Question 1. Create a function prime() that receives an integer and returns whether n is
prime or not. Print all prime numbers from 1 to 100 by calling prime() function.

In [1]:

```python
def prime(n):
    if n <= 1:
        return False
    for i in range(2, n):
        if n % i == 0:
            return False
    return True
for i in range(1, 101):
    if prime(i):
        print(i)
```

```
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
79
83
89
97
```

Question 2. Develop a simple arithmetic calculator for 4 operations. The program should
continue calculation until user types ‘q’ to quit. A sample user interaction can be:
⍰ Enter operator (q to quit): +
⍰ Enter value 1: 10
⍰ Enter value 2: 20
⍰ Result = 30
Create 4 functions add(), subtract(), multiply() and divide() that receives two values and

returns the result of the operation.

In [13]:

```
def add(a,b):
    c = a + b
    return c
def subtract(a,b):
    c = a - b
    return c
def multiply(a,b):
    c = a * b
    return c
def divide(a,b):
    c = a / b
    return c
a = int(input("Enter the number:"))
b = int(input("Enter the number:"))
result = add(a,b)
result_2 = subtract(a,b)
result_3 = multiply(a,b)
result_4 = divide(a,b)
print(result)
print(result_2)
print(result_3)
print(result_4)
```

```
Enter the number:2
Enter the number:3
5
-1
6
0.6666666666666666
```

Question3. Create a function factorial() that takes an integer and returns its factorial value.
 You can create as a non-recursive version of factorial.
 Also, check factorial of negative number does not exist.
 Factorial of 0 is 1.
 Save this Python file as factorial_definition.py.

In [7]:

```
def factorial(a):
    if(a<2):
        return 1
    else:
        return a * factorial(a-1)
a=int(input())
result = factorial(a)
print(result)
```

```
6
720
```

Question4. Develop a function count_letter(string, search) that returns the number of times search character appears in a string.

In [8]:

```
def count_letters(word, char):
    count = 0
    while count <= len(word):
        for char in word:
            if char == word[count]:
                count += 1
        return count
result = count_letters('Hello World', 'o')
print (result)
```

```
11
```

Question5. Write a program that counts the number of spaces, digits, vowels and
consonants in a string that the user inputs. Print the string, no of spaces, no of digits, no of
vowels and no of consonants.

In [9]:

```
def count_chars(string):
    vowels = 0
    consonants = 0
    digits = 0
    spaces = 0
    for char in string:
        if char.isalpha():
            if char.lower() in ['a', 'e', 'i', 'o', 'u']:
                vowels += 1
            else:
                consonants += 1
        elif char.isdigit():
            digits += 1
        elif char.isspace():
            spaces += 1
    print("String", string)
    print("No of spaces:",spaces)
    print("No of digits:",digits )
    print("No of vowels:", vowels)
    print("No of consonants:", consonants)
count_chars("Bishop Heber College 17")
```

```
String Bishop Heber College 17
No of spaces: 3
No of digits: 2
No of vowels: 7
No of consonants: 11
```

Question6. Develop a function remove_punctuation(str) that returns the string after
removing the following punctuations.
Punctuation List = "!\"#$%&'()*+,-./:;<=>?@[\]^`{|}~"

In [10]:

```
def remove_punctuation(str):
    punctuations = '''!\"#$%&'()*+,-./:;<=>?@[\]^`{|}~'''
    for char in str:
        if char in punctuations:
            str = str.replace(char, "")
    return str
str1= "Bishop's College !....."
str2 = "#bhc trending @cs $placements::>."
print (remove_punctuation (str1))
print (remove_punctuation(str2))
```

```
BishopsCollege
bhctrendingcsplacements
```

Question7. Write a program that asks the user for a word. Translate their word into Pig
Latin. Pig Latin game takes the first consonant (or set of first consonants) of an English word,
moves it to the end of the word and suffixes an ay. If the first letter is a vowel, do not move
that vowel, but instead add “way” at the end of the word.

In [11]:

```
def piglatin (word):
    vowels = ['a', 'e', 'i', 'o', 'u']
    if word[0] in vowels:
        return word + '-way'
    else:
        return word[1:] + '-' + word[0] +'ay'
print(piglatin('pig'))
print(piglatin('banana'))
print(piglatin('trash'))
print(piglatin('apple'))
print(piglatin('orange'))
```

```
ig-pay
anana-bay
rash-tay
apple-way
orange-way
```

In [ ]: