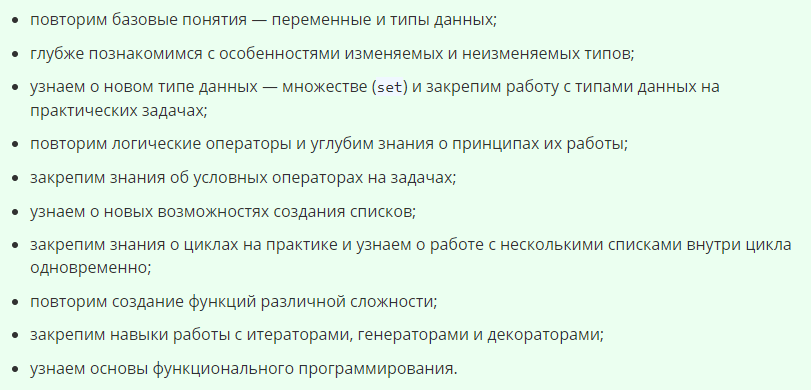
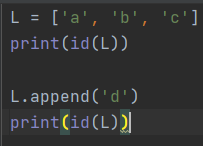
**B5**

В этом модуле будут практические задачи по пройденным темам. Посмотрим.



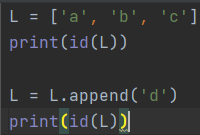
**B5.2. Переменные и типы данных**

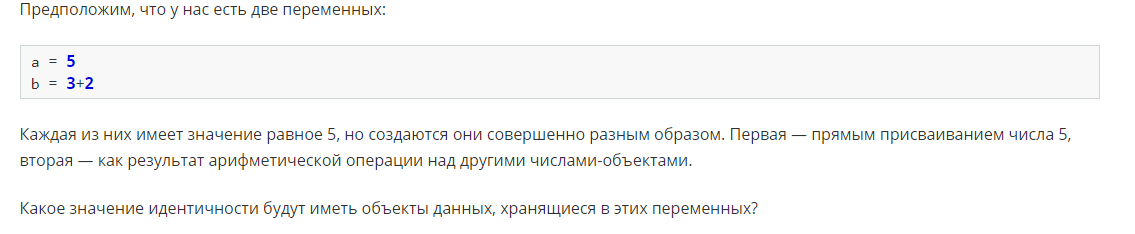


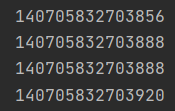
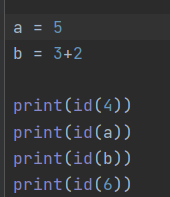
Напоминаю, что есть изменяемые и не изменяемые типы данных. Вот пример списка который можно изменить, но его id не изменится.



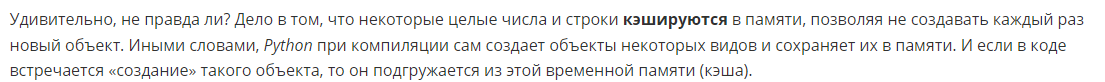
Но если произвести перезапись, то и ID изменится.





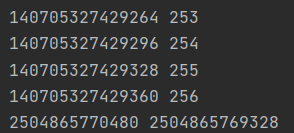
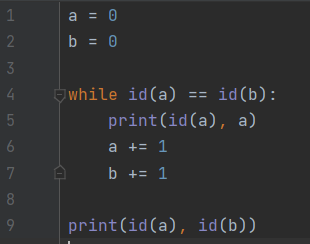


Я был уверен, что id будут разными, но нет. Не важно как сформировалась переменная. Важно что в ней хранится. Кстати, id целых чисел меняется через 2 и какой-то десяток...

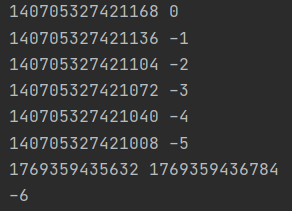
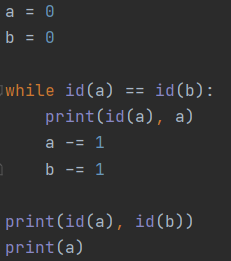


Понятно. Я этого не знал.

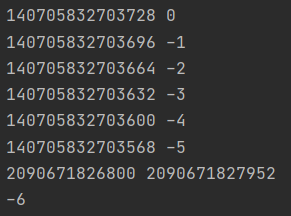
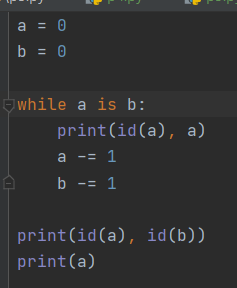


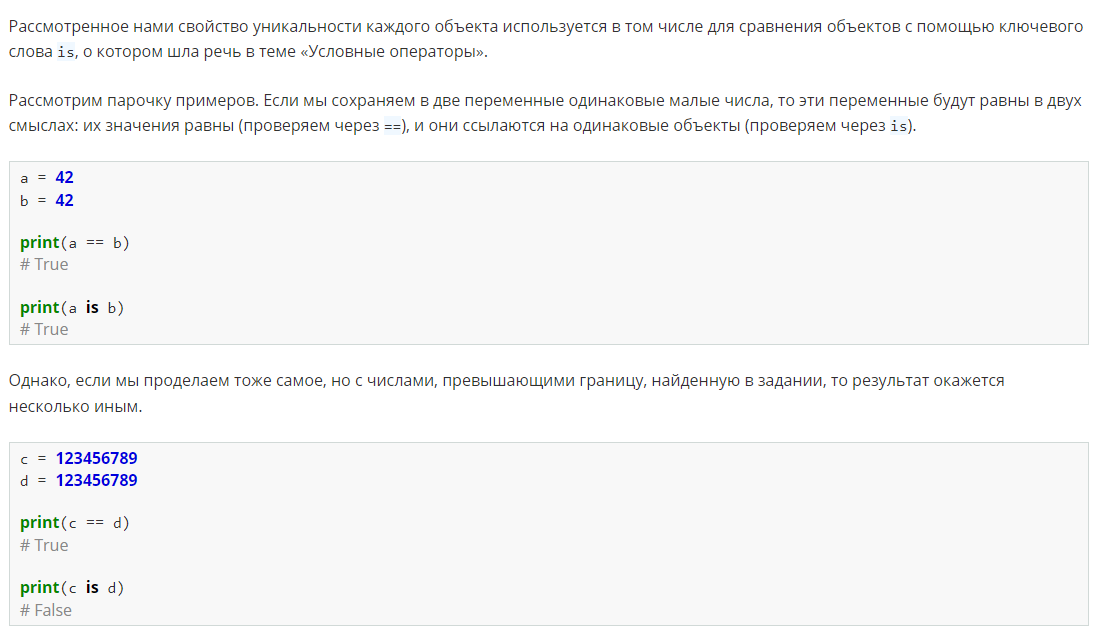


Таким образом Python может закэшировать до 257 переменных. Отрицательных чисел python хранит намного меньше.



К этому относится и сравнение чисел через **is**

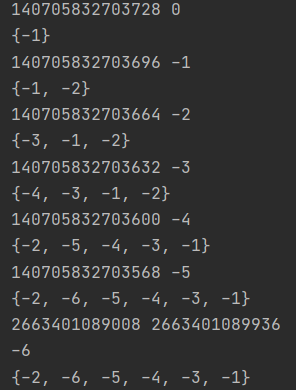
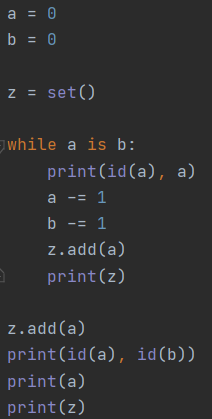




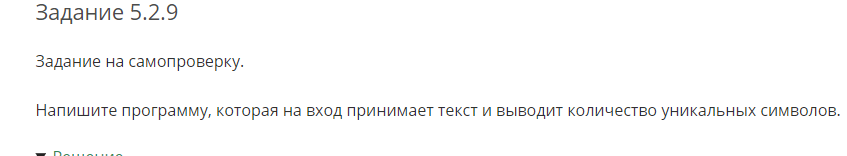
**Про неизменяемость кортежей.** Да, кортеж после создания изменить нельзя. Но если внутри кортежа есть переменные с изменяемым типом данных, то эти переменные можно свободно изменять, так как от этого их уникальный id изменится.

