1. Write a Python program which accepts the radius of a circle from the user and compute the area.
2. Write a Python program to test whether a number is within 100 of 1000 or 2000.
3. Write a Python program to calculate the sum of three given numbers, if the values are equal then return thrice of their sum.
4. Write a Python program to find whether a given number (accept from the user) is even or odd, print out an appropriate message to the user.
5. Write a Python program that will accept the base and height of a triangle and compute the area.
6. Write a Python program that will return true if the two given integer values are equal or their sum or difference is 5.
7. Write a Python program to solve (x + y) * (x + y).
8. Write a python program to sum of the first n positive integers.
9. Write a Python function to check whether a number is divisible by another number. Accept two integers values form the user.
10. Write a Python program to count the number of characters (character frequency) in a string
11. Write a Python program to get a string made of the first 2 and the last 2 chars from a given a string. If the string length is less than 2, return “empty string”.
12. Write a Python program to get a string from a given string where all occurrences of its first char have been changed to '$', except the first char itself.
13. Write a Python program to add 'ing' at the end of a given string (length should be at least 3). If the given string already ends with 'ing' then add 'ly' instead. If the string length of the given string is less than 3, leave it unchanged.
14. Write a Python program to count the occurrences of each word in a given sentence.
15. Write a Python program to sum all the items in a list.
16. Write a Python program to remove duplicates from a list.
17. Write a Python function that takes two lists and returns True if they have at least one common member.
18. Write a Python program to find the second smallest number in a list.
19. Write a Python program to get the maximum and minimum value in a dictionary.
20. Write a Python program to print all unique values in a dictionary.
21. Write a Python program to reverse a tuple.
22. Write a Python program to find those numbers which are divisible by 7 and multiple of 5, between 1500 and 2700 (both included).
23. Write a Python program which iterates the integers from 1 to 50. For multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print "FizzBuzz".
24. Write a Python class to reverse a string word by word.
25. Write a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.
26. Write a Python function that takes a number as a parameter and check the number is prime or not.
27. Write a Python program to print the even numbers from a given list.
28. Write a Python function to check whether a number is perfect or not
    According to Wikipedia : In number theory, a perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself (also known as its aliquot sum). Equivalently, a perfect number is a number that is half the sum of all of its positive divisors (including itself).
29. Write a Python program to determine whether a given year is a leap year.
30. Write a Python program to subtract five days from current date.

31. Write a Python program to print yesterday, today, tomorrow.
32. Write a Python program to print next 5 days starting from today.
33. Write a Python program to get days between two dates.
34. Write a Python program to count the number of characters (character frequency) in a string.
35. Write a Python program to add 'ing' at the end of a given string (length should be at least 3). If the given string already ends with 'ing' then add 'ly' instead. If the string length of the given string is less than 3, leave it unchanged.
36. Write a Python program to find the first appearance of the substring 'not' and 'poor' from a given string, if 'not' follows the 'poor', replace the whole 'not'...'poor' substring with 'good'. Return the resulting string.
37. Sample String : 'The lyrics is not that poor!'

    'The lyrics is poor!'

    Expected Result : 'The lyrics is good!'

    'The lyrics is poor!' Sample String : 'abc'

    Expected Result : 'abcing'

    Sample String : 'string'

    Expected Result : 'stringly'
38. Write a Python program to remove the nth index character from a nonempty string.
39. Write a Python script that takes input from the user and displays that input back in upper and lower cases.
40. Write a Python program that accepts a comma separated sequence of words as input and prints the unique words in sorted form (alphanumerically).
41. Write a Python program to sum all the items in a list.
42. Write a Python program to multiplies all the items in a list.
43. Write a Python program to get the largest number from a list.
44. Write a Python program to get a list, sorted in increasing order by the last element in each tuple from a given list of non-empty tuples.
45. Sample List : [(2, 5), (1, 2), (4, 4), (2, 3), (2, 1)]

    Expected Result : [(2, 1), (1, 2), (2, 3), (4, 4), (2, 5)]
46. Write a Python program to create a list by concatenating a given list which range goes from 1 to n.

47. Sample list : ['p', 'q']

    n =5

    Sample Output : ['p1', 'q1', 'p2', 'q2', 'p3', 'q3', 'p4', 'q4', 'p5', 'q5']

48. Write a Python script to check if a given key already exists in a dictionary.

49. Write a Python script to print a dictionary where the keys are numbers between 1 and 15 (both included) and the values are square of keys.

50. Write a Python program to combine two dictionary adding values for common keys.

51. 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})

52. Write a Python program to create a dictionary from a string.

    Note: Track the count of the letters from the string.

53. Write a Python program to find the repeated items of a tuple.

54. Write a Python program to find those numbers which are divisible by 7 and multiple of 5, between 1500 and 2700 (both included).

55. Write a Python program to count the number of even and odd numbers from a series of numbers.

56. Write a Python program to get the Fibonacci series between 0 to 50.

57. Write a Python program that accepts a string and calculate the number of digits and letters.

    Sample Data : Python 3.2

    Expected Output :

    Letters 6

    Digits 2

58. Write a Python function to multiply all the numbers in a list.

59. *Sample List* : (8, 2, 3, -1, 7)

    Expected Output : -336

60. Write a Python function that accepts a string and calculate the number of upper case letters and lower case letters.

61. Write a Python program that accepts a hyphen-separated sequence of words as input and prints the words in a hyphen-separated sequence after sorting them alphabetically.

    Sample Items : green-red-yellow-black-white

    Expected Result : black-green-red-white-yellow

62. Make a two-player Rock-Paper-Scissors game. (Hint: Ask for player plays (using input), compare them, print out a message of congratulations to the winner, and ask if the players want to start a new game)
63. Remember the rules:
64. Rock beats scissors
65. Scissors beats paper
66. Paper beats rock
67. Write a program that takes a list and returns a new list that contains all the elements of the first list minus all the duplicates.
68. Write a Python script to display the
    a) Current date and time
    b) Current year
    c) Month of year
    d) Week number of the year
    e) Weekday of the week
    f) Day of year
    g) Day of the month
    h) Day of week
69. Write a Python program to determine whether a given year is a leap year.
70. Write a Python program to convert a string to datetime.
71. Write a Python program to subtract five days from current date.
72. Write a Python program to print yesterday, today, tomorrow.
73. Write a Python program to get days between two dates.
74. Write a Python program calculates the date six months from the current date using the datetime module.
75. Write a Python program to convert two date difference in days, hours, minutes, seconds.
76. Write a Python program to read first n lines of a file.
77. Write a Python program to read a file line by line and store it into a list.
78. Write a Python program to count the number of lines in a text file.
79. Write a Python program to count the frequency of words in a file.
80. Write a Python program to copy the contents of a file to another file .
81. Write a Python program to write a list to a file.
82. Write a Python program to assess if a file is closed or not.