# PRASAD V. POTLURI SIDDHARTHA INSTITUTE OF TECHNOLOGY

(Autonomous)
Kanuru, Vijayawada-520007

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (DATA SCIENCE)

## II B. Tech – I Semester CSE (DATA SCIENCE)

## PYTHON PROGRAMMING LAB

| | | | | | |
|---|---|---|---|---|---|
| **Course Code** | 23SO8355 | **Year** | II | **Semester** | I |
| **Course Category** | Skill Enhancement Course | **Branch** | CSE (Data Science) | **Course Type** | Practical |
| **Credits** | 2 | **L – T – P** | 0-1-2 | **Prerequisites** | Nil |
| **Continuous Internal Evaluation** | 30 | **Semester End Exam** | 70 | **Total Marks** | 100 |

| **Course Outcomes** | | |
|---|---|---|
| Upon successful completion of the course, the student will be able to: | | |
| **CO1** | Demonstrate experimental procedures through oral communication and submit comprehensive documentation reports. | L2 |
| **CO2** | Apply Python programming constructs for solving problems | L3 |
| **CO3** | Implement programs as an individual on different IDEs/ online platforms. | L3 |
| **CO4** | Analyze outputs using given constraints/test cases. | L4 |

**Contribution of Course Outcomes towards achievement of Program Outcomes & Strength of correlations (3:High, 2: Moderate, 1:Low)**

| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | | | | | | | | | 2 | | | | |
| **CO2** | 3 | | | | | | | | | | | | | |
| **CO3** | 3 | | | | 3 | | | | | | | | | |
| **CO4** | | 3 | | | | | | | | | | 3 | | |

# Syllabus

| Unit No. | CONTENTS | Mapped CO |
|---|---|---|
| I | **UNTI-I:**<br>History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook.<br>**Parts of Python Programming Language**: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly Typed Language.<br>**Control Flow Statements**: if statement, if-else statement, if...elif…else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.<br><br>**Sample Experiments**<br>1. Write a Program to print the student details using Escape sequence characters.(Example:\n,\t,\").<br>2. The total number of students in a class are 45 out of which 25 are boys. If 80% of the total students secured grade 'A' out of which 16 are boys, then Develop a Program to calculate the total number of girls getting grade 'A'.<br>3. Develop a Program to calculate the sum of the first and the last digit of a 56743<br>4. Write a program for calculating the bill amount for an item with the following scenarios<br>• The quantity of item sold, and price of the item must read from the user and calculate the bill<br>• After that there is a 10% discount on bill amount<br>• There is a tax amount of 12%<br>• Find the total bill after availing the discount and applying the tax<br>5. Implement a program to calculate in how many days a work will be completed by three persons A, B and C together. A, B, C take x days, y days and z days respectively to do the job alone. The formula to calculate the number of days if they work together is xyz/(xy + yz + xz) days where x, y, and z are given as input to the program.<br>6. Implement a program to read two complex numbers and perform addition ,subtraction<br>7. Develop a program to demonstrate evolution of following arithmetic expressions?<br>• Consider b=4, c=8, d=2,e=4,f=2<br>• a=b+c/d+e*f<br>• a=(b+c)/d+e*f<br>• a=b+c/((d+e)*f)<br>8. Write a Python program that takes two lists as input and concatenates them using the "+" operator.<br>9. Write a program to enter the marks of a student in four subjects. Then | **CO1, C02, CO3, CO4, CO5** |

calculate the total and aggregate, and display the grade obtained by the student. If the student scores an aggregate greater than 75%, then the grade is Distinction. If aggregate is >=60 and <75, then the grade is First Division. If aggregate is >=50 and <60, then the grade is Second division. If aggregate is >=40 and <50, then the grade is third division. Else the grade is Fail.

