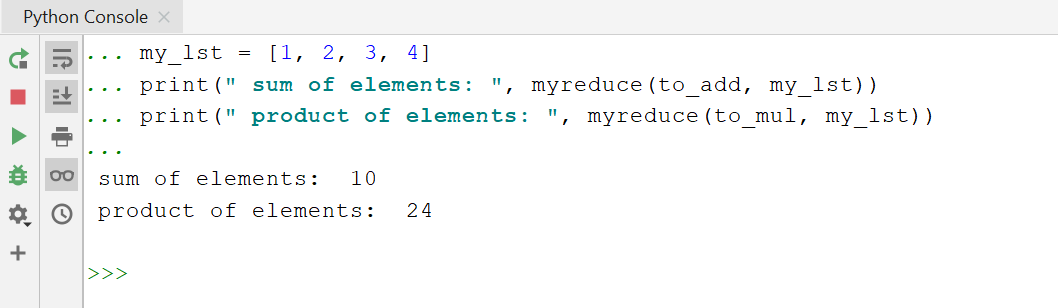
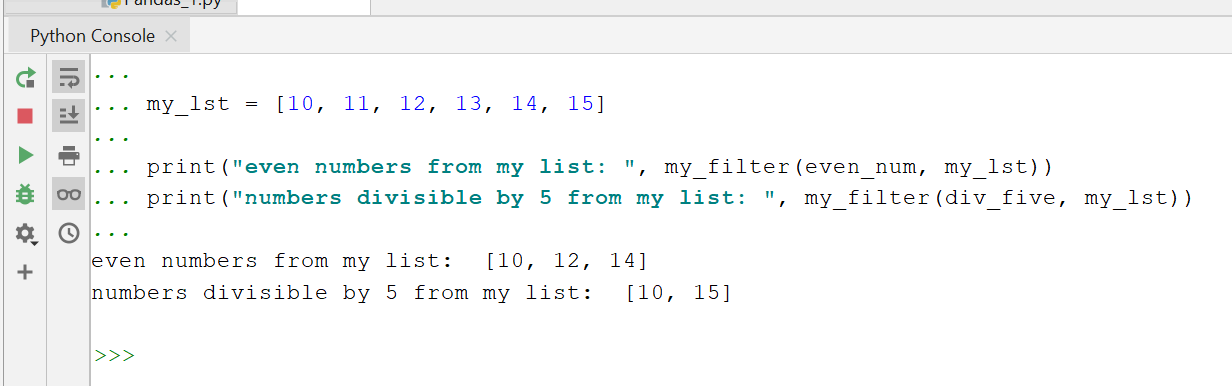
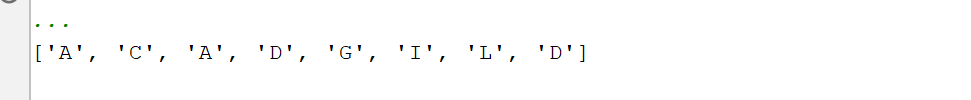
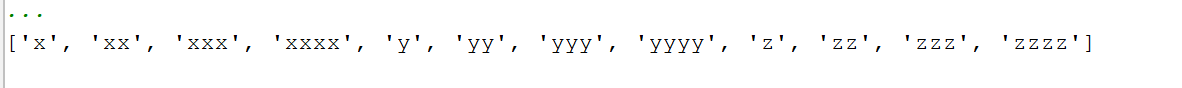
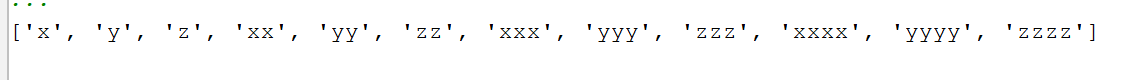
*# 1.1 create your own reduce function in python***def** myreduce(fun, lst):  
 a = lst[0] *# storing the first value of list to be used in function* **for** i **in** range(1, len(lst)):  
 a = fun(a, lst[i])  
  
