Рекурсия

|  |  |
| --- | --- |
| def sum\_list(L):      print(L)      return 0 if not L else L[0]+ sum\_list(L[1:])  list=[1,2,3,4,5,6,7,8,9,10]  print(sum\_list(list)) | Рекурсия. Вызов функции на саму себя.  [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  [2, 3, 4, 5, 6, 7, 8, 9, 10]  [3, 4, 5, 6, 7, 8, 9, 10]  [4, 5, 6, 7, 8, 9, 10]  [5, 6, 7, 8, 9, 10]  [6, 7, 8, 9, 10]  [7, 8, 9, 10]  [8, 9, 10]  [9, 10]  [10]  []  55 |
| def sum\_tree(L):      sum=0      for i in L:          if not isinstance(i, list) and not isinstance(i, tuple):              sum=sum+i          else:              sum = sum + sum\_tree(i)      return sum  list\_r= [1,[2,3,4],5,[((6),7),8,9],10]  print(sum\_tree(list\_r)) | Функция рекурсивная считает сумму всех элементов списка включая все вложенные структуры списков и кортежей |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |