Solution,17jan,2022

1.write a program to check whether the string is empty or not?

Solution:

```
>>> strA = " "

>>> # checking if string with space is empty
>>> print("Check if the strA with space is empty : ", end="")
>>> if(not strA):
>>>     print("The string is empty")
>>> else:
>>>     print("No it is not empty")
```

2. Write a Python program to Find the strings in a list, starting with a given prefix?

Solution:

```
>>> def test(strs, prefix):
>>>      return [s for s in strs if s.startswith(prefix)]
>>> strs =  ['cat', 'car', 'fear', 'center']
>>> prefix = "ca"
>>> print("Original strings:")
>>> print(strs)
>>> print("Starting prefix:", prefix)
>>> print("Strings with a given prefix:")
>>> print(test(strs, prefix))
```

3.Write a program, which will find all such numbers between 1000 and 3000 (both included) such that each digit of the number is an even number.
The numbers obtained should be printed in a comma-separated sequence on a single line.?

Solution:

```
>>> values = []
>>> for i in range(1000, 3001):
>>>     s = str(i)
>>>     if (int(s[0])%2==0) and (int(s[1])%2==0) and (int(s[2])%2==0) and (int(s[3])%2==0):
>>>         values.append(s)
>>> print( ",".join(values))
```

4.Define a function which can generate a dictionary where the keys are numbers between 1 and 20 (both included) and the values are square of keys. The function should just print the values only.?

Solution:

```
>>> def printDict():
>>>     d=dict()
```

```
>>>     for i in range(1,21):
>>>             d[i]=i**2
>>>     for (k,p) in d.items():
>>>             print (p)


>>> printDict()
```

5. Write a program which can filter even numbers in a list by using filter function. The list is: [1,2,3,4,5,6,7,8,9,10,45,67,88,90,68,56].

Solution:

```
>>> li = [1,2,3,4,5,6,7,8,9,10,45,67,88,90,68,56]
>>> evenNumbers = filter(lambda x: x%2==0, li)
>>> print(evenNumbers)
```

6.write a program to Find solution of quadratic equation?.

Solution:

```
>>> import cmath

>>> a = 1
>>> b = 5
>>> c = 6

>>>#calculate the discriminant
>>> d = (b**2) - (4*a*c)

>>> # find two solutions
>>> sol1 = (-b-cmath.sqrt(d))/(2*a)
>>> sol2 = (-b+cmath.sqrt(d))/(2*a)

>>> print('The solution are {0} and {1}'.format(sol1,sol2))
```

7. Write a Python Program to find the area of triangle?

Solution:

```
>>>  a = float(input('Enter first side: '))
>>> b = float(input('Enter second side: '))
>>>  c = float(input('Enter third side: '))

>>> # calculate the semi-perimeter
>>> s = (a + b + c) / 2
```

```
>>> # calculate the area
>>> area = (s*(s-a)*(s-b)*(s-c)) ** 0.5
>>> print('The area of the triangle is %0.2f' %area)
```

8.write a program to get the all permutations of string?

Solution:

```
>>> def get_permutation(string, i=0):

>>>     if i == len(string):
>>>         print("".join(string))

>>>     for j in range(i, len(string)):

>>>         words = [c for c in string]

>>>         # swap
        words[i], words[j] = words[j], words[i]

>>>         get_permutation(words, i + 1)

>>> print(get_permutation('yup'))
```

Or

```
>>> from itertools import permutations

>>> words = [''.join(p) for p in permutations('pro')]

>>>print(words)
```

9.write a program to remove all punctuation from the string?

Solution:

```
>>> #define punctuation
>>>punctuations='''!()-[]{};:'"\,<>./?@#$%^&*_~'''


>>> # To take input from the user
>>> my_str = input("Enter a string: ")

>>> # remove punctuation from the string
>>> no_punct = ""
>>> for char in my_str:
>>>     if char not in punctuations:
```

```
>>>         no_punct = no_punct + char

>>> # display the unpunctuated string
>>> print(no_punct)
```

10.write a Program to transpose a matrix using a nested loop?

Solution:

```
>>> X = [[12,7],
    [4 ,5],
    [3 ,8]]

>>> result = [[0,0,0],
          [0,0,0]]


>>> # iterate through rows
>>> for i in range(len(X)):
>>>    # iterate through columns
>>>     for j in range(len(X[0])):
>>>         result[j][i] = X[i][j]

>>> for r in result:
>>>     print(r)
```