10. Write a program to calculate roots of a quadratic equation. The programmer has to identify whether the roots are real, equal or imaginary
11. A company decides to give bonus to all its employees on Diwali. A 5% bonus on salary is given to the male workers and 10% bonus on salary to the female workers. Write a program to enter the salary and gender of the employee. If the salary of the employee is less than Rs. 10,000 then the employee gets an extra 2% bonus on salary. Calculate the bonus that must be given to the employee and display the salary that the employee will get.
12. Demonstrate a program to print the sum of the series $1/1^2+ 1/2^2+1/3^2 +\ldots\ldots. +1/n^2$. where n is taken from the user.
13. Write a program to implement the below scenarios
    - Sum of cubes of numbers from 1 to n using range ().
    - Display the numbers in descending order using range ().
    - Sum of squares of even numbers from 1 to n using range ().
    - Display all leap years from 2000 – 2200 using range ().
14. Write a program to print the below patterns:

| | | | | |
|---|---|---|---|---|
| 1<br>2 3<br>4 5 6<br>7 8 9 10<br>11 12 13 14 15 | 1<br>2 1<br>3 2 1<br>4 3 2 1<br>5 4 3 2 1 | 5 4 3 2 1<br>4 3 2 1<br>3 2 1<br>2 1<br>1 | *<br>* *<br>* * *<br>* * * *<br>* * * * * | 1<br>1 2<br>1 2 3<br>1 2 3 4 |

15. Create a library with functions to input the values with exception handling in Python
16. Write a Python program input and add two integers only and handle the exceptions.

**II**

**UNIT-II:**

**Functions**: Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

**CO1, C02, CO3, CO4, CO5**

**Strings**: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings. Regular expression: Matching the patterns, Search and replace.

**Sample Experiments**

1. Write a program to find sum of all odd numbers between 1 to n using functions.
2. Write a program to demonstrate default arguments with keyword arguments, to display name, age and salary of an employee. Where course (B. Tech) is passed as default argument.
3. Write a program to find the sum of first 10 natural numbers using lambda or anonymous function using range () function.
4. Demonstrate a program to convert time into minutes using functions
5. Write a program to calculate simple interest. Suppose the customer is a senior citizen. He is being offered 12% rate of interest (ROI).For all other customers, the ROI is 10%.
6. Python Program to check if two numbers are amicable numbers or not. Two different numbers are called amicable numbers if the sum of the proper divisors of each is equal to the other number.
7. Demonstrate a program to sum the series 1/1!+4/2!+27/3+…n using functions
8. Write a program to generate the following pattern using default arguments. Consider four types in calling the function.
    - Do not pass arguments
    - Pass only the character as argument
    - Pass character and no. of rows as argument
    - Pass character, no. of rows and columns as arguments.
9. Write a program using recursive functions:
    - Counting the no. of times, a recursive function is called
    - Power of a number
    - GCD of two given numbers
    - Print the Fibonacci series
10. Write a python program without using the built in functions to find the length of the string, reverse the string.
11. Write a python program to arrange string characters such that lowercase letters should come first.
12. Write a program that uses regular expressions to validate dates entered by users. The program should check that the date is in a valid format, such as MM/DD/YYYY and that the month, day, and year values are within a valid range.
13. Write a program to validate a password using regular expressions using the following rules
    - At least 8 characters long
    - Contains at least one uppercase letter
    - Contains at least one lowercase letter
    - Contains at least one digit
14. Write a program to remove all non-alphanumeric characters from a given string using regular expressions.

