

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.7
дисциплины «Основы кроссплатформенного программирования»
Вариант ____

Выполнил:
Дмитриченко Марат Заурович
1 курс, группа ИТС-б-0-22-1,
11.03.02 «Инфокоммуникационные
технологии и системы связи»,
направленность (профиль)
«Инфокоммуникационные системы и
сети», очная форма обучения

(подпись)

Руководитель практики:
Воронкин Р. А., доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

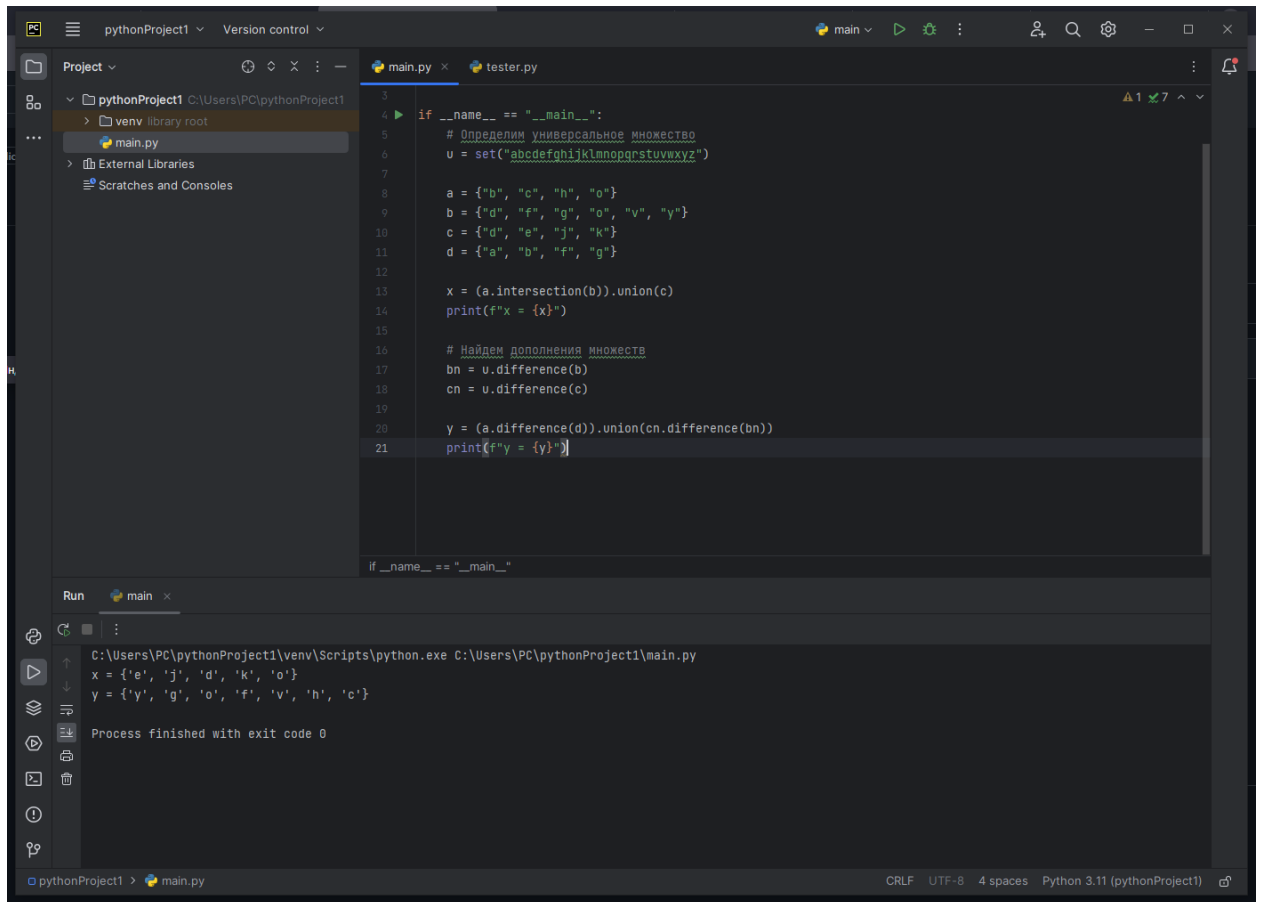
Ставрополь, 2023 г.