 **return** a  
  
**def** to\_add(p, q):  
 **return** p + q  
  
**def** to\_mul(p, q):  
 **return** p \* q  
  
my\_lst = [1, 2, 3, 4]  
print(**" sum of elements: "**, myreduce(to\_add, my\_lst))  
print(**" product of elements: "**, myreduce(to\_mul, my\_lst))

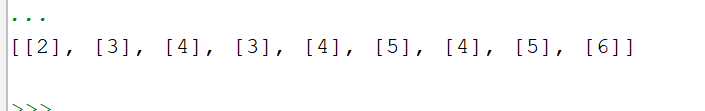
  
  
*# 1.2 create own myfilter function***def** my\_filter(fun, lst):  
 a = []  
 **for** i **in** range(0, len(lst)):  
 **if** fun(lst[i]) **is True**:  
 a.append(lst[i])  
  
 **return** a  
  
**def** even\_num(x):  
 **if** x % 2 == 0:  
 **return True  
  
def** div\_five(x):  
 **if** x % 5 == 0:  
 **return True**my\_lst = [10, 11, 12, 13, 14, 15]  
  
print(**"even numbers from my list: "**, my\_filter(even\_num, my\_lst))  
print(**"numbers divisible by 5 from my list: "**, my\_filter(div\_five, my\_lst))

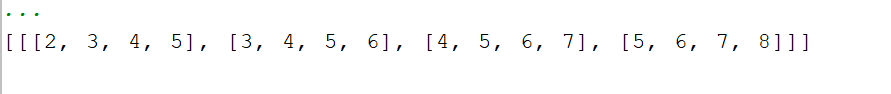
  
  
*# 2 List Comprehensions #################*list1 = [i **for** i **in 'ACADGILD'**]  
print(list1) *# ['A', 'C', 'A', 'D', 'G', 'I', ’L’, ‘ D’]*

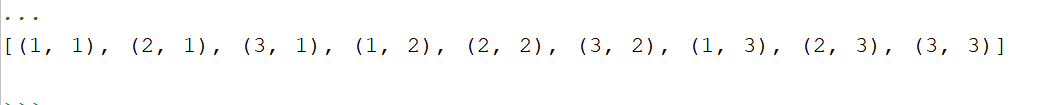
list2 = [i \* j **for** j **in** [**'x'**, **'y'**, **'z'**] **for** i **in** range(1, 5)]  
print(list2) *# ['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzzz']*

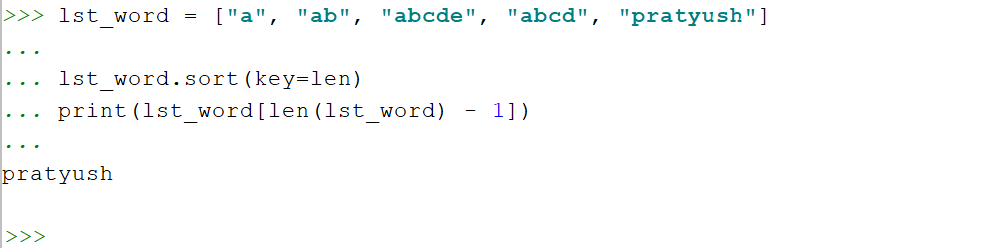
list3 = [i \* j **for** j **in** range(1, 5) **for** i **in** [**'x'**, **'y'**, **'z'**]]  
print(list3)

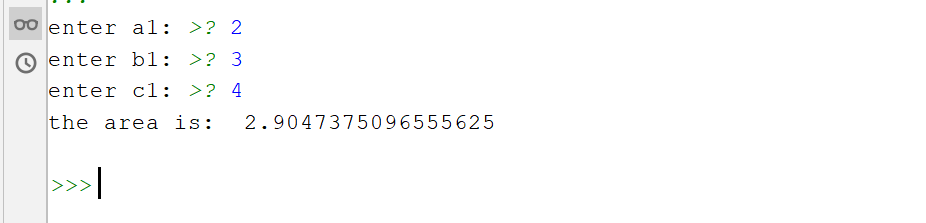
  
  
list4 = [[i + j] **for** j **in** [1, 2, 3] **for** i **in** range(1, 4)]  
print(list4)

