1.Write a python function which should be capable of finding the factorial of any given number as an argument.

```
fact=1
 for i in range(1,num+1):
    fact=fact*i
 return fact
print(factorial(100))
```

2. Luke Skywalker has family and friends. Help him remind them who is who. Given a string with a name, return the relation of that person to Luke.

```
def relation_to_luke(name):
    relation={'Darth Vader':'father','Lela':'sister','Han':'brother in
law','R2D2':'droid'}
    return relation[name]

print("Luke, I am your "+ relation_to_luke('Darth Vader'))
```

3. Create a function which takes a number as its argument and return the number of digits in it. Use of len function is not allowed. For example for 5 it should return 1, for 32 it should return 2 and 123 , 3 should be returned and so on.

```
def no_of_digits(num):
    temp=num
    count=0
    while(temp>0):
        x=temp%10
        count+=1
        temp=temp//10

    return count

print(no_of_digits(123))
```

4. Write a function which takes a number as argument suppose 5 and gives results as multiplication of factorial of each positive number less than or equal to the number given. i.e !5*!4*!3*!2*!1 = 34560.

```
def factorial(num):
  fact=1
  for i in range(1,num+1):
```

```
      fact=fact*i
  return fact

def mul_fact(num):
    temp=num
    result=1
    while(temp>0):
        curr_fact=factorial(temp)
        result=result*curr_fact
        temp-=1
    return result

print(mul_fact(5))
```

5. Write a function which takes any number of arguments from a user and return the result which should be output of a2 + b2+ c2+... if a , b ,c are numbers supplied ..i.e if 1,2,3 are supplied then result returned should be 14. But user may supply any number of inputs so make the function to adapt to that.

```
def args_example(*argv):
    result=0
    for num in argv:

        temp=num**2
        result=result+temp
    return result

print(args_example(1,2,3,4))
```

6.Write a function which accepts 3 arguments from the user.1 .number 1, 2. Number2 and 3. An operation. The operation supported should be +, -, *, and /. The function should return the result of given operation. For example arguments are 3,2,+ then result returned should be 5.

```
def operation(num1,num2,op):
    if(op not in ['+','-','*','/']):
        return
    else:
        if(op=='+'):
            return num1+num2
        elif(op=='-'):
            return num1-num2
        elif(op=='*'):
            return num1*num2
        elif(op=='/'):
            return num1/num2

print(operation(2,3,'^'))
```

7. Write a function which takes an argument which should be a numeric +ve integer. Depending on the input supplied you have to print “I CAN”, “I WILL”. Suppose some one enters argument as 1 then only “I CAN” should be printed. But if some one enters 2 then first “I CAN” should be printed then “I WILL”. And if someone enters 3 then following should be printed in corresponding order: “I CAN”, “I WILL”, “I CAN” and so on for any numbers entered.

```
def i_can_i_will(num):
    temp=num
    a=0
    while(temp>0):
        if(a==0):
            print("I CAN",end=' ')
            a=1
        else:
            print("I WILL",end=' ')
            a=0
        temp-=1

i_can_i_will(4)
```

8. We have been given a list of whole numbers which represents the color of each gloves, determine how many pairs of gloves with matching colors there are. For example, there are 7 gloves with colors [1, 2, 1, 2, 1, 3, 2]. There is one pair of color 1 and one of color 2. There are three odd gloves left, one of each color. The number of pairs is 2. Create a function that returns an integer representing the number of matching pairs of gloves that are available.

[1,2,1,2,1,3,4,2]

[1,1,1,2,2,2,3,4]

```
def consecutive_sequence(l1):
    l1.sort()
    i=0
    count=0
    flag=True
    while(i<len(l1)-1):
        #print(l1[i+1])
        if(l1[i+1]-l1[i]==0):
            count+=1
            i+=2

        else:
            i+=1
    return count



print(consecutive_sequence([1,2,1,2,1,3,2,4]))
```

8. We have been given a list of whole numbers which represents the color of each gloves, determine how many pairs of gloves with matching colors there are. For example, there are 7 gloves with colors [1, 2, 1, 2, 1, 3, 2]. There is one pair of color 1 and one of color 2. There are three odd gloves left, one of each color. The number of pairs is 2. Create a function that returns an integer representing the number of matching pairs of gloves that are available.

9.

```
def consecutive_sequence(l1,l2):
    l1.extend(l2)
    l1.sort()
    i=0
    flag=True
    while(i<len(l1)-1):
        print(l1[i+1])
        if(l1[i+1]-l1[i]==1):

            i+=1
            flag=True
        else:
            flag=False
            break
    return flag
```

10. You work for a manufacturer, and have been asked to calculate the total profit made on the sales of a product. You are given a dictionary containing the cost price per unit (in dollars), sell price per unit (in dollars), and the starting inventory. Return the total profit made, rounded to the nearest dollar.

```python
def profit(adictionary):
    profit = int((adictionary['sell_price'] - adictionary['cost_price']) * adictionary['inventory'])
    return profit

print(profit({
   "cost_price": 32.67,
   "sell_price": 45.00,
   "inventory": 1200
 }))
```