PYTHON PROGRAMMING LAB
(Skill Oriented Course-1)

Course Code: 22IT11S1

L	T	P	C
			2

COURSE OUTCOMES:

At the end of the Course the student shall be able to

CO1: apply comprehensions, different Decision-Making statements and Functions. (L3)

CO2: implement various data types like lists, tuples, strings. (L3)

CO3: use different File handling operations and Maps. (L3)

CO4: apply Object oriented programming in Python types. (L3) **CO5:** Use
Pandas and Matplotlib in developing various applications. (L3)

LIST OF PROGRAMS:
(Any Twelve of the following Programs should be carried out)

1. **Input and Output**

   (a) Write a program to find the largest element among three Numbers.

   (b) Write a program to print the sum of all the even numbers in the range 1 - 50 and print the even sum.

   (c) Write a Program to display all prime numbers within an interval of given X1 & X2.

2. **Variables and Functions**

   a. Write a program to swap two numbers without using a temporary variable.

   b. Write a program to define a function with multiple return values.

   c. Write a program to define a function using default arguments.

3. **Loops and conditionals**

   a. Write a program to print the following patterns using loop:
   ```
   *
   ** *** ****
   ```

   b. Write a program to print multiplication table of a given number X1 to range X2.

4. **Strings**

   a. Write a program to find the length of the string without using any library functions.

   b. Write a program to check if two strings are anagrams or not.

   c. Write a program to check if the substring is present in a given string or not.

5. **Lists**

   a. Write a program to perform the given operations on a list:
      i. add   ii. insert   iii. slicing

   b. Write a program to perform any 5 built-in functions by taking any list.

c. Write a program to get a list of the even numbers from a given list of numbers. (use only comprehensions)

6. Tuples
a.Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
b.Write a program to return the top 'n' most frequently occurring chars and their respective counts. e.g. aaaaabbbbcccc, 2 should return [(a 5) (b 4)]

7. Sets
a. Write a program to count the number of vowels in a string (No control flow allowed).
b. Write a program that displays which letters are present in both strings.
c. Write a program to sort given list of strings in the order of their vowel counts.

8. Dictionaries
a. Write a program to check if a given key exists in a dictionary or not.
b. Write a program to add a new key-value pair to an existing dictionary.
c. Write a program to sum all the items in a given dictionary.

9. Files
a.Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper case words from source must be lowered.
b.Write a program to find the most frequent words in a text. (read from a text file).

10. Classes
a. Write a Python class named Person with attributes name, age, weight (kgs), height (ft) and takes them through the constructor and exposes a method get_bmi_result() which returns one of "underweight", "healthy", "obese".
b. Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle.

11. Arrays
a. Write a program to create, display, append, insert and reverse the order of the items in the array.
b. Write a program to add, transpose and multiply two matrices.

12. Python Maps , Filters & Generators:

a. Accept two lists, one list represents temperatures in Fahrenheit and another list represents temperatures in Celsius. Perform map operations Fahrenheit-Celsius and Celsius-Fahrenheit using lambda
b. Create a Fibonacci sequence that contains 'N' terms, and filter only even terms using lambda
c. Write a program to find the number of rows in a text file using Generator and yield.
d. Find Sum of Squares of 1 to n numbers using Generator Expressions.

13. **Python - Regular Expressions:** Let's take a password as a combination of alphanumeric characters along with special characters, and check whether the password is valid or not with the help of a few conditions.

Conditions for a valid password are:
- Should have at least one number.
- Should have at least one uppercase and one lowercase character.
- Should have at least one special symbol.
- ***********Should be between 6 to 20 characters long.

Input : Gvpce12#   Output :
Password is valid.
    Input : asd123
  Output : Invalid Password !!

14. **Programs on Packages:**
a) Create two packages one with name "Arithmetic" that contains all arithmetic related operations and another with name "interest" that contains operations that implement simple and compound interests

b) Create the directory name called "Cars" inside this directory create sub-directories "BMW", "Audi" and "Nissan" and inside these sub directories create the corresponding python files that maintain the corresponding Car models, prices, and Manufacturing years. Create "view.py" in "Cars" that takes a particular Car name and outputs the corresponding details.

15. **Programs on Numpy-1:**
a) Write a Python Program to demonstrate numpy arrays creation using array () function.
b) Write a python program to demonstrate use of ndim, shape, size, dtype.
c) Python program to demonstrate basic slicing, integer and Boolean indexing.
d) Write a python program to find min, max, sum, cumulative sum of array.

16. **Programs on Numpy-2:**
a) Create two single dimensional NumPy arrays, one is height, and another is weight, calculate BMI (weight/height**2) and keep all BMI values in another NumPy Array. Calculate mean, median, and standard deviation of BMI values.

b) Integrate the above height and weight arrays as single 2D NumPy array, and show slicing of elements, apply sum() and sort() operations on each row of the 2D NumPy array

17. **Programs on Pandas and Data Visualization:**

a) Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows:

     i.   Apply head () function to the pandas data frame
     ii.   Perform various data selection operations on Data Frame

b) Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib

18. Matplotlib -1

a. Write a Python program to draw a line using given axis values with suitable label in the x axis , y axis and a title.

b. Write a Python program to plot several lines with different format styles in one command using arrays.

c. Write a Python program to draw a line using given axis values taken from a text file, with suitable label in the x axis, y axis and a title.

19. Matplotlib-2

a. Write a Python programming to display a bar chart of the popularity of programming Languages.
Sample data:
Programming languages: Java, Python, PHP, JavaScript, C#, C++
Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7

b. Write a Python program to display a horizontal bar chart of the popularity of programming Languages.
Sample data:
Programming languages: Java, Python, PHP, JavaScript, C#, C++
Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7

20. Matplot-3

a. Write a Python program to create a bar plot of scores by group and gender. Use multiple X values on the same chart for men and women.
Sample Data:
Means (men) = (22, 30, 35, 35, 26)
Means (women) = (25, 32, 30, 35, 29)

b. Write a Python programming to create a pie chart with a title of the popularity of programming Languages.
Sample data:

Programming languages: Java, Python, PHP, JavaScript, C#, C++ Popularity:

22.2, 17.6, 8.8, 8, 7.7, 6.7

TEXT BOOKS:

1. A Krishna Mohan, T Murali Mohan & Karunakar, *Python with Machine Learning,* 1st Edition, S Chand Publications, 2019
2. Y. Daniel Liang, *Introduction to programming using Python,* 1st Edition, Pearson Publications, 2017.

REFERENCES:
1. Sheetal Taneja, *Python Programming A Modular Approach* ,1st Edition Pearson Publications, 2017.
2. Brett Slatkin (C), *Effective Python: 59 Specific Ways to Write Better Python,* I/C, 1st Edition Pearson Publications, 2015.
3. Ashok Namdev Kamathane and Amit Ashok Kamathane, *Programming and Problem Solving with Python*, 1st Edition, McGraw Hill Education (India) Private Limited, 2017.

WEB REFERENCES:

1. https://onlinecourses.nptel.ac.in/noc21_cs78/preview
2. https://www.coursera.org/learn/python?specialization=python#syllabus
3. https://www.coursera.org/learn/python-data?specialization=python#syllabus
4. https://www.coursera.org/learn/python-databases?specialization=python#syllabus