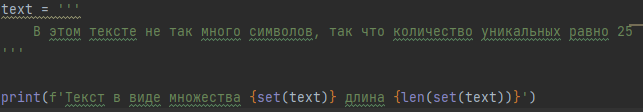
Просто напомню про еще один изменяемы тип данных, множества – **set.**

Суть в том, что во множестве не может быть элементов с одним и тем же **ID.** Что значит во множестве могут хранится только уникальные значения. Я тут подумал, а можно ли это попробовать обойти. Наверно все же речь не об **id.** А именно в значениях. Иначе условные -7, -8, -7 можно было бы туда засунуть, попробую.



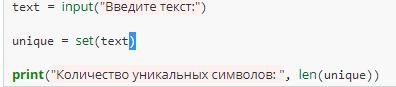
Да! Множества не зависят от **id.** Для них важны именно значения. Думаю два одинаковых списка туда тоже не добавить. Поправка. Один тоже не добавить. Множества вообще не поддерживают изменяемые типы данных, похоже.







Их вариант.



Тоже самое, только я забыл про ввод текста.

Хм…







Первый вариант это текст в программе через ‘’’ ‘’’. А второй вариант, это скопированный в консоль текст. И я кажется уже понял в чем дело. Секунду.

Да.. в консоль скопировалась только первая строка.



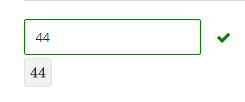
Чтобы вставить в консоль текст без пробелов, я удалил все переносы в ручную в notepade



Но тем не менее ответ я все равно получил 45



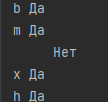
Хотя **skillfactory считают что ответ 44.**

****

И сейчас я выясню, в чем моя ошибка.



Кажется проблема в символе:



И этот символ был скрыт в самом конце.



/t это кстати, табуляция.

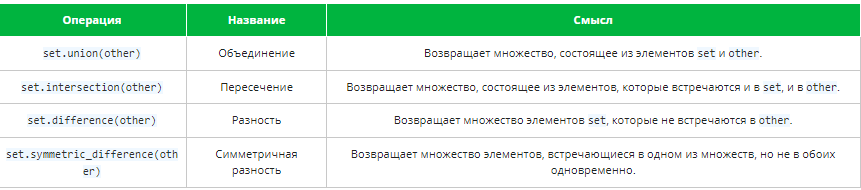
Блин, из-за проблемы с авто, думается не очень. Ладно дальше. У множества есть еще и свои методы.

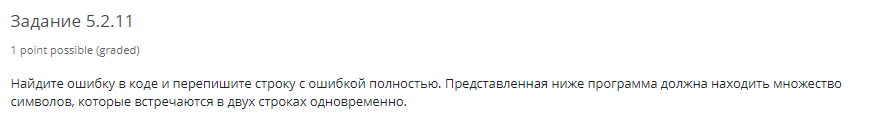
**Set.uninon(other) –** объединение двух множеств.

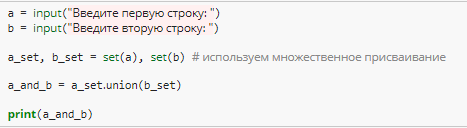
**Set.intersection(other) –** Возвращает пересечение двух множеств.

**Set.difference(other) –** возвращает разность двух множеств

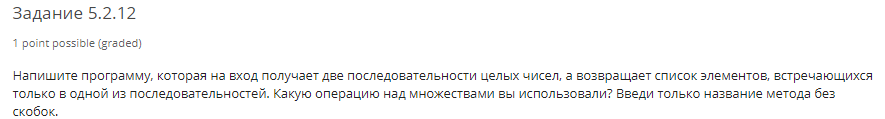
**Set.symmetric\_difference(other) –** Возварщает множество элементов, встречающиеся в одном из множеств, но не в обоих одновременно.

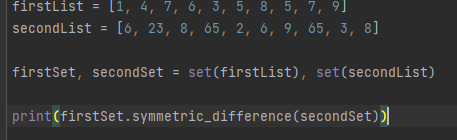






**a\_and\_b = a\_set.intersection(b\_set)**

****

****

****

**B5.3. Условия и циклы**

Далее.