

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 «Программирование на Python»

Вариант 15

Выполнил:
Кенесбаев Хилол Куат улы
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», 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.

Порядок выполнения работы:

Проработал примеры лабораторной работы:

```
1
2  if __name__ == "__main__":
3      # Определим универсальное множество
4      u = set("abcdefghijklmnopqrstuvwxyz")
5      a = {"b", "c", "h", "o"}
6      b = {"d", "f", "g", "o", "v", "y"}
7      c = {"d", "e", "j", "k"}
8      d = {"a", "b", "f", "g"}
9      x = (a.intersection(b)).union(c)
10     print(f"x = {x}")
11     # Найдем дополнения множеств
12     bn = u.difference(b)
13     cn = u.difference(c)
14     y = (a.difference(d)).union(cn.difference(bn))
15     print(f"y = {y}")
```

Рисунок 1 primer.py

```
C:\Users\HP-PC\AppData\Local\Programs\Python\Python312\python.exe C:\Users\HP-PC\PycharmProjects\lab-10\primer.py
x = {'j', 'd', 'k', 'e', 'o'}
y = {'g', 'y', 'o', 'h', 'f', 'v', 'c'}

Process finished with exit code 0
```

Рисунок 2 Результат работы программы primer.py

1. Решил задачу 1: подсчитайте количество гласных в строке, введенной с клавиатуры с использованием множеств.

Код программы:

```
#!/usr/bin/env
python3 # -*-
coding: utf-8 -*-
if __name__ == "__main__":
    vowels_all = {'a', 'e', 'i', 'o', 'u', 'y', 'a', 'e', 'ё', 'и', 'o', 'y', 'ы', 'э', 'ю', 'я'}
    string = input("Введите строку: ")
    count = len([i for i in string.lower() if i in vowels_all])
    print("Количество гласных в строке:", count)
```

```
C:\Users\HP-PC\AppData\Local\Programs\Python\Python312\python.exe C:\Users\HP-PC\PycharmProjects\lab-10\task1.py
Введите строку: лабораторная работа по Python
Количество гласных в строке: 12

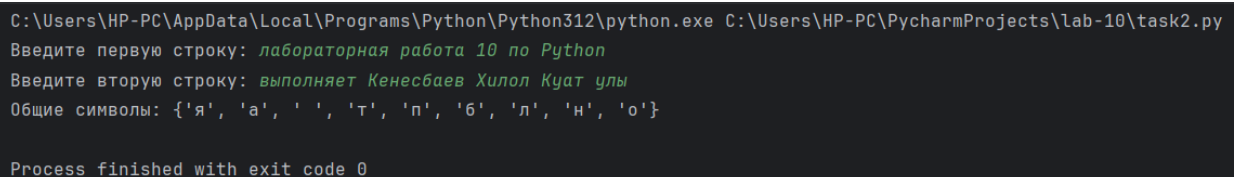
Process finished with exit code 0
```

Рисунок 3 Результат работы программы task1.py

2. Решил задачу 2: определите общие символы в двух строках, введенных с клавиатуры.

Код программы:

```
#!/usr/bin/env
python3 # -*-
coding: utf-8 -*-
if __name__ == "__main__":
    string1 = input("Введите первую строку: ")
    string2 = input("Введите вторую строку: ")
    set1 = set(string1)
    set2 = set(string2)
    common_characters = set1.intersection(set2)
    print("Общие символы:", common_characters)
```



```
C:\Users\HP-PC\AppData\Local\Programs\Python\Python312\python.exe C:\Users\HP-PC\PycharmProjects\lab-10\task2.py
Введите первую строку: лабораторная работа 10 по Python
Введите вторую строку: выполняет Кенесбаев Хилол Куат улы
Общие символы: {'я', 'а', ' ', 'т', 'п', 'б', 'л', 'н', 'о'}
Process finished with exit code 0
```


Рисунок 4 Результат работы программы task2.py

Выполнил индивидуальное задание:

$$A = \{b, d, f, g, l, u\}; \quad B = \{d, e, f, m, n, z\}; \quad C = \{h, i, r, x, y\}; \quad D = \{a, e, f, k, r, s, x\};$$
$$X = (A/B) \cap (C \cup D); \quad Y = (\bar{A} \cap D) \cup (C/B).$$