Тема: Работа с множествами в языке Python

Цель: приобретение навыков по работе с множествами при написании программ с помощью языка программирования Python версии 3.x.

Ход работы

Условие примера: определить результат выполнения операций над множествами. Считать элементы множества строками.



```
3
4 if __name__ == "__main__":
5     # Определим универсальное множество
6     u = set("abcdefghijklmnopqrstuvwxyz")
7
8     a = {"b", "c", "h", "o"}
9     b = {"d", "f", "g", "o", "v", "y"}
10    c = {"d", "e", "j", "k"}
11    d = {"a", "b", "f", "g"}
12
13    x = (a.intersection(b)).union(c)
14    print(f"x = {x}")
15
16    # Найдем дополнения множеств
17    bn = u.difference(b)
18    cn = u.difference(c)
19
20    y = (a.difference(d)).union(cn.difference(bn))
21    print(f"y = {y}")
22
23 if __name__ == "__main__":
```

Run main x

```
C:\Users\PC\pythonProject1\venv\Scripts\python.exe C:\Users\PC\pythonProject1\main.py
x = {'e', 'j', 'd', 'k', 'o'}
y = {'y', 'g', 'o', 'f', 'v', 'h', 'c'}
Process finished with exit code 0
```

pythonProject1 > main.py CRLF UTF-8 4 spaces Python 3.11 (pythonProject1)

Рисунок 1. Реализация примера

Условие примера: подсчитайте количество гласных в строке, введенной с клавиатуры с использованием множеств.

