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

In [1]:
```
def mul(list1):
    result =1
    for item in list1:
        result+=item
    return result

l1 =[1,2,3,4,5]
mul(l1)
```

Out[1]: 16

2. Write a Python function to reverse a string use the while loop also?

In [8]:
```
str1 = "this is a string"
idx = -1
while idx >= (0-len(str1)):
    # print(idx,end=" ")
    print(str1[idx],end= "")
    idx-=1
```

```
gnirts a si siht
```

3. Write a function to add and subtract two variables

In [10]:
```
def add_sub(num1, num2):
    print("addition: ",num1+num2)
    print("subtraction: ", num1-num2)

num1=12
num2=8
add_sub(num1,num2)
```

```
addition:  20
subtraction:  4
```

4. Write a function to check the number is divisible by 12

In [11]:
```
def check(num):
    if num%12==0:
        return True
    else:
        return False
nums = (1, 24, 32,36)
for num in nums:
    print(f"is {num} divisible by 12: {check(num)}")
```

```
is 1 divisible by 12: False
is 24 divisible by 12: True
is 32 divisible by 12: False
is 36 divisible by 12: True
```

5. Write a function to calculate the number of days and weeks

In [30]:
```
def calculate_days_and_weeks(year):
    if year%4==0:
        days=366
    else:
        days = 365

    weeks = days//7
    return (days, weeks)

(days,weeks)=calculate_days_and_weeks(2014)
print(days, weeks)
```

```
365 52
```

6.Write a Python function to Find the 5!?

In [12]:
```
def fact(num):
    if num ==1:
        return num
    result = num * fact(num-1)
    return result
```

```
num =5
print(fact(num))
```

```
120
```

7. Write a Python function Find the unique elements of the first list = [1,2,3,3,3,3,4,5,4,2,4,2,4,4,2,4,5,4,34,654,5,7,6,5,4,3,]?

In [14]:
```
def unique_elements(list1):
    return list(set(list1))

l1 = [1,2,3,3,3,3,4,5,4,2,4,2,4,4,2,4,5,4,34,654,5,7,6,5,4,3]
print(unique_elements(l1))
```

```
[1, 2, 3, 4, 5, 34, 7, 6, 654]
```

Types of arguments

1. Required arguments
2. Keyword arguments
3. Default arguments

1. Required arguments: (the function simple_interest accepts three arguments and returns the simple interest accordingly)

In [16]:
```
def calc_si(principal, ir, duration):
    return principal*(ir/100)*duration
principal = 5000
ir = 10
duration = 4
calc_si(principal, ir, duration)
```

Out[16]:
```
2000.0
```

2.Keyword arguments:(Function is called with the name and message as the keyword arguments)

In [17]:
```
calc_si(duration=2, principal=10000, ir=2)
```

Out[17]: 400.0

3.Default Arguments:()

In [18]:
```
def calc_si(principal, ir=5, duration=2):
    return principal*(ir/100)*duration

calc_si(20000)
```

Out[18]: 2000.0

lambda function

1. Write a lambda function to find the product of two numbers

In [20]:
```
mul = lambda x,y: x*y
mul(12,2)
```

Out[20]: 24

Map

1. Adding two lists using map

```
lis1 = [12, 24, 36]
lis2 = [41, 54, 69]
```

Map

In [23]:
```
 lis1 = [12, 24, 36]
 lis2 = [41, 54, 69]

idx= 0
sum_list = list(map(lambda x:x+lis2[lis1.index(x)], lis1))
sum_list
```

Out[23]: [53, 78, 105]

1.filter the lis1 = [3,12, 24, 36,43,654,65432,2,654,455,43,543] which is not divided by 2?

In [28]:
```
lis3 = [3,12, 24, 36,43,654,65432,2,654,455,43,543]
filtered_list = list(filter(lambda x: x if x%2==0 else None, lis3))
filtered_list
```

Out[28]: [12, 24, 36, 654, 65432, 2, 654]