

List of Course Experiments: 1

Learning outcome: Students will be able to perform basic python programs.

Sl. No.	Program Listing
1	Write a python to find the factorial of a given input number.
2	Write a Python program to check whether a given input number is a prime number or not.
3	Write a Python program to display Fibonacci series upto n numbers.
4	Write a Python program to find the reverse of a given integer number.
5	Write a Python program to calculate the sum of all numbers from 1 to a given number
6	Write aPython program to print the following number pattern using loop. * * * * * * * * * * * * * * *
7	Write a Python program to check a given number a palindrome or not.
8	Write a Python program find the sum of the following series. Also print the value of the sin(x) from math module and verify the result. $x - x^3/3! + x^5/5! - \dots n$ terms

List of Course Experiments: 2

Learning outcome: Students will be able to perform List operations.

Sl. No.	Program Listing
1	Write a program that reverses the order of elements in a given list without using the built-in reverse() function.
2	Write a program that sorts a list of numbers in ascending order using bubble sorting algorithm.
3	Write a program that generates the Fibonacci sequence as a list up to a specified number of terms.
4	Write a program that takes a list of numbers and generates a histogram displaying the frequency of each number.
5	Write a program that finds the intersection and union of two lists.
6	Write a program that removes duplicate elements from a list and make a new list with only the unique elements.
7	Write a program that rotates the elements of a list by a specified number of positions to the left or right.
8	Write a program that uses list comprehensions to generate lists of numbers, even numbers, odd numbers, squares, etc.
9	Write a program that produces a list of prime numbers.
10	Write an interactive program to performs all operations of stack using list.

List of Course Experiments: 3

Learning outcome: Students will be able to perform String operations.

Sl. No.	Program Listing
1	Write a program that checks if a substring exists within a given string and prints a message indicating whether it was found.
2	Write a program that counts the number of occurrences of all the characters in a given string.
3	Write a program to find the number of vowels, consonants, digits, and whitespaces in the input string.
4	Write a program that reverses a given string and prints the reversed string.
5	Write a program to check whether the input email-ID is valid or not. The common rules for a valid email address: No spaces, no special characters excepts (@ hyphens (-), and periods(.), can consist of letters (a-z, A-Z), digits (0-9).
6	Write a program that check if the input sentence is palindrome or not, ignoring spacing, punctuation and capital. e.g. never odd or even, no lemon, no melon, My gym
7	Write a program to check whether the input strings are anagrams of each other or not. e.g. listen, silent
8	Write a program that counts the number of words in a given text string. Consider words as sequences of characters separated by spaces or punctuation. Display the list of words.
9	Write a program that converts English text into Pig Latin. Following the rules: If the word starts with a vowel, "way" is added to the end of the word (e.g., "apple" becomes "appleway"). If the word starts with one or more consonants, those consonants are moved to the end of the word, followed by "ay" (e.g., "banana" becomes "ananabay").
10	Write a program that implement a simple Caesar cipher encryption and decryption program that works on strings.

List of Course Experiments: 4

Learning outcome: Students will be able to implement user-defined functions.

Sl. No.	Program Listing
1	Write a program to compute the mean, mode and median of the given a list of numbers by using functions.
2	Write a function to check whether a given number is prime number or not.
3	Write a function to calculate the factorial of a number using recursion.
4	Write a function to implement binary search using recursion.
5	Write a function to compute the LCM of two numbers.
6	Write a conversion program to convert decimal to binary and binary to decimal using functions.
7	Write a program to add, multiply, transpose of matrices by using functions.
8	Write a program that converts temperatures between Celsius and Fahrenheit using lambda functions.
9	Write a program that calculates and displays the volume of sphere, cylinder, cube and cone, based on user input. Define a function for each shape, using default values for some of the arguments (e.g., radius, length).
10	Write a program that calculates the average of a variable number of values passed as arguments to a function.
11	Write a password validation program that checks if a given password meets certain criteria (e.g. at least 8 characters, one capital case, one lower case, one numeric and one special character). Use a nested function to validate each criterion separately.
12	Write a program that stores and manages student information using a Python function with keyword arguments. Name (required), Age (optional, default to None) and Grade (optional, default to None). Perform the following tasks: add, delete, update and display student information.

List of Course Experiments: 5

Learning outcome: Students will be able to perform dictionary operations.

Sl. No.	Program Listing
1	Write a program that takes a list of names and creates a dictionary with the names as keys and their lengths as values.
2	Write a program that takes a string as input and creates a dictionary of character-frequency pairs.
3	Write a program that takes a dictionary of student names and their scores, and calculates and displays the average score.
4	Write a program that takes a dictionary and creates a new dictionary with keys and values swapped.
5	Write a program that takes two dictionaries and merges their contents, resolving conflicts for duplicate keys.
6	Write a program that takes a dictionary and sort by values in ascending and descending order.
7	Write a program to perform different operations on dictionary like add a key, remove a key, display the keys, display the values, search for a key.

List of Course Experiments: 6

Learning outcome: Students will be able to perform file operations and exception handling.

Sl. No.	Program Listing
1	Write a program that generate 100 random numbers in range 10 to 1000 and store the numbers (10 numbers a line) into a new file name “numbers.txt”.
2	Write a program to read the file named “numbers.txt” and sort them in ascending order.
3	Write a program to merge two input files into a single file.
4	Write a program to display frequency of words in the input file.
5	Write a program to count the number of lines, words, characters in the input file.
6	Write a program to read student name and marks and store them in a file.
7	Write a program to read student names and marks of five subjects from the file and compute the average marks.
8	Write a program that asks the user to enter two numbers and then performs division on these numbers. Handle any possible exceptions that may occur during the division operation, such as Zero Division Error or Value Error if the user enters invalid input.
9	Create a custom exception class called "Negative Value Error." Write a Python program that accepts a positive integer from the user and raises a "Negative Value Error" if the user enters a negative number. Handle this custom exception to display an appropriate error message.
10	Write a Python program that reads an integer from the user and attempts to perform a specific operation on it. Use a try-except-else-finally block to handle exceptions. The else block should execute when no exceptions occur, and the finally block should always execute.
11	Write a program to add, subtract, division and multiplication of complex numbers by using module.
12	Write a program to compute mean, mode, median of the list of numbers by using module.