| | | |
|---|---|---|
| III | **UNIT-III:**<br><br>**Lists**: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.<br><br>**Dictionaries**: Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.<br><br>**Tuples and Sets**: Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset.<br><br>**Sample Experiments**<br>1. Write a python program to add each element of list x with list y using nested loops.<br>2. Write a python program to print index at which a particular value exists. If the value exists at multiple locations in the list, then print all the indices. Also, count the number of times that value is repeated in the list.<br>3. Write a python program applying all the list methods ('append', 'clear', 'copy', 'count', 'extend', 'index', 'insert', 'pop', 'remove', 'reverse', 'sort') on the given list.<br>List = [100,'a','b',102,2.3,4.5]<br>4. Write a python program to add each element of x list with each element of y list.<br>• Using loops<br>• Using list comprehension<br>5. Write a program using lambda and below functions to perform tasks<br>▪ Using filter () to filter out even numbers from a list.<br>▪ Find squares of elements in a list using map ().<br>▪ Product of elements of a list using reduce() function<br>6. Write a python program to do the below matrix operations<br>• Addition<br>• Subtraction<br>• Multiplication<br>7. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.<br>8. Write a program to count the number of vowels in a string (No control flow allowed).<br>9. Write a program to check if a given key exists in a dictionary or not.<br>10. Write a program to add a new key-value pair to an existing dictionary.<br>11. Write a program to sum all the items in a given dictionary.<br>12. Write a program that reads string from user. Your program should create a dictionary having key as word length and value is count of words of that length. For example, if user enters 'A fat cat is on the mat'. The content of dictionary should be {1:1, 3:4, 2:2} | **CO1, C02, CO3, CO4, CO5** |

| | | |
|---|---|---|
| **IV** | **UNIT-IV:**<br><br>**Files**: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.<br><br>**Object-Oriented Programming**: Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.<br><br>**Sample Experiments**<br><br>1. 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.<br>2. Python program to print each line of a file in reverse order.<br>3. Python program to compute the number of characters, words and lines in a file.<br>4. Write a function lines_count() that reads lines from a text file named 'zen.txt' and displays the lines that begin with any vowel. Assume the file contains the following text and already exists on the computer's disk:<br>Beautiful is better than ugly.<br>Explicit is better than implicit.<br>Simple is better than complex.<br>Complex is better than complicated.<br>The lines_count() function should display the output as:<br>Explicit is better than implicit.<br>5. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.<br>6. Create a Parallelepiped child class inheriting from the Rectangle class and with a height attribute and anotherVolume() method to calculate the volume of the Parallelepiped.<br>7. Write a Python program to create a class representing a stack data structure. Include methods for pushing, popping and displaying elements<br>8. Write a Python program to create a class representing a queue data structure. Include methods for enqueueing and dequeueing elements<br>9. Write the complete code for BankAccount class based on the description given below:<br>• Create a Python class called BankAccount which represents a bank account, having as attributes: accountNumber(numeric type), Name(name of the account owner as string type), balance.<br>• Create a constructor with parameters: accountNumber, name, balance<br>• Create a Deposit () method which manages the deposit actions. | **CO1, C02, CO3, CO4, CO5** |

| | | |
|---|---|---|
| | • Create a Withdrawal () method which manages withdrawals actions.<br>• Create a bankFees() method to apply the bank fees with a percentage of 5% of the balance account.<br>• Create a display() method to display account details. | |
| **V** | **UNIT-V:**<br><br>**Introduction to Data Science**: Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.<br><br>**Sample Experiment**<br><br>1. Python program to check whether a JSON string contains complex object or not.<br>2. Python Program to demonstrate NumPy arrays creation using array () function.<br>3. Python program to demonstrate use of ndim, shape, size, dtype.<br>4. Python program to demonstrate basic slicing, integer and Boolean indexing.<br>5. Python program to find min, max, sum, cumulative sum of array<br>6. 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:<br>a) Apply head () function to the pandas data frame<br>b) Perform various data selection operations on Data Frame<br>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 | **CO1, C02, CO3, CO4, CO5** |

| **Learning Resources** |
|---|
| **Text Books** |
| 1. Introduction to Python Programming , Gowrishankar S, Veena A , CRC Press. |
| **Reference Books** |
| 1. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, 2024, Pearson.<br>2. Introduction to Programming Using Python, Y. Daniel Liang, Pearson. |
| **E-Resources & other digital material** |
| 1. https://www.coursera.org/learn/python-for-applied-data-science-ai<br>2. https://www.coursera.org/learn/python?specialization=python#syllabus |