THADOMAL SHAHANI ENGINEERING COLLEGE

List of Experiments

Subject: Python

Branch: Computer   **Semester: IV**

Experiment No 1.

1.   Write a Python program to print the following string in a specific format
         Twinkle, twinkle, little star,
                "How I wonder what you are! "
                        Up above the world so high,
                        Like a diamond in the sky.
         Twinkle, ' twinkle ', little star,
                How I wonder what you are
         Using only one print() function.

2.   Program to show output formatting take two values and display them using single
        print function using
        •   str.format()
        •   % operator

3.   Write a Python program that takes two numbers as input and then allows the user to
        choose an operation (addition, subtraction, multiplication, or division) to perform.
        The program should then display the result of the operation.

4.   Program to print all armstrong number in range 1 to 1000.

5.   Program to find fibonacci series of n terms

6.   Program on pattern
        A   *****   1   *
        B B   ****   121   * *
        C C C   ***   12321   * * *
        D D D D   **   1234321   * * * *
        E E E E E   *   123454321   * * * * *

Experiment No 2.

1. Python program to
   ● Read an array and display
   ● Append a new item to the end of the array.
   ● To reverse the order of the items in the array (slice operator)
   ● Get the length in bytes of one array item
   ● To append items from another array
   ● Remove a specified item using the index from an array

2. Python program to remove prime numbers from an array.

   Sample input arr[] = {3,4,6,9,13,14,16,17}
   Output arr[] = {4,6,9,13,16}

3. Python program to change all occurrences of a first character of a string to @ except for first occurrence.
   Sample String : 'apple a day'
   ● Expected Result : 'apple @ d@y'

4. Python Program
   ● to sort group of strings into alphabetical order
   ● to check whether entered string is palindrome or not
   ● to count the number of vowels (a, e, i, o, u) in a given string.

Experiment - 3

LOI

1. Write a Python function to check whether a number is perfect or not.
   (Note : The first perfect number is 6, because 1, 2, and 3 are its proper positive divisors, and 1 + 2 + 3 = 6. Equivalently, the number 6 is equal to half the sum of all its positive divisors: (1 + 2 + 3 + 6) / 2 = 6. The next perfect number is 28 = 1 + 2 + 4 + 7 + 14. This is followed by the perfect numbers 496 and 8128.)

OR

1. Write a Python function to check whether a string is a pangram or not.
   (Note : Pangrams are words or sentences containing every letter of the alphabet at least once.
   For example : "The quick brown fox jumps over the lazy dog")

2. Python menu driven program to develop simple calculator using variable length argument

3. Python program to calculate square and cube of a number and use two decorators, one to increase result by 4 and another to multiply result by 2.

4. Write menu driven python program that accept list of numbers and performs following operation on list written in another module

   - Summation of all elements
   - Product of all elements
   - Summation of elements at even indices
   - add elements in the list

1.   Write a menu driven program to demonstrate use of list in python
     ● Put even and odd elements in two different list
     ● Merge and sort two list
     ● Update the first element with a value X
     ● Print middle element of list
2.   Write a menu driven program to demonstrate use of tuple in python
     ● Add and show details i.e roll no, name and marks of three subjects of N students in a list of tuple
     ● Display details of a student whose name is X
3.   Write a menu driven program to demonstrate use of set in python
     ● Read two sets A and B from user and display set A and B
     ● Perform intersection A ∩ B of two sets A and B
     ● Perform union A U B of two sets A and B
     ● Perform set difference A - B of two sets A and B
     ● Perform symmetric difference A ^ B of two sets A and B
4.   Write a program to demonstrate use of dictionary in python
     ● Read a dictionary from the user and display.
     ● To sort a dictionary by key
     ● Concatenate two Python dictionaries into a new one

Experiment 5

1. Python program to create and **list of employees** using **Employee class**.
    Program should also print **total number of employees**.
    Employee class should have
    - *empcount* (class variable)
    - *id and name* (instance variable)
    - **constructor** to set *id*,
    - *set_name( ), get_name ( ), get_id( )* methods (instance method)
    - *set_emp_count ( )* (class method)
2. Python program to demonstrate **Multiple Inheritance**

3. Python program to overload greater than (>) operator to make it act on user defined class objects
4. Python program to demonstrate concept of **Interfaces.**
    Program contains **Printer** interface and its subclasses to send text to any printer.

Experiment 6   LO1, LO2, LO3
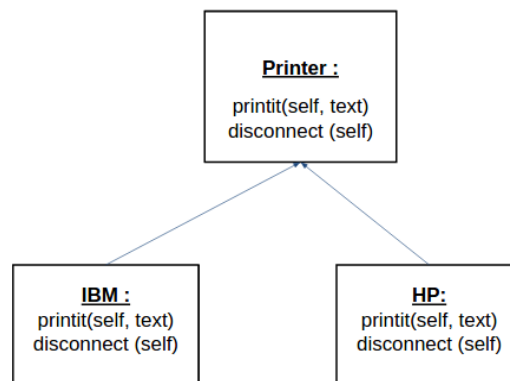
1.   Python program
   - To create class **Student with rno, name, marks** as instance variable and **constructor** to initialize these instance variables.
   - Instantiate **n** instances of classes and save details in list.
   - Create an **user defined exception** class **Fail** to raise an exception if marks is less than 40.
   - Display details of students and also raise exceptions for marks less than 40.
2.   Python program to
   - create a file
   - count no. of lines , words and characters in a file.
   - write content of a file in a new file and read that new file.
3.   Python program to
   - create a **class Customer** with **id, name, mobile number** as instance variable and **constructor** to initialize these instance variables.
   - Instantiate **n** instances of classes
   - Save details of all customer in a **file** and read back from that file.

