1. Write a Python function to sum all the numbers in a list.

Sample List : (8, 2, 3, 0, 7)  
Expected Output : 20

**CODE:**

def sum\_list(numbers):

total = 0

for num in numbers:

total += num

return total

sample\_list = [8, 2, 3, 0, 7]

result = sum\_list(sample\_list)

print("Sum of the numbers in the list:", result)

1. Write a Python function to multiply all the numbers in a list.   
   Sample List : (8, 2, 3, -1, 7)  
   Expected Output : -336

**CODE:**

def multiply\_list(numbers):

    result = 1

    for num in numbers:

        result \*= num

    return result

sample\_list = [8, 2, 3, -1, 7]

result = multiply\_list(sample\_list)

print("Multiplication of the numbers in the list:", result)

1. Write a Python function to check whether a number is in a given range.

**CODE:**

def test\_range(n):

    if n in range(3, 9):

        print("%s is in the range" % str(n))

    else:

        print("The number is outside the given range.")

test\_range(5)

1. Write a Python function that takes a list and returns a new list with unique elements of the first list.

Sample List : [1,2,3,3,3,3,4,5]  
Unique List : [1, 2, 3, 4, 5]

**CODE:**

def unique\_list(l):

    x = []

    for a in l:

        if a not in x:

            x.append(a)

    return x

print(unique\_list([1, 2, 3, 3, 3, 3, 4, 5]))

1. Write a Python function that takes a number as a parameter and check the number is prime or not.

Note : A prime number (or a prime) is a natural number greater than 1 and that has no positive divisors other than 1 and itself.

**CODE:**

import math

def is\_prime(number):

    if number <= 1:

        return False

    for i in range(2, int(math.sqrt(number)) + 1):

        if number % i == 0:

            return False

    return True

num = 17

if is\_prime(num):

    print(num, "is a prime number.")

else:

    print(num, "is not a prime number.")

1. Write a Python program to print the even numbers from a given list.

**CODE:**

def print\_even\_numbers(sample\_list):

    print("Even numbers from the given list:")

    for num in sample\_list:

        if num % 2 == 0:

            print(num)

sample\_list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

print\_even\_numbers(sample\_list)

1. Write a Python function to calculate the factorial of a number (a non-negative integer). The function accepts the number as an argument.

**CODE:**

def factorial(n):

    if n == 0:

        return 1

    else:

        return n \* factorial(n - 1)

n = int(input("Input a number to compute the factorial: "))

print(factorial(n))

1. Write a Python function to check whether a number is perfect or not. [Go to the editor](https://www.w3resource.com/python-exercises/python-functions-exercises.php#EDITOR)  
   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).  
   Example : The first perfect number is 6, because 1, 2, and 3 are its proper positive divisors, and 1 + 2 + 3 = 6. Equivalently, the number 6 is equal to half the sum of all its positive divisors: ( 1 + 2 + 3 + 6 ) / 2 = 6. The next perfect number is 28 = 1 + 2 + 4 + 7 + 14. This is followed by the perfect numbers 496 and 8128.

def perfect\_number(n):

    sum = 0

    for x in range(1, n):

        if n % x == 0:

            sum += x

    return sum == n

print(perfect\_number(6))

1. Write a Python function to create and print a list where the values are square of numbers between 1 and 30 (both included).

def printValues():

    l = list()

    for i in range(1, 31):

        l.append(i\*\*2)

    print(l)

printValues()