**Program List of Python Sem2 2020 – 2021**

1. Write first python program to print your introduction like name, address, phone number, email id etc.
2. WAP to declare multiple variable with all different types of value and check data type of all declare variable using type() method.
3. WAP to take basic details of employee like name, phone number, email id, salary, designation, address, blood group and birth date through keyboard and print it in proper format.
4. Write a menu driven program which show use of all arithmetic operators and print its result. Take input form key board.
5. WAP to find power and square root of given number without using inbuilt function. (note : find power using \*\* operator and square root using 0.5 value)
6. WAP to find simple interest for given principal amount. Take input of principal amount, rate of interest and year from user. (Intrest=(PRN)/100)
7. WAP menu driven program to find area of triangle, circle, rectangle, square and cylinder.
8. Write a menu driven program to convert kilometer to miles, Celsius to Fahrenheit, meter to centimeter, acer to square meter.
9. WAP to solve quadratic equation (ax2 + bx + c).
10. WAP to swap value of two variableswith all five possibilities
    1. Using temp variable
    2. Using + and – Operator
    3. Using // and \* Operator
    4. Using ^ (XOR) operator
    5. By using concept of different value to multiple variable allocation concept
11. WAP to check entered number is positive, negative, zero, odd or even using if else statement.
12. WAP to find maximum (largest) number among three numbers. (also try for 5 numbers)
13. WAP to print multiplication table in proper format for entered number using for loop as well as while loop.
14. WAP to find number is prime or not.
15. WAP to print list of prime number from given interval.
16. WAP to find factorial of given number.
17. WAP to print Fibonacci series/ sequence.
18. WAP to find whether inputted number is Armstrong or not.
19. WAP to print list of Armstrong number from given interval.
20. WAP to find sum of natural numbers using for loop as well as while loop.
21. WAP which shows use of datetime package of python.
22. Write a menu driven program to create simple calculator using user defined function.
23. WAP to find length of string without using inbuilt function len() function.
24. WAP to traverse string using for loop.
25. WAP which shows use slicing on string and any other data structure of python.
26. Write a menu driven program which shows the use of string inbuilt function and its operation like concatenation, repletion and slicing.
27. WAP which show how string is immutable by passing string as argument in user defined function.
28. Write a program to perform all bitwise operation using user defined function.
29. WAP to create list and perform operation like searching element, adding element, update element, removing element, traverse list in both direction left to right and right to left by passing list as argument in user defined function.
30. WAP to program to find binary of inputted number and store it into list and print it.
31. WAP program to covert list of word form given sentence by using split() method and also find position of entered word in list if it present in list else print appropriate message.
32. WAP to create list of words and convert only those words into upper case which start with vowels (a,e,I,o,u) other remain as it is.
33. WAP to create list of cube and square for upto given number. E.g. if user enter 10 than store cube and square to 1 to 10 into list.
34. WAP to program to create tuple and perform operation like searching, find length, slicing, also change first and last or entered index element by using concept of slicing and concatenation.
35. WAP to find list of prime numbers from tuple and store it into list.
36. WAP to create set of numbers and perform updating and deletion operation using its all inbuilt function.
37. WAP to perform all set operation like intersection, union, difference, symmetric difference and other operation.
38. WAP to remove all the duplicate element from list by using single list.
39. Write a Python script to generate and print a dictionary that contains a number (between 1 and n) in the form (x, x\*x).   
    Sample Dictionary . Expected Output : {1: 1, 2: 4, 3: 9, 4: 16, 5: 25} where value of n is enter by user.
40. Create a dictionary which shows the occurrence/ frequency of each character present in string. [hint : if user enter “wel come to parul university” than your dictionary contain {a:1,b:0,c:1,d:0,e:3, f:0……} like this.
41. Write a Python script to concatenate following dictionaries to create a new one.  Sample Dictionary :  
    dic1={1:10, 2:20}  
    dic2={3:30, 4:40}  
    dic3={5:50,6:60}
42. Write a Python program to combine two dictionary adding values for common keys. d1={'a':100,'b':200,'c':300}, d2={'a':300,'b':200,'d':400} **Sample output**: Counter({'a': 400, 'b': 400, 'd': 400, 'c': 300})
43. Write a python program to create dictionary of binary number up to given range. If user enter 15 than store binary number of 1 to 15 into dictionary and print it in proper format.
44. Write a menu driven program which shows the use of all inbuilt method of dictionary.
45. WAP to write your introduction like name, birth date, email id, address, contact details into file. And also print it in proper format after reading the data.
46. WAP which shows the use of seek() and tail() method to set and to get the position of file pointer.
47. WAP to store output of python script into text file. Open file using with statement.
48. Create below listed array using Numpy and check its data type.
49. type1 = np.array([1, 2, 3, 4, 5, 6])
50. type2 = np.array([1.5, 2.5, 0.5, 6])
51. type3 = np.array(['a', 'b', 'c'])
52. type4 = np.array(["Canada", "Australia"], dtype='U5')
53. type5 = np.array([555, 666], dtype=float)

1. Create below listed more than one dimensions array using Numpy and check its dimension
2. array1d = np.array([1, 2, 3, 4, 5, 6])
3. array2d = np.array([[1, 2, 3], [4, 5, 6]])
4. array3d = np.array([[[1, 2, 3], [4, 5, 6]], [[7, 8, 9], [10, 11, 12]]])
5. WAP Program to Transform List or Tuple into NumPy array.
6. Perform the following Indexing Operations using Numpy array.

array1d = np.array([1, 2, 3, 4, 5, 6])

1. Get first value
2. Get last value
3. Get 4th value from first
4. Get 5th value from last
5. Get multiple values
6. Perform the following Indexing Operations on Numpy array. Where array2d = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
7. Get first row first col (0,0) element
8. Get first row second col (0,1) element
9. Get first row second col (0,1) element
10. Get second row second col (2,1) element
11. Perform the followingSingle Dimensional Slicing Operations using Numpy array.

array1d = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

1. from index 4 to last index
2. From index 0 to 4 index
3. From index 4(included) up to index 7(excluded)
4. Excluded last element
5. Up to second last index(negative index)
6. From last to first in reverse order(negative step)
7. All odd numbers in reversed order
8. All even numbers in reversed order
9. All elements
10. Perform the following Multidimensional Dimensional Slicing Operations using Numpy array. Where array2d = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
11. 2nd and 3rd col
12. 2nd and 3rd row
13. Reverse an array
14. Resize array
15. Reshape array
16. Perform the following operations to Manipulating the Dimensions and the Shape of Arrays(Flips the order of the Axes)

array2d = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])

1. Permute the dimensions of an array

2. Flip array in the left/right direction

3. Flip array in the up/down direction

4. Rotate an array by 90 degrees in the plane specified by axes

1. Perform the following operations to Manipulating the Dimensions and the Shape of Arrays(Joining and Stacking)

array1 = np.array([[1, 2, 3], [4, 5, 6]])

array2 = np.array([[7, 8, 9], [10, 11, 12]])

1. Stack arrays in sequence horizontally (column wise).

2. Stack arrays in sequence vertically (row wise)

3. Stack arrays in sequence depth wise (along third axis)

4.Appending arrays after each other, along a given axis

5. Append values to the end of an array

1. Perform the following Arithmetic Operations using Numpy Array.

array1 = np.array([[1, 2, 3], [4, 5, 6]])

array2 = np.array([[7, 8, 9], [10, 11, 12]])

1. array1 + array2

2. array1 - array2

3. array1 \* array2

4. array2 / array1

5. array1 \*\* array2

1. Perform the following Scalar Arithmetic Operationsusing Numpy Array.

array1 = np.array([[10, 20, 30], [40, 50, 60]])

1. array1 + 2

2. array1– 5

3. array1 \* 2

4. array1 / 5

5. array1 \*\* 2

1. Perform the following Elementary Mathematical Functionsusing Numpy Array.

array1 = np.array([[10, 20, 30], [40, 50, 60]])

1. sin(array1)

2. cos(array1)

3.tan(array1)

4. sqrt(array1)

5. exp(array1)

6. log10(array1)

1. Perform the following Element-wise Mathematical Operations using Numpy Array.

array1 = np.array([[10, 20, 30], [40, 50, 60]])

array2 = np.array([[2, 3, 4], [4, 6, 8]])

array3 = np.array([[-2, 3.5, -4], [4.05, -6, 8]])

1. Addition & subtraction of array1 and array2

2. Multiplication & division & reminder of array1 and array2

3. Power of array1 and array2

1. Perform the following Aggregate and Statistical Functions using Numpy Array.

array1 = np.array([[10, 20, 30], [40, 50, 60]])

1. Mean

2. Standard deviation

3. Variance

4. Sum of array elements

5. Product of array elements

1. Use the Where(), Select() and Choose() function to identify the element is less than 4, mul by 2 else by 3.

np.array([[1, 2, 3], [4, 5, 6]])

1. Perform the following Logical Operations using Numpy Array.

thearray = np.array([[10, 20, 30], [14, 24, 36]])

1. logical\_or(Condition array<10, array>15)

2. logical\_and(Condition array<10, array>15)

3. logical\_not(Condition array<20)

1. Perform the following Standard Set Operations using Numpy Array.

array1 = np.array([[10, 20, 30], [14, 24, 36]])

array2 = np.array([[20, 40, 50], [24, 34, 46]])

1. Find the union of two arrays

2. Find the intersection of two arrays

3. Find the set difference of two arrays

<https://www.pythonprogramming.in/numpy-tutorial-with-examples-and-solutions.html>