomes (PSOs)			
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CLO1	H		L						L				H			
CLO2	H	M	M		L	M		L	M				H	M	M	L
CLO3	H	M	H	L		M		L	M		L	L	H	H	H	M

H: High, M: Medium, L: Low