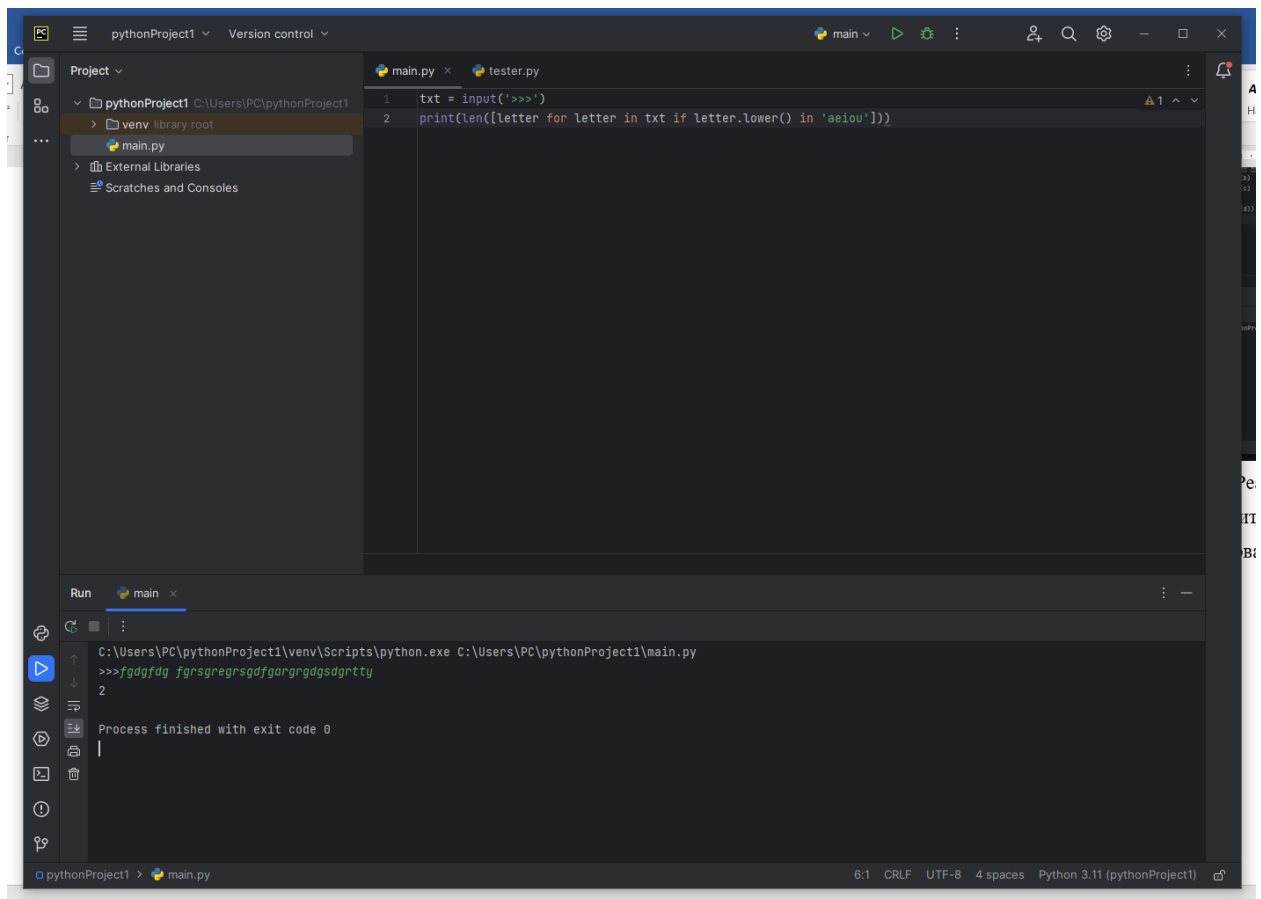


Рисунок 2. Реализация общего задания 1

Условие примера: определите общие символы в двух строках, введенных с клавиатуры.

