

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего
образования
«СЕВЕРОКАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ
УНИВЕРСИТЕТ»**

**Кафедра
инфокоммуникаций
Институт цифрового
развития**

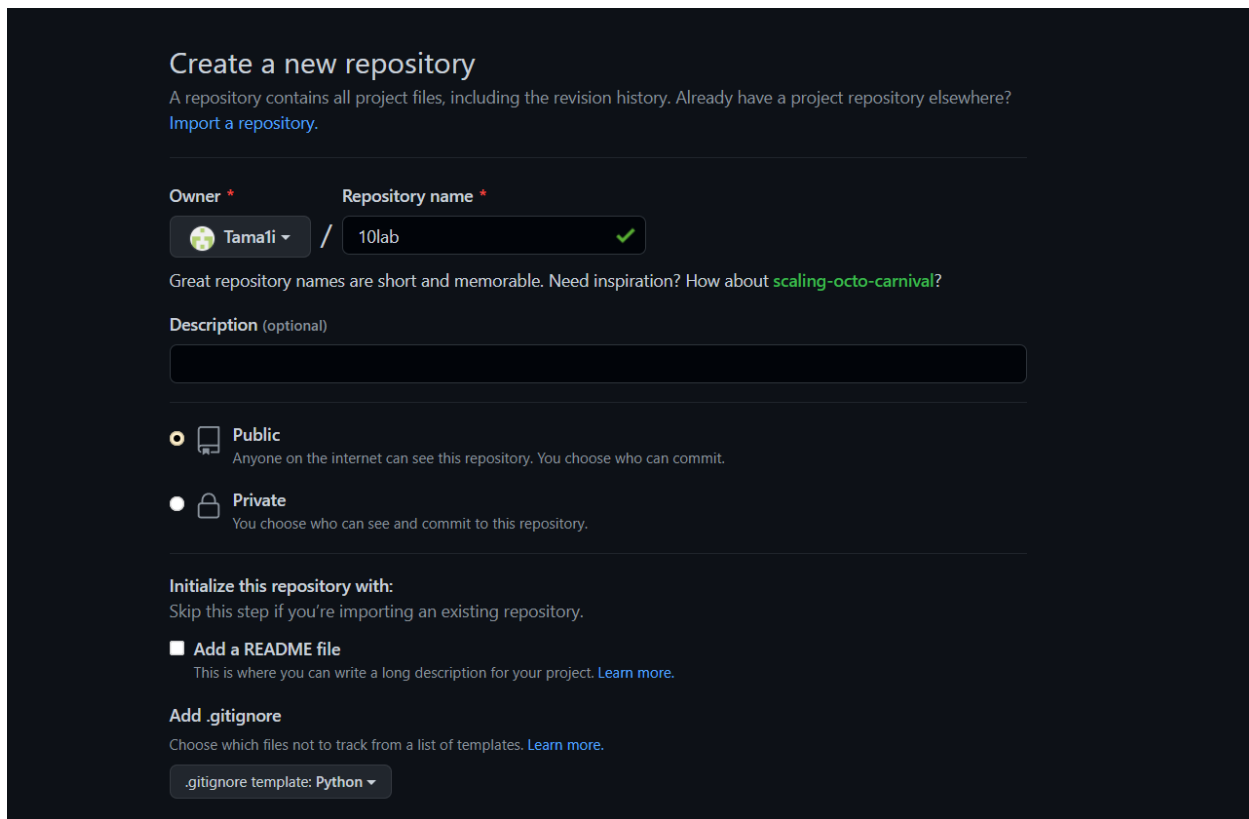
**ОТЧЁТ
по лабораторной работе №10**
Дисциплина: «Основы программной инженерии»
Тема: «Работа с множествами в языке Python»

Выполнил: студент 2
курса группы Пиж-б-о-
21-1
Рязанцев Матвей
Денисович

Ставрополь 2022

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

Выполнение работы



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner * Repository name *

Tama1i / 10lab

Great repository names are short and memorable. Need inspiration? How about [scaling-octo-carnival?](#)

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ Add a README file
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: Python

Рисунок 1 -создание репозитория

```
D:\gite\10lab>git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [D:/gite/10lab/.git/hooks]
```

Рисунок 2 - Организация репозитория по модели ветвления git flow

Код общего задания:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
```

```

if __name__ == "__main__":
    # Определим универсальное множество
    u = set("abcdefghijklmnopqrstuvwxyz")

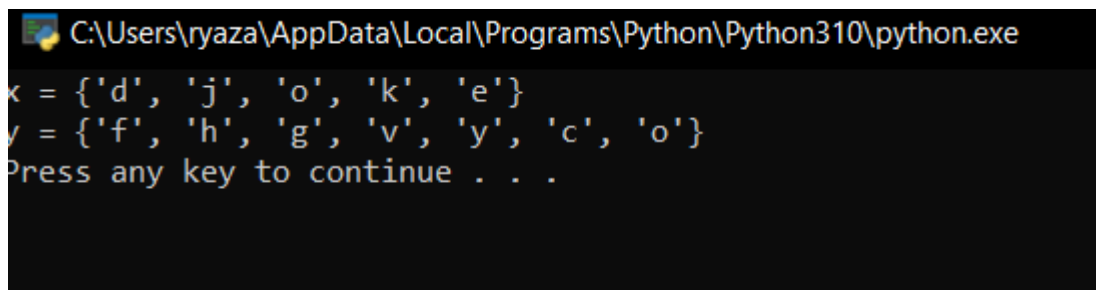
    a = {"b", "c", "h", "o"}
    b = {"d", "f", "g", "o", "v", "y"}
    c = {"d", "e", "j", "k"}
    d = {"a", "b", "f", "g"}

    x = (a.intersection(b)).union(c)
    print(f"x = {x}")

    # Найдем дополнения множеств
    bn = u.difference(b)
    cn = u.difference(c)

    y = (a.difference(d)).union(cn.difference(bn))
    print(f"y = {y}")

```

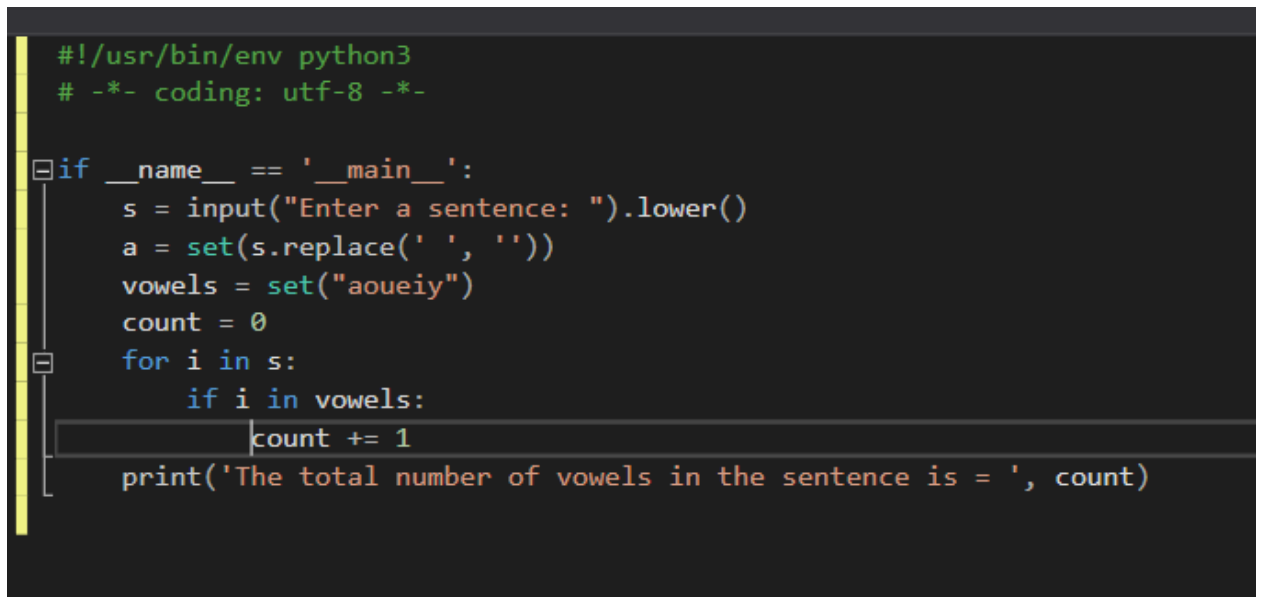


```

C:\Users\ryaza\AppData\Local\Programs\Python\Python310\python.exe
x = {'d', 'j', 'o', 'k', 'e'}
y = {'f', 'h', 'g', 'v', 'y', 'c', 'o'}
Press any key to continue . . .

```

Рисунок 1 – результат выполнения программы



```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-

if __name__ == '__main__':
    s = input("Enter a sentence: ").lower()
    a = set(s.replace(' ', ''))
    vowels = set("aoueiy")
    count = 0
    for i in s:
        if i in vowels:
            count += 1
    print('The total number of vowels in the sentence is = ', count)

```

Рисунок 2 – код программы задание 1

```
C:\Users\ryaza\AppData\Local\Programs\Python\Python310\python.exe
Enter a sentence: aegwegareh
The total number of vowels in the sentence is = 5
Press any key to continue . . .
```

Рисунок 3 – результат работы программы

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

if __name__ == "__main__":
    first_string = set(input("Введите первую строку: ").lower())
    second_string = set(input("Введите вторую строку: ").lower())
    print(first_string & second_string)
```

