Course Code	Course name	L	T	P	C
CSEG1135	Python Programming Lab	0	0	4	2
Total Units to be Covered: 10		Total Contact Hours: 60			
Prerequisite(s):		Syllabus version: 1.0			

Course Objectives

1. Develop a strong foundation in Python programming language, including syntax, data types, control structures, and functions, enabling students to write efficient and reliable code.
2. Understand and apply object-oriented programming (OOP) principles in Python to design and build modular, reusable, and maintainable software solutions.
3. Gain proficiency in utilizing Python libraries and modules for tasks such as data manipulation, web scraping, data analysis, and visualization, empowering students to work with real-world data effectively.
4. Explore advanced topics in Python, including concurrency, file I/O, exception handling and equipping students with the skills to build robust and scalable applications.

Course Outcomes

On completion of this course, the students will be able to:

CO1. Demonstrate proficiency in Python programming by writing code that adheres to Python syntax, utilizes appropriate data types, and implements control structures effectively.

CO2. Apply Python collections, such as lists, tuples, dictionaries, and sets, along with the design and implementation of reusable functions, to solve complex programming problems, demonstrating proficiency in data organization, manipulation, and modular code design.

CO3. Implement advanced Python features and techniques, such as modules and packages, file handling, exception handling and regular expression to create robust and reliable applications.

CO4. Apply object-oriented programming (OOP) concepts in Python to design and develop modular software solutions that promote code reusability and maintainability.

CO5. Utilize Python libraries and modules for data manipulation, analysis, and visualization, demonstrating the ability to work with real-world data sets and extract meaningful insights.

CO-PO Mapping

Program Outcomes / Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	-	-	-	-	2	-	-	-	-	-	-	-	2	1	-
CO 2	-	-	-	-	2	-	-	-	-	-	-	-	2	1	-
CO 3	-	-	-	-	2	-	-	-	-	-	-	-	2	1	-
CO 4	-	-	-	-	2	-	-	-	-	-	-	-	2	1	-
CO 5	-	-	-	-	2	-	-	-	-	-	-	-	2	2	-
Average	-	-	-	-	2	-	-	-	-	-	-	-	2	1.2	-

1 – Weakly Mapped (Low) **2 – Moderately Mapped (Medium)**

3 – Strongly Mapped (High) **"_" means there is no correlation**

List of Experiments

Experiment 1: Python Installation and starting with python

1. Install Python and understand difference between scripting and interactive modes in IDLE.
2. Write Python programs to print strings in the given manner:
   a) Hello Everyone !!!
   b) Hello
      World
   c) Hello
         World
   d) ' Rohit' s date of birth is 12\05\1999'
3 Declare a string variable called x and assign it the value "Hello".

Print out the value of x

4 Take different data types and print values using print function.

5 Take two variable a and b. Assign your first name and last name. Print your Name after adding your First name and Last name together.

6 Declare three variables, consisting of your first name, your last name and Nickname. Write a program that prints out your first name, then your nickname in parenthesis and then your last name.

Example output : George (woody) Washington.

7 Declare and assign values to suitable variables and print in the following way :

NAME : NIKUNJ BANSAL

SAP ID : 500069944

DATE OF BIRTH : 13 Oct 1999

ADDRESS : UPES

Bidholi Campus

Pincode : 248007

Programme : AI & ML

Semester : 2

Experiment 2: Use of input statements, operators

1. Declare these variables (x, y and z) as integers. Assign a value of 9 to x, Assign a value of 7 to y, perform addition, multiplication, division and subtraction on these two variables and Print out the result.
2. Write a Program where the radius is taken as input to compute the area of a circle.
3. Write a Python program to solve (x+y)*(x+y)
   Test data : x = 4 , y = 3
   Expected output: 49
4. Write a program to compute the length of the hypotenuse (c) of a right triangle using Pythagoras theorem.
5. Write a program to find simple interest.
6. Write a program to find area of triangle when length of sides are given.
7. Write a program to convert given seconds into hours, minutes and remaining seconds.

