

```

1  """Урок 8 Задание 3
2  Создайте собственный класс-исключение, который должен
3  проверять содержимое списка на
4  наличие только чисел. Проверить работу исключения на
5  реальном примере. Необходимо
6  запрашивать у пользователя данные и заполнять список
7  только числами. Класс-исключение
8  должен контролировать типы данных элементов списка.
9  Примечание: длина списка не фиксирована. Элементы
10 запрашиваются бесконечно, пока
11 пользователь сам не остановит работу скрипта, введя,
12 например, команду "stop". При этом
13 скрипт завершается, сформированный список с числами
14 выводится на экран.
15 Подсказка: для данного задания примем, что
16 пользователь может вводить только числа и
17 строки. При вводе пользователем очередного элемента
18 необходимо реализовать проверку
19 типа элемента и вносить его в список, только если
20 введено число. Класс-исключение должен
21 не позволить пользователю ввести текст (не число) и
22 отобразить соответствующее
23 сообщение. При этом работа скрипта не должна
24 завершаться.
25 """
26
27 class Error:
28     def __init__(self, *args):
29         self.my_list = []
30
31     def my_input(self):
32
33         while True:
34             try:
35                 val = int(input('Введите значения и
36 нажимайте Enter - '))
37                 self.my_list.append(val)
38                 print(f'Текущий список - {self.
39 my_list} \n ')
40             except:
41                 print(f"Недопустимое значение -
42 строка и булево")
43                 y_or_n = input(f'Попробовать еще раз
44 ? Y/N ')

```

```
30
31         if y_or_n == 'Y' or y_or_n == 'y':
32             print(try_except.my_input())
33         elif y_or_n == 'N' or y_or_n == 'n':
34             return f'Вы вышли'
35         else:
36             return f'Вы вышли'
37
38
39 try_except = Error(1)
40 print(try_except.my_input())
```