1.6 Наследование классов

Реализуйте структуру данных, представляющую собой расширенную структуру стек. Необходимо поддерживать добавление элемента на вершину стека, удаление с вершины стека, и необходимо поддерживать операции сложения, вычитания, умножения и целочисленного деления.

Операция сложения на стеке определяется следующим образом. Со стека снимается верхний элемент (**top1**), затем снимается следующий верхний элемент (**top2**), и затем как результат операции сложения на вершину стека кладется элемент, равный **top1 + top2**.

Аналогичным образом определяются операции вычитания (**top1 - top2)**, умножения (**top1 \* top2**) и целочисленного деления (**top1 // top2**).

Реализуйте эту структуру данных как класс **ExtendedStack**, отнаследовав его от стандартного класса **list**.  
Требуемая структура класса:

class ExtendedStack(list):  
 def sum(self):  
 # операция сложения  
  
 def sub(self):  
 # операция вычитания  
  
 def mul(self):  
 # операция умножения  
  
 def div(self):  
 # операция целочисленного деления

**Примечание**

Для добавления элемента на стек используется метод **append**, а для снятия со стека – метод **pop**.

Гарантируется, что операции будут совершаться только когда в стеке есть хотя бы два элемента.