8. Write a program to swap two numbers without taking additional variable.
9. Write a program to find sum of first n natural numbers.
10. Write a program to print truth table for bitwise operators(& , | and ^ operators)
11. Write a program to find left shift and right shift values of a given number.
12. Using membership operator find whether a given number is in sequence (10,20,56,78,89)
13. Using membership operator find whether a given character is in a string.

Experiment 3 : Conditional Statements

1. Check whether given number is divisible by 3 and 5 both.
2. Check whether a given number is multiple of five or not.
3. Find the greatest among two numbers. If numbers are equal than print "numbers are equal".
4. Find the greatest among three numbers assuming no two values are same.
5. Check whether the quadratic equation has real roots or imaginary roots. Display the roots.
6. Find whether a given year is a leap year or not.
7. Write a program which takes any date as input and display next date of the calendar

   e.g.

   I/P: day=20 month=9 year=2005

   O/P: day=21 month=9 year 2005
8. Print the grade sheet of a student for the given range of cgpa. Scan marks of five subjects and calculate the percentage.

CGPA=percentage/10

CGPA range:

0 to 3.4 -> F

3.5 to 5.0->C+

5.1 to 6->B

6.1 to 7-> B+

7.1 to 8-> A

8.1 to 9->A+

9.1 to 10-> O (Outstanding)

Sample Gradesheet

Name: Rohit Sharma

Roll Number: R17234512 SAPID: 50005673

Sem: 1 Course: B.Tech. CSE AI&ML

Subject name: Marks

PDS: 70

Python: 80

Chemistry: 90

English: 60

Physics: 50

Percentage: 70%

CGPA:7.0

Grade:

Experiment 4: Loops

1. Find a factorial of given number.
2. Find whether the given number is Armstrong number.
3. Print Fibonacci series up to given term.
4. Write a program to find if given number is prime number or not.
5. Check whether given number is palindrome or not.
6. Write a program to print sum of digits.
7. Count and print all numbers divisible by 5 or 7 between 1 to 100.
8. Convert all lower cases to upper case in a string.
9. Print all prime numbers between 1 and 100.
10. Print the table for a given number:

    5 * 1 = 5

    5 * 2 = 10……..

Experiment 5: String and Sets

1. Write a program to count and display the number of capital letters in a given string.
2. Count total number of vowels in a given string.
3. Input a sentence and print words in separate lines.
4. WAP to enter a string and a substring. You have to print the number of times that the substring occurs in the given string. String traversal will take place from left to right, not from right to left.

   Sample Input

   ABCDCDC

   CDC

   Sample Output

   2
5. Given a string containing both upper and lower case alphabets. Write a Python program to count the number of occurrences of each alphabet (case insensitive) and display the same.

   Sample Input

   ABaBCbGc

   Sample Output

   2A

   3B

   2C

   1G
6. Program to count number of unique words in a given sentence using sets.
7. Create 2 sets s1 and s2 of n fruits each by taking input from user and find:
   a) Fruits which are in both sets s1 and s2
   b) Fruits only in s1 but not in s2
   c) Count of all fruits from s1 and s2
8. Take two sets and apply various set operations on them :

   S1 = {Red ,yellow, orange , blue }

   S2 = {violet, blue , purple}

Experiment 6: Lists, tuples, dictionary

1. Scan n values in range 0-3 and print the number of times each value has occurred.
2. Create a tuple to store n numeric values and find average of all values.
3. WAP to input a list of scores for N students in a list data type. Find the score of the runner-up and print the output.

   Sample Input

   N = 5

   Scores= 2 3 6 6 5

   Sample output

   5

   Note: Given list is [2, 3, 6, 6, 5]. The maximum score is 6, second maximum is 5. Hence, we print 5 as the runner-up score.
4. Create a dictionary of n persons where key is name and value is city.

   a) Display all names

   b) Display all city names

   c) Display student name and city of all students.

   d) Count number of students in each city.
5. Store details of n movies in a dictionary by taking input from the user. Each movie must store details like name, year, director name, production cost, collection made (earning) & perform the following :-

   a) print all movie details

   b) display name of movies released before 2015

   c) print movies that made a profit.

   d) print movies directed by a particular director.

Experiment 7: Functions

1. Write a Python function to find the maximum and minimum numbers from a sequence of numbers. (Note: Do not use built-in functions.)
2. Write a Python function that takes a positive integer and returns the sum of the cube of all the positive integers smaller than the specified number.
3. Write a Python function to print 1 to n using recursion. (Note: Do not use loop)

4. Write a recursive function to print Fibonacci series upto n terms.
5. Write a lambda function to find volume of cone.
6. Write a lambda function which gives tuple of max and min from a list.
   Sample input: [10, 6, 8, 90, 12, 56]
   Sample output: (90,6)
7. Write functions to explain mentioned concepts:
   a. Keyword argument
   b. Default argument
   c. Variable length argument

Experiment 8: File Handling and Exception Handling

1. Add few names, one name in each row, in “name.txt file”.
   a. Count no of names
   b. Count all names starting with vowel
   c. Find longest name
2. Store integers in a file.
   a. Find the max number
   b. Find average of all numbers
   c. Count number of numbers greater than 100
3. Assume a file city.txt with details of 5 cities in given format (cityname population(in lakhs) area(in sq KM)):
   Example:
   Dehradun 5.78 308.20
   Delhi 190 1484
   ……
   Open file city.txt and read to:
   a. Display details of all cities
   b. Display city names with population more than 10Lakhs
   c. Display sum of areas of all cities
4. Input two values from user where the first line contains N, the number of test cases. The next N lines contain the space separated values of a and b. Perform integer division and print a/b. Handle exception in case of ZeroDivisionError or ValueError.
   Sample input

1 0

2 $

3 1

Sample Output :

Error Code: integer division or modulo by zero

Error Code: invalid literal for int() with base 10: '$' 3

5. Create multiple suitable exceptions for a file handling program.

Experiment 9: Classes and objects

1.Create a class of student (name, sap id, marks[phy,chem,maths]). Create 3 objects by taking inputs from the user and display details of all students.

2. Add constructor in the above class to initialize student details of n students and implement following methods:

a) Display() student details

b) Find Marks_percentage() of each student

c) Display result() [Note: if marks in each subject >40% than Pass else Fail]

Write a Function to find average of the class.

3. Create programs to implement different types of inheritances.

4. Create a class to implement method Overriding.

5. Create a class for operator overloading which adds two Point Objects where Point has x & y values

e.g. if

P1(x=10,y=20)

P2(x=12,y=15)

P3=P1+P2 => P3(x=22,y=35)

Experiment 10: Data Analysis and Visualization

1. Create numpy array to find sum of all elements in an array.
2. Create numpy array of (3,3) dimension. Now find sum of all rows & columns individually. Also find 2nd maximum element in the array.
3. Perform Matrix multiplication of any 2 n*n matrices.
4. Write a Pandas program to get the powers of an array values element-wise.

Note: First array elements raised to powers from second array

Sample data: {'X':[78,85,96,80,86], 'Y':[84,94,89,83,86],'Z':[86,97,96,72,83]}

Expected Output:

X Y Z

0 78 84 86

1 85 94 97

2 96 89 96

3 80 83 72

4 86 86 83

5. Write a Pandas program to get the first 3 rows of a given DataFrame.

Sample Python dictionary data and list labels:

exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],

'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],

'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],

'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']}

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

Expected Output:

First three rows of the data frame:

attempts name qualify score

a 1 Anastasia yes 12.5

b 3 Dima no 9.0

c 2 Katherine yes 16.5

6. Write a Pandas program to find and replace the missing values in a given DataFrame which do not have any valuable information.

7. Create a program to demonstrate different visual forms using Matplotlib.

Total Lab hours 60

Textbooks

1. Reema Thareja, "Python Programming", Oxford University Press, 2017.

2. Mark Lutz, "Learning Python", 5th ed., O'reilly publication, 2013.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination

Examination Scheme: Continuous Assessment

Components	Quiz & Viva	Performance & Lab Report
Weightage (%)	50	50