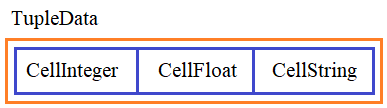
Видео-разбор подвига (решение смотреть только после своей попытки): <https://youtu.be/wLaOyNN8x7E>

**Значимый подвиг 7.** Вам поручается разработать класс TupleData, элементами которого могут являются только объекты классов: CellInteger, CellFloat и CellString.



Вначале в программе нужно объявить класс CellInteger, CellFloat и CellString, объекты которых создаются командами:

cell\_1 = CellInteger(min\_value, max\_value)

cell\_2 = CellFloat(min\_value, max\_value)

cell\_3 = CellString(min\_length, max\_length)

где min\_value, max\_value - минимальное и максимальное допустимое значение в ячейке; min\_length, max\_length - минимальная и максимальная допустимая длина строки в ячейке.

В каждом объекте этих классов должны формироваться локальные атрибуты с именами \_min\_value, \_max\_value или \_min\_length, \_max\_length и соответствующими значениями.

Запись и считывание текущего значения в ячейке должно выполняться через объект-свойство (property) с именем:

**value** - для записи и считывания значения в ячейке (изначально возвращает значение None).

Если в момент записи новое значение не соответствует диапазону [min\_value; max\_value] или [min\_length; max\_length], то генерируется исключения командами:

raise CellIntegerException('значение выходит за допустимый диапазон')  # для объектов класса CellInteger

raise CellFloatException('значение выходит за допустимый диапазон')    # для объектов класса CellFloat

raise CellStringException('длина строки выходит за допустимый диапазон')  # для объектов класса CellString

Все три класса исключений должны быть унаследованы от одного общего класса:

**CellException**

Далее, объявите класс TupleData, объекты которого создаются командой:

ld = TupleData(cell\_1, ..., cell\_N)

где cell\_1, ..., cell\_N - объекты классов CellInteger, CellFloat и CellString (в любом порядке и любом количестве).

Обращение к отдельной ячейке должно выполняться с помощью оператора:

value = ld[index] # считывание значения из ячейке с индексом index

ld[index] = value # запись нового значения в ячейку с индексом index

Индекс index отсчитывается с нуля (для первой ячейки) и является целым числом. Если значение index выходит за диапазон [0; число ячеек-1], то генерировать исключение **IndexError**.

Также с объектами класса TupleData должны выполняться следующие функции и операторы:

res = len(ld) # возвращает общее число элементов (ячеек) в объекте ld

for d in ld:  # перебирает значения ячеек объекта ld (значения, а не объекты ячеек)

    print(d)

Все эти классы в программе можно использовать следующим образом:

ld = TupleData(CellInteger(0, 10), CellInteger(11, 20), CellFloat(-10, 10), CellString(1, 100))

try:

ld[0] = 1

ld[1] = 20

ld[2] = -5.6

ld[3] = "Python ООП"

except CellIntegerException as e:

print(e)

except CellFloatException as e:

print(e)

except CellStringException as e:

print(e)

except CellException:

print("Ошибка при обращении к ячейке")

except Exception:

print("Общая ошибка при работе с объектом TupleData")

P.S. Данная программа должна быть выполнена штатно, без ошибок. На экран отображать ничего не нужно.

# здесь объявляйте классы CellException, CellIntegerException, CellFloatException, CellStringException

# здесь объявляйте классы CellInteger, CellFloat, CellString

# здесь объявляйте класс TupleData

# эти строчки в программе не менять!

ld = TupleData(CellInteger(0, 10), CellInteger(11, 20), CellFloat(-10, 10), CellString(1, 100))

try:

ld[0] = 1

ld[1] = 20

ld[2] = -5.6

ld[3] = "Python ООП"

except CellIntegerException as e:

print(e)

except CellFloatException as e:

print(e)

except CellStringException as e:

print(e)

except CellException:

print("Ошибка при обращении к ячейке")

except Exception:

print("Общая ошибка при работе с объектом TupleData")