Код программы:

```
if name == "main":
    # Определим универсальное множество
    u = set("abcdefghijklmnopqrstuvwxyz")
    A = {"b", "d", "f", "g", "l", "u"}
    B = {"d", "e", "f", "m", "n", "z"}
    C = {"h", "i", "r", "x", "y"}
    D = {"a", "e", "f", "k", "r", "s", "x"}
    x = (A.intersection(B)).union(D)
    print(f"x = {x}")
    # дополнения множеств
    an = u.difference(A)
    bn = u.difference(B)
    y = (an.intersection(bn)).difference(C.union(D))
    print(f"y = {y}")
```



```
C:\Users\HP-PC\AppData\Local\Programs\Python\Python312\python.exe C:\Users\HP-PC\Desktop\study\python\непесдача\Lab2.7\код\individualное.py
x = {'a', 'd', 'r', 's', 'e', 'f', 'k', 'x'}
y = {'q', 'p', 'c', 'o', 't', 'w', 'v', 'j'}
```

Рисунок 5 Результат работы программы individualное.py

Ответы на контрольные вопросы:

1. Что такое множества в языке Python?

Множеством в языке программирования Python называется неупорядоченная совокупность уникальных значений.

2. Как осуществляется создание множеств в Python?

Создать множество можно, просто присвоив переменной последовательность значений, выделив их фигурными скобками или с помощью метода `set`.

3. Как проверить присутствие/отсутствие элемента в множестве?

Для проверки присутствия используется `in`, для проверки отсутствия `not in`.

4. Как выполнить перебор элементов множества?

`for a in set: print(a)`.

5. Что такое `set comprehension`?

Генератор, позволяющий заполнять списки, а также другие наборы данных с учетом неких условий.

6. Как выполнить добавление элемента во множество?

Чтобы внести новые значения, потребуется вызывать метод `add`. Аргументом в данном случае будет добавляемый элемент последовательности.

7. Как выполнить удаление одного или всех элементов множества?

Для удаления элементов из множества используются следующие функции в Python:

`remove` – удаление элемента с генерацией исключения в случае, если такого элемента нет;

`discard` – удаление элемента без генерации исключения, если элемент

отсутствует;

pop – удаление первого элемента, генерируется исключение при попытке удаления из пустого множества.

Иногда необходимо полностью убрать все элементы. Чтобы не удалять каждый элемент отдельно, используется метод `clear()`, не принимающий аргументов.

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

Чтобы объединить все элементы двух разных множеств, стоит воспользоваться методом `union()` на одном из объектов.

Чтобы найти общие элементы для двух разных множеств, следует применить функцию `intersection()`, принимающую в качестве аргумента один из наборов данных.

Чтобы вычислить разность для двух разных множеств, необходимо воспользоваться методом `difference()`.

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

Чтобы выяснить, является ли множество `a` подмножеством `b`, стоит попробовать вывести на экран результат выполнения метода `issubset`.

Чтобы узнать, является ли множество `a` надмножеством `b`, необходимо вызвать метод `issuperset` и вывести результат его работы на экран.

10. Каково назначение множеств `frozenset`?

Множество, содержимое которого не поддается изменению имеет тип `frozenset`. Значения из этого набора нельзя удалить, как и добавить новые.

Поскольку содержимое `frozenset` должно всегда оставаться статичным,

перечень функций, с которыми такое множество может взаимодействовать, имеет ограничения.

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

Для преобразования множества в строку используется конкатенация текстовых значений, которую обеспечивает функция `join()`. В этом случае ее аргументом является набор данных в виде нескольких строк.

Чтобы получить из множества словарь, следует передать функции `dict` набор из нескольких пар значений, в каждом из которых будет находиться ключ.

По аналогии с предыдущими преобразованиями можно получить список неких объектов. На этот раз используется вызов `list`, получающий в качестве аргумента множество `a`.

Вывод: в результате выполнения работы были приобретены навыки по работе с множествами при написании программ с помощью языка программирования Python версии 3.x.