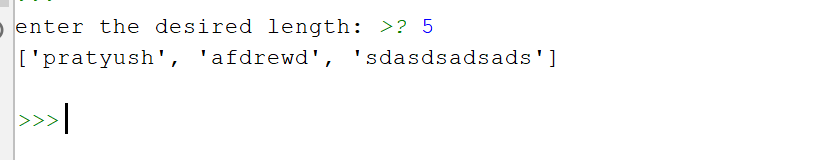
  
  
list5 = [[[i + j **for** j **in** [0, 1, 2, 3]] **for** i **in** range(2, 6)]]  
print(list5)

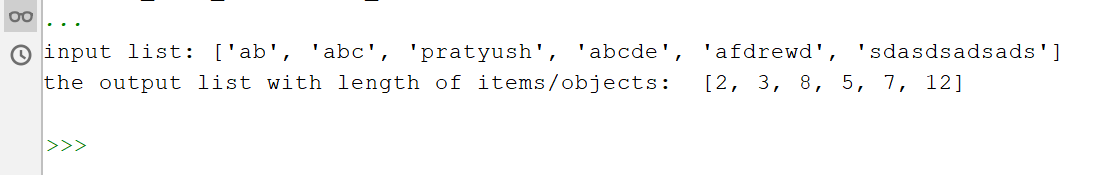
  
  
list6 = [(i, j) **for** j **in** [1, 2, 3] **for** i **in** [1, 2, 3]] *# left to right operation for loop.*print(list6)

  
  
*# 3 : function to retrieve longest word ###*lst\_word = [**"a"**, **"ab"**, **"abcde"**, **"abcd"**, **"pratyush"**]  
  
lst\_word.sort(key=len)  
print(lst\_word[len(lst\_word) - 1])

  
  
*# task 2 : 1.1 ###########################***import** math  
  
**class** Tri:  
 **def** \_\_init\_\_(self, a, b, c):  
 self.a = a  
 self.b = b  
 self.c = c  
  
**class** Area(Tri):  
 **def** \_\_init\_\_(self, \*args):  
 super(Area, self).\_\_init\_\_(\*args)  
  
 **def** area\_calc(self):  
 s = (self.a + self.b + self.c) / 2  
 **return** math.sqrt(s \* (s - self.a) \* (s - self.b) \* (s - self.c))  
  
a1 = int(input(**"enter a1: "**))  
b1 = int(input(**"enter b1: "**))  
c1 = int(input(**"enter c1: "**))  
  
area\_val = Area(a1, b1, c1)  
print(**"the area is: "**, area\_val.area\_calc())

  
  
*# task 2 : 1.2 ###################################################*a = []  
**def** filter\_long\_words(lst, n):  
 **for** i **in** range(0, len(lst)):  
 **if** len(lst[i]) > n:  
 a.append(lst[i])  
 print(a)  
  
lst\_word = [**'a'**, **'abc'**, **'pratyush'**, **'abcde'**, **'afdrewd'**, **'sdasdsadsads'**]  
*# print(len(lst\_word[1]))*numb = int(input(**"enter the desired length: "**))  
filter\_long\_words(lst\_word, numb)

  
*# task 2.1 ###################################################*p = []  
**def** lst\_item\_length(lst):  
 **for** i **in** range(0, len(lst)):  
 p.append(len(lst[i]))  
 print(**"the output list with length of items/objects: "**, p)  
  
lst\_word = [**'ab'**, **'abc'**, **'pratyush'**, **'abcde'**, **'afdrewd'**, **'sdasdsadsads'**]  
print(**"input list:"**, lst\_word)  
lst\_item\_length(lst\_word)

  
  
*# task 2.2 ###################################################***def** is\_vowel(n):  
 vowel\_set = **'aeiou'  
 if** n **in** vowel\_set:  
 print(**'True'**)  
 **else**:  
 print(**'False'**)  
  
user\_char = input(**"enter the char: "**)  
is\_vowel(user\_char)

