

Content	Reference	Chapter	# Periods
<b>Python Programming: An Introduction</b>  Structure of a Python program, understanding Python interpreter/Pythonshell, indentation. Atoms, identifiers and keywords, literals, Python strings, arithmetic operator, relational operator, logical or Boolean operator, bit wise operators.	1	1	2
<b>Variables and Functions</b>  Python standard libraries such as sys and math. Variables and assignment statements. Built-in functions such as input and print.	1	2.1 (19-27)	2
<b>Control Structures</b>  if conditional statement and for loop, While loop, break, continue, and pass statement, else statement. Infinite loop	1	3	3
<b>Functions</b>  Function definition and call, default parameter values, keyword arguments, assert statement	2  1	6 (172-190), excluding 6.3.1, 6.8  2.4 (39-40)	2
<b>Strings and Lists</b>  Strings-slicing, membership, and built-in functions on strings  Lists- list operations.	1  2	6.1-6.2,  10.1-10.2.2	2
<b>Mutable object</b>  Lists- built-in functions, list	2	10.2.3 – 10.2.14,	2

comprehension, passing list as arguments, copying list objects.		10.6 – 10.8	
Sets, tuples, and dictionary- associated operations and built-in functions.	2	14	3
<b>Testing and Debugging</b>  Determining test cases, use of python debugger tool- pydb for debugging	1	4  (to be done in the practical)	1
<b>Searching and Sorting</b>  Linear search, binary search, selection sort, insertion sort, and bubblesort	2	10.10 – 10.11	2
<b>Python 2D and 3D Graphics</b>  Visualization using graphical objects like point, line, histogram, sineand cosine curve, 3D objects	1	17.1- 17.2 (to be done in the practical)	2
<b>File Handling</b>  Reading and writing text and structured files.	2	13.1 – 13.2	3
<b>Errors and Exceptions</b>  Types of errors and exceptions, and exception handling	2	13.6 – 13.7	2
<b>Classes</b>  Notion of class, object, and method	2	7.1 – 7.2	2

### **Books**

1. Taneja, S. & Kumar, N., (2018), Python Programming- A Modular Approach. Pearson Education.
2. Liang, Y.D. (2017), Introduction to programming using Python. Pearson Education.

Course Teaching Learning Process



### List of Practical

(Use Python debugger tool-pydb and PythonTutor for debugging where ever required.)

1. Write a function that takes the lengths of three sides: side1, side2 and side3 of the triangle as the input from the user using input function and return the area and perimeter of the triangle as a tuple. Also, assert that sum of the length of any two sides is greater than the third side.

2. Consider a showroom of electronic products, where there are various salesmen. Each salesman is given a commission of 5%, depending on the sales made per month. In case the sale done is less than 50000, then the salesman is not given any commission. Write a function to calculate total sales of a salesman in a month, commission and remarks for the salesman. Sales done by each salesman per week is to be provided as input. Use tuples / list to store data of salesmen.

Assign remarks according to the following criteria:

Excellent: Sales  $\geq 80000$

Good: Sales  $\geq 60000$  and  $< 80000$

Average: Sales  $\geq 40000$  and  $< 60000$

Work Hard: Sales  $< 40000$

3. Write a Python function to find the nth term of Fibonacci sequence and its factorial. Return the result as a list.

4. Write a function that takes a number ( $\geq 10$ ) as an input and return the digits of the number as a set.

5. Write a function that finds the sum of the  $n$  terms of the following series. Import the factorial function created in question 4.

$$1 - x^2/2! + x^4/4! - x^6/6! + \dots x^n/n!$$

6. Consider a tuple  $t1=\{1,2,5,7,9,2,4,6,8,10\}$ . Write a program to perform following operations:

- a) Print another tuple whose values are even numbers in the given tuple.
- b) Concatenate a tuple  $t2=\{11,13,15\}$  with  $t1$ .
- c) Return maximum and minimum value from this tuple.

7. Write a menu driven program to perform the following on strings:

- a) Find the length of string.
- b) Return maximum of three strings.
- c) Accept a string and replace all vowels with “#”
- d) Find number of words in the given string.
- e) Check whether the string is a palindrome or not.

8. Write a Python program to perform the following using list:

- a) Check if all elements in list are numbers or not.
- b) If it is a numeric list, then count number of odd values in it.
- c) If list contains all Strings, then display largest String in the list.
- d) Display list in reverse form.
- e) Find a specified element in list.

f) Remove the specified element from the list.

g) Sort the list in descending order.

h) accept 2 lists and find the common members in them.

9. Use dictionary to store marks of the students in 4 subjects. Write a function to find the name of the student securing highest percentage. (Hint: Names of students are unique).

10. Write a function that takes a sentence as input from the user and calculates the frequency of each letter. Use a variable of dictionary type to maintain the count.

11. Write a menu-driven program to accept a list of student names and perform the following

a. search an element using linear search/ binary search.

b. Sort the elements using bubble sort/ insertion sort/ selection sort.

12. Write a program that makes use of a function to accept a list of n integers and displays a histogram.

13. Write a program that makes use of a function to display sine, cosine, polynomial and exponential curves.

14. Write a function that reads a file file1 and copies only alternative lines to another file

file2. Alternative lines copied should be the odd numbered lines. Use Exception.

15. Define a class Student to store his/ her name and marks in three subjects. Use a class variable to store the maximum average marks of the class. Use

constructor and destructor to initialize and destroy the objects.