Рисунок 4 – код программы задание 2

```
C:\Users\ryaza\AppData\Local\Programs\Python\Python310\python.exe
Введите первую строку: agsjkngl
Введите вторую строку: aegjnf
{'a', 'g', 'j', 'n'}
Press any key to continue . . .
```

Рисунок 5 – результат работы программы

Индивидуальное задание

Код программы:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
if __name__ == "__main__":
    # Определим универсальное множество
```

```

u = set("abcdefghijklmnopqrstuvwxyz")

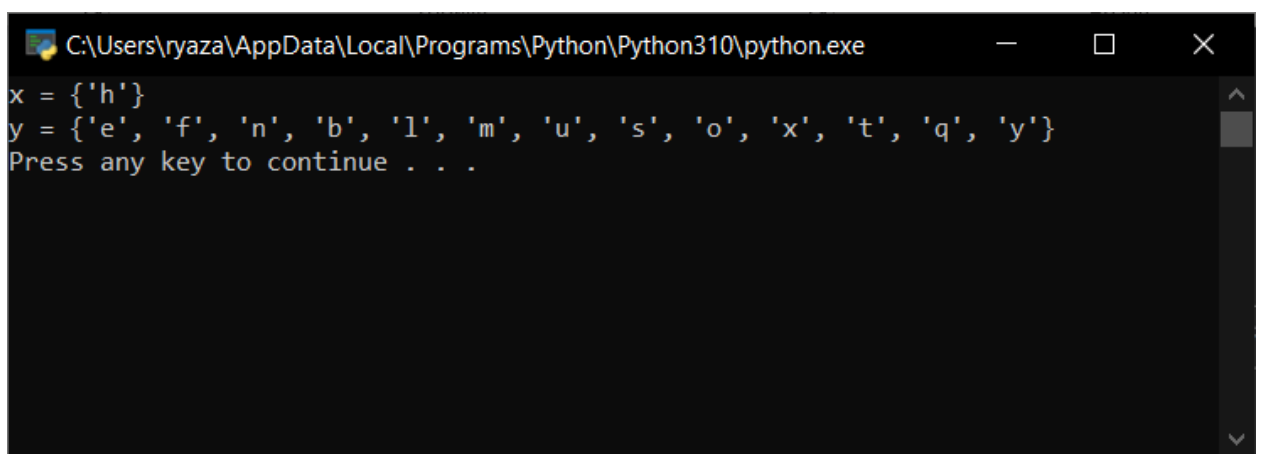
a = {"a", "h", "k"}
b = {"c", "d", "h", "p", "r"}
c = {"h", "i", "z"}
d = {"c", "g", "j", "v", "w"}

x = (a.union(b)).intersection(c)
print(f"x = {x}")

# Найдем дополнения множеств
an = u.difference(a)
bn = u.difference(b)

y = (an.intersection(bn)).difference(c.union(d))
print(f"y = {y}")

```



```

C:\Users\ryaza\AppData\Local\Programs\Python\Python310\python.exe
x = {'h'}
y = {'e', 'f', 'n', 'b', 'l', 'm', 'u', 's', 'o', 'x', 't', 'q', 'y'}
Press any key to continue . . .

```

Рисунок 6 – результат работы программы идз

Контрольные вопросы

1. Что такое множества в языке Python? 5 Множеством в языке программирования Python называется неупорядоченная совокупность уникальных значений.
2. Как осуществляется создание множеств в Python? $a = \{1, 2, 0, 1, 3, 2\}$
 $a = \text{set}('data')$
3. Как проверить присутствие/отсутствие элемента в множестве? $a = \{0, 1, 2, 3\}$ $\text{print}(2 \text{ in } a) \rightarrow \text{True}$

4. Как выполнить перебор элементов множества? `for a in {0, 1, 2}: print(a)`
5. Что такое set comprehension? `a = {i for i in [1, 2, 0, 1, 3, 2]}`
6. Как выполнить добавление элемента во множество? `a = {0, 1, 2, 3}`
`a.add(4)`
7. Как выполнить удаление одного или всех элементов множества?
Удаление одного элемента: `a = {0, 1, 2, 3}` `a.remove(3)` Удаление всех элементов множества: `a.clear()`
8. Как выполняются основные операции над множествами: объединение, пересечение, разность? объединение - `a.union(b)` или `a | b` пересечение - `a.intersection(b)` или `a & b` разность - `a.difference(b)` или `a - b`
9. Как определить, что некоторое множество является надмножеством или подмножеством другого множества? Подмножество – `a.issubset(b)`
Надмножество – `a.issuperset(b)`
10. Каково назначение множеств `frozenset`? Множество, содержимое которого не поддается изменению
11. Как осуществляется преобразование множеств в строку, список, словарь? Строка – `a = {'set', 'str', 'dict', 'list'}` `b = ','.join(a)` Словарь – `a = {'a', 2}, ('b', 4)}` `b = dict(a)` Список – `a = {1, 2, 0, 1, 3, 2}` `b = list(a)`