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

2. Write a Python program to convert temperatures to and from Celsius and Fahrenheit.
[Formula : c/5 = f-32/9 [where c = temperature in celsius and f = temperature in fahrenheit]
Expected Output :
60°C is 140 in Fahrenheit
45°F is 7 in Celsius

3. Write a Python program to guess a number between 1 and 9.
Note : User is prompted to enter a guess. If the user guesses wrong then the prompt appears again until the guess is correct, on successful guess, user will get a "Well guessed!" message, and the program will exit.

4. Write a Python program to construct the following pattern, using a nested for loop.

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

5. Write a Python program that accepts a word from the user and reverses it.

6. Write a Python program to count the number of even and odd numbers in a series of numbers
Sample numbers : numbers = (1, 2, 3, 4, 5, 6, 7, 8, 9)
Expected Output :

Number of even numbers : 5
Number of odd numbers : 4

7. Write a Python program that prints each item and its corresponding type from the following list.
Sample List : datalist = [1452, 11.23, 1+2j, True, 'w3resource', (0, -1), [5, 12], {"class":'V', "section":'A'}]

8. Write a Python program that prints all the numbers from 0 to 6 except 3 and 6.
Note : Use 'continue' statement.
Expected Output : 0 1 2 4 5

9. Write a Python program to get the Fibonacci series between 0 and 50.
Note : The Fibonacci Sequence is the series of numbers :
0, 1, 1, 2, 3, 5, 8, 13, 21, ...
Every next number is found by adding up the two numbers before it.
Expected Output : 1 1 2 3 5 8 13 21 34

10. Write a Python program that iterates the integers from 1 to 50. For multiples of three print "Fizz" instead of the number and for multiples of five print "Buzz". For numbers that are multiples of three and five, print "FizzBuzz".
Sample Output :
fizzbuzz
1
2
fizz
4
buzz

11. Write a Python program that takes two digits m (row) and n (column) as input and generates a two-dimensional array. The element value in the i-th row and j-th column of the array should be i*j.

Note :
i = 0,1.., m-1
j = 0,1, n-1.

Test Data : Rows = 3, Columns = 4
Expected Result : [[0, 0, 0, 0], [0, 1, 2, 3], [0, 2, 4, 6]]

12. Write a Python program that accepts a sequence of lines (blank line to terminate) as input and prints the lines as output (all characters in lower case).

13. Write a Python program that accepts a sequence of comma separated 4 digit binary numbers as its input. The program will print the numbers that are divisible by 5 in a comma separated sequence.
Sample Data : 0100,0011,1010,1001,1100,1001
Expected Output : 1010

14. Write a Python program that accepts a string and calculates the number of digits and letters.
Sample Data : Python 3.2
Expected Output :
Letters 6
Digits 2

15. Write a Python program to check the validity of passwords input by users.
Validation :

- At least 1 letter between [a-z] and 1 letter between [A-Z].
- At least 1 number between [0-9].
- At least 1 character from [$#@].
- Minimum length 6 characters.
- Maximum length 16 characters.

16. Write a Python program to find numbers between 100 and 400 (both included) where each digit of a number is an even number. The numbers obtained should be printed in a comma-separated sequence.

17. Write a Python program to print the alphabet pattern 'A'.
Expected Output:

```
  ***
 *   *
 *   *
 *****
 *   *
 *   *
 *   *
```

18. Write a Python program to print the alphabet pattern 'D'.
Expected Output:

```
 ****
 *   *
 *   *
 *   *
 *   *
 *   *
 ****
```

19. Write a Python program to print the alphabet pattern 'E'.
Expected Output:

```
 *****
 *
 *
 ****
 *
 *
 *****
```

20. Write a Python program to print the alphabet pattern 'G'.
Expected Output:

```
  ***
 *   *
 *
 *  ***
 *   *
 *   *
  ***
```

21. Write a Python program to print the alphabet pattern 'L'.
Expected Output:

```
*
*
*
*
*
*
*****
```

22. Write a Python program to print the alphabet pattern 'M'.
Expected Output:

```
*       *
*       *
* *   * *
*   *   *
*       *
*       *
*       *
```