Experiment 7   LO1, LO2

1.   Python program to demonstrate use of packages.
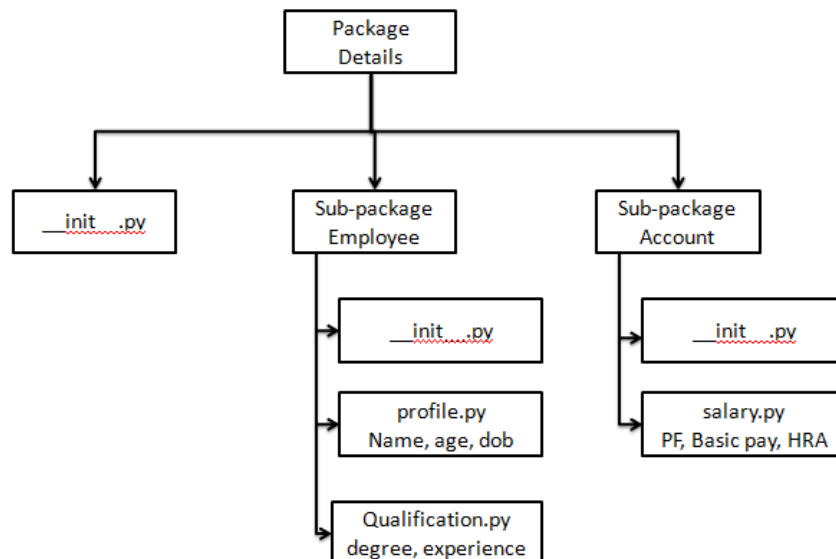
Create a file to import all the packages , calculate the salary and display all the details of the Employee

Salary = Basic +H.R.A – P.F.

Experiment 8

LO1, LO3

1.   Python program to demonstrate use of regular expression
   - Create string with name of cities in india separated by spaces.
   - Find all cities ending with "ai"
   - Find all cities starting with "Mu" or "Ma"

   print name of cities with 'u' as second letter and 'a' as second last letter

2.   Python program to demonstrate use of regular expression
   - Create a **phone list** using file **(surname   name   number)**

   Find all the entries of phone book with surname as **"Rao"** and first name starting with **'J' or 'K'**

3.   Python program to
   - create directories using mkdir( ) and makedirs ( )
   - remove directories using rmdir( ) and removedirs( )

   change current directory

Experiment 9

LO1, LO3

1.   Python program to demonstrate use of Tkinter interface module.
   Create a GUI based application using widgets Entry, Label, Text, Button, RadioButton, CheckButton, ListBox, Menu, Spinbox **(any five)**.

   Save the details in a file and read back from file on python prompt.

Experiment 10

LO1, LO5

1.   Python program to demonstrate MYSQL database connectivity with python.
   Create a GUI based application using widgets Entry, Label, Text, Button, RadioButton, CheckButton, ListBox, Menu, Spinbox **(any five)**.

   Save the details in a database and read back from file on python prompt.

Experiment   11   LO1, LO3

1.   Creating web application using Django web framework
        ● Installing Django
        ● Creating project
        ● Creating App and Views
        ● Creating and activating model
        ● Admin interface - Modify database from admin interface

Experiment   12   LO1, LO2

1.   Write a pandas program to:
        i) add, subtract, multiple and divide two pandas series
        ii) compare the elements of the two Pandas Series.
        iii) convert a dictionary to a Pandas series.
        iv) convert a NumPy array to a Pandas series.
2.   Write a program to read csv file in a dataframe, replace missing values with any value,
        drop the row if all values are missing or contain null values.
3.   Write a program to demonstrate merging of Frames:
        i) on the basis of id
        ii) using how

Assignment 1 - Data Structures

1.   Python program to implement operations of linked list
    1. Display list
    2. Insert at beginning
    3. Insert at End
    4. Insert at specified postion
    5. Delete from beginning
    6. Delete from end
    7. Delete at specified postion
    8. Delete a particular element
    9. Search an element
    10. Replace element at specified index
    11. Forward traversal
    12. Reverse traversal
    13. Insert after an element
2.   Python program to implement **Stack** data structure
      create class Stack with following functions.
    - Push an element
    - Pop an element
    - Top of Stack
    - Search an element
    - Display stack

      create instance and perform all operations.
3.   Python program to implement **Queue** data structure
      create class Queue with following functions.
    - Insert an element
    - Remove an element
    - Search an element
    - Display queue
4.   Python program to use deque class from collections with following functions.
    - Add element at Front
    - Remove element from Front
    - Add element at Rear
    - Remove element from Rear
    - Search for an element

1. Python program to create a 8x8 matrix and fill it with a checkerboard pattern (slice operator)

```
Checkerboard pattern:
[[0 1 0 1 0 1 0 1]
 [1 0 1 0 1 0 1 0]
 [0 1 0 1 0 1 0 1]
 [1 0 1 0 1 0 1 0]
 [0 1 0 1 0 1 0 1]
 [1 0 1 0 1 0 1 0]
 [0 1 0 1 0 1 0 1]
 [1 0 1 0 1 0 1 0]]
```

2. Menu driven code for numpy array
   - create using array() and arange()
   - sum of array
   - sort array

   compare two arrays

3. Python program
   - To read a two matrices from user
   - Perform matrix multiplication
   - Display diagonal Elements

   Check whether its a square matrix

4. Write a program, which can run two-thread simultaneously. One thread will print odd numbers and another thread will print even number.

5. Write a Python program with multiple threads, where each thread prints a message after a 3-second delay. Use the join() method to ensure the main program waits for all threads to complete before printing "All threads finished."