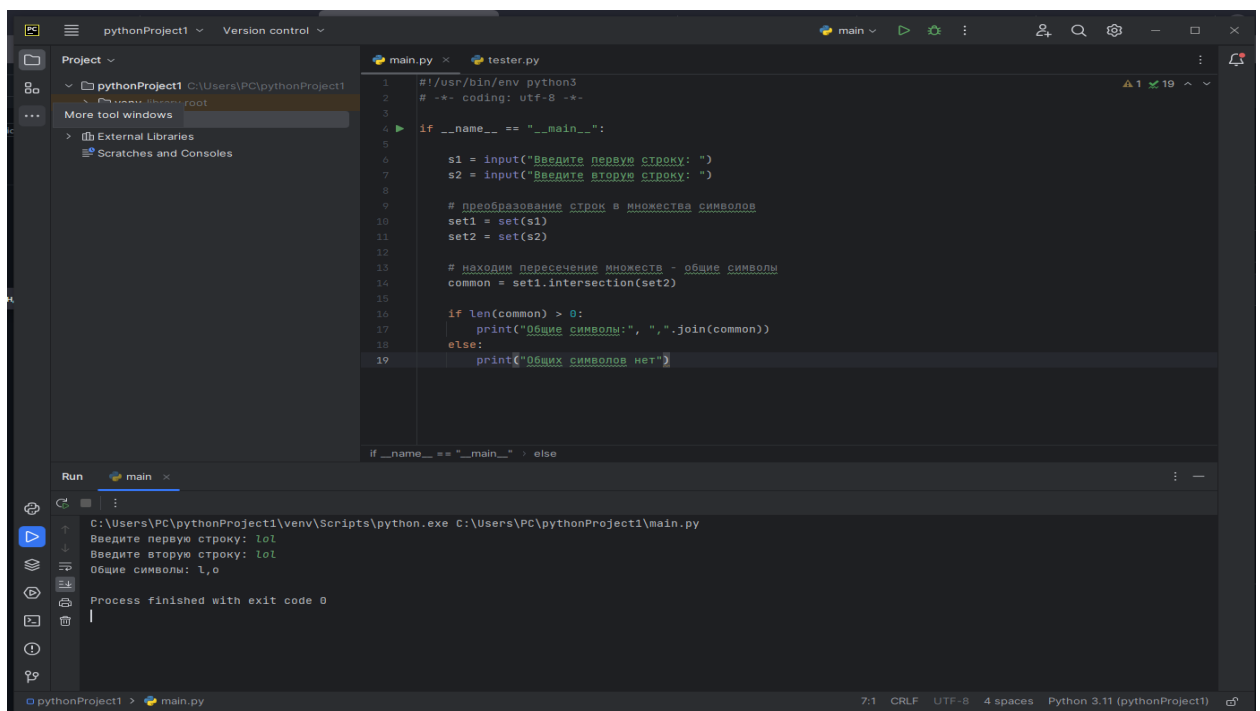


Рисунок 3. Реализация общего задания 2

Индивидуальное задание

Определить результат выполнения операций над множествами. Считать элементы множества строками. Проверить результаты вручную. Номер варианта задания необходимо получить у преподавателя.

5.
$$X = (A/B) \cap (C \cup D); \quad Y = (A \cap \bar{B}) \cup (C/D). \quad (5)$$

$$A = \{a, d, k, l, o, s\}; \quad B = \{d, e, k, s, u, x\}; \quad C = \{o, p, w\}; \quad D = \{d, n, r, y, z\}; \quad (7)$$

6.
$$X = (A/B) \cup (C \cap D); \quad Y = (\bar{A} \cap \bar{B}) / (C \cup D).$$

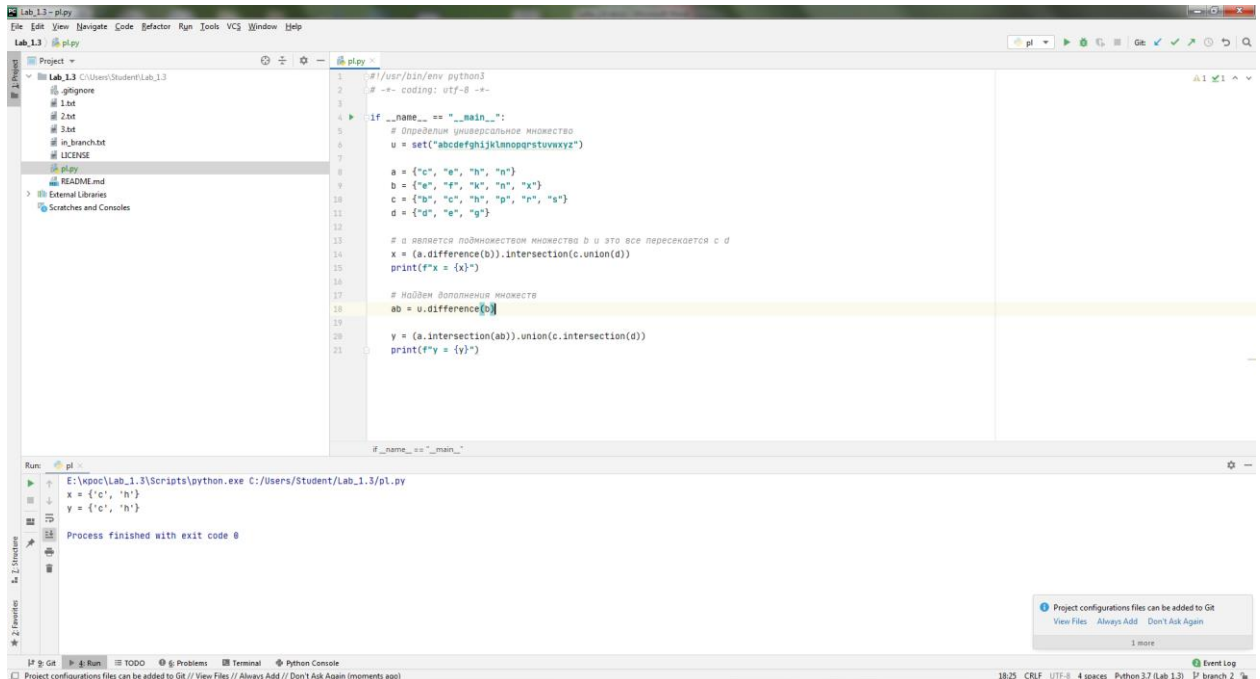


Рисунок 4. Реализация индивидуального задания.

Ответы на контрольные вопросы

1. Что такое множества в языке Python?

Множества в языке Python - это неупорядоченные коллекции уникальных элементов, где каждый элемент может присутствовать только один раз. В качестве элементов этого набора данных могут выступать любые неизменяемые объекты, такие как числа, символы, строки.

2. Как осуществляется создание множеств в Python?

Создание множества осуществляется с помощью фигурных скобок {} или функции set(). Например, a = {1, 2, 3} или b = set([4, 5, 6]).

3. Как проверить присутствие/отсутствие элемента в множестве?

Для проверки присутствия/отсутствия элемента в множестве в Python используется ключевое слово in.

4. Как выполнить перебор элементов множества?

Перебор элементов множества можно выполнить с помощью цикла `for`.

5. Что такое `set comprehension`?

`Set comprehension` (генератор множеств) - это способ создания и заполнения множества в Python одной строки кода.

6. Как выполнить добавление элемента во множество?

Для добавления элемента во множество в Python используется метод `add()`.

7. Как выполнить удаление одного или всех элементов множества?

Удаление элемента из множества можно выполнить с помощью метода `remove()`, который удаляет заданный элемент, или метода `discard()`, который также удаляет элемент, но не вызывает исключение, если элемент не найден. Удаление всех элементов множества может быть выполнено с помощью метода `clear()`.

8. Как выполняются основные операции над множествами: объединение, пересечение, разность?

Основные операции над множествами в Python:

1. Объединение (`union`) - создает новое множество, содержащее все элементы из двух или более множеств.

2. Пересечение (`intersection`) - создает новое множество, содержащее только те элементы, которые присутствуют в обоих множествах.

3. Разность (`difference`) - создает новое множество, состоящее из элементов первого множества, которые не присутствуют во втором множестве.

9. Как определить, что некоторое множество является надмножеством или подмножеством другого множества?

Для определения, является ли некоторое множество подмножеством или надмножеством другого множества в Python, можно воспользоваться методами `issubset()` и `issuperset()`. Метод `issubset()` проверяет, является ли одно множество подмножеством другого. Если все элементы первого множества присутствуют во втором множестве, то метод вернет `True`, иначе

False. Метод `issuperset()` проверяет, является ли одно множество надмножеством другого. Если все элементы второго множества присутствуют в первом множестве, то метод вернет `True`, иначе `False`.

10. Каково назначение множеств `frozenset`?

Множества `frozenset` являются неизменяемыми (`immutable`) множествами, то есть их элементы нельзя добавлять, удалять или изменять после создания. Они используются, когда требуется иметь неизменяемое множество, которое можно использовать как элемент словаря или ключ в другом множестве.

11. Как осуществляется преобразование множеств в строку, список, словарь?

Для преобразования множества в строку, список или словарь в языке Python можно использовать следующие функции: - `str()` - для преобразования множества в строку - `list()` - для преобразования множества в список - `dict.fromkeys()` - для преобразования множества в словарь

Вывод: приобрел навыки по работе с множествами при написании программ с помощью языка программирования Python версии 3.x.