23. Write a Python program to print the alphabet pattern 'O'.
Expected Output:

```
  ***
 *   *
 *   *
 *   *
 *   *
```

```
*    *
 ***
```

24. Write a Python program to print the alphabet pattern 'P'.
Expected Output:

```
****
*   *
*   *
****
*
*
*
```

25. Write a Python program to print the alphabet pattern 'R'.
Expected Output:

```
****
*   *
*   *
****
* *
*  *
*   *
```

26. Write a Python program to print the following patterns.
Expected Output:

```
 ****
*
*
 ***
    *
    *
****

 oooooooooooooooo
oooooooooooooooooo
oooooooooooooooooo
oooo
oooo
oooo
```

```
oooooooooooooooooo
oooooooooooooooooo
oooooooooooooooooo
                oooo
                oooo
                oooo
oooooooooooooooooo
oooooooooooooooooo
oooooooooooooooooo
```

27. Write a Python program to print the alphabet pattern 'T'.
Expected Output:

```
 *****
   *
   *
   *
   *
   *
   *
```

28. Write a Python program to print the alphabet pattern 'U'.
Expected Output:

```
*    *
*    *
*    *
*    *
*    *
*    *
 ***
```

29. Write a Python program to print the alphabet pattern 'X'.
Expected Output:

```
*    *
*    *
 * *
  *
 * *
*    *
*    *
```

30. Write a Python program to print the alphabet pattern 'Z'.
Expected Output:

```
*******
      *
     *
    *
   *
  *
*******
```

31. Write a Python program to calculate a dog's age in dog years.
Note: For the first two years, a dog year is equal to 10.5 human years. After that, each dog year equals 4 human years.
Expected Output:

```
Input a dog's age in human years: 15
The dog's age in dog's years is 73
```

32. Write a Python program to check whether an alphabet is a vowel or consonant.
Expected Output:

```
Input a letter of the alphabet: k
k is a consonant.
```

33. Write a Python program to convert a month name to a number of days.
Expected Output:

```
List of months: January, February, March, April, May, June,
July, August
, September, October, November, December
Input the name of Month: February
No. of days: 28/29 days
```

34. Write a Python program to sum two integers. However, if the sum is between 15 and 20 it will return 20.

35. Write a Python program that checks whether a string represents an integer or not.
Expected Output:

```
Input a string: Python
The string is not an integer.
```

36. Write a Python program to check if a triangle is equilateral, isosceles or scalene.
Note :
An equilateral triangle is a triangle in which all three sides are equal.
A scalene triangle is a triangle that has three unequal sides.
An isosceles triangle is a triangle with (at least) two equal sides.
Expected Output:

```
Input lengths of the triangle sides:
x: 6
y: 8
z: 12
Scalene triangle
```

37. Write a Python program that reads two integers representing a month and day and prints the season for that month and day.
Expected Output:

```
Input the month (e.g. January, February etc.): july
Input the day: 31
Season is autumn
```

38. Write a Python program to display the astrological sign for a given date of birth.
Expected Output:

```
Input birthday: 15
Input month of birth (e.g. march, july etc): may
Your Astrological sign is : Taurus
```

39. Write a Python program to display the sign of the Chinese Zodiac for the given year in which you were born.
Expected Output:

```
Input your birth year: 1973
Your Zodiac sign : Ox
```

40. Write a Python program to find the median of three values.
Expected Output:

```
Input first number: 15
Input second number: 26
Input third number: 29
The median is 26.0
```

41. Write a Python program to get the next day of a given date.
Expected Output:

```
Input a year: 2016
Input a month [1-12]: 08
Input a day [1-31]: 23
The next date is [yyyy-mm-dd] 2016-8-24
```

42. Write a Python program to calculate the sum and average of n integer numbers (input from the user). Input 0 to finish.

43. Write a Python program to create the multiplication table (from 1 to 10) of a number.
Expected Output:

```
Input a number: 6
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
```

```
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60
```

44. Write a Python program to construct the following pattern, using a nested loop number.

Expected Output:

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```