

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ**

**Отчет о лабораторной работе №10 по дисциплине основы программной
инженерии**

Выполнил:

Выходцев Егор Дмитриевич,
2 курс, группа ПИЖ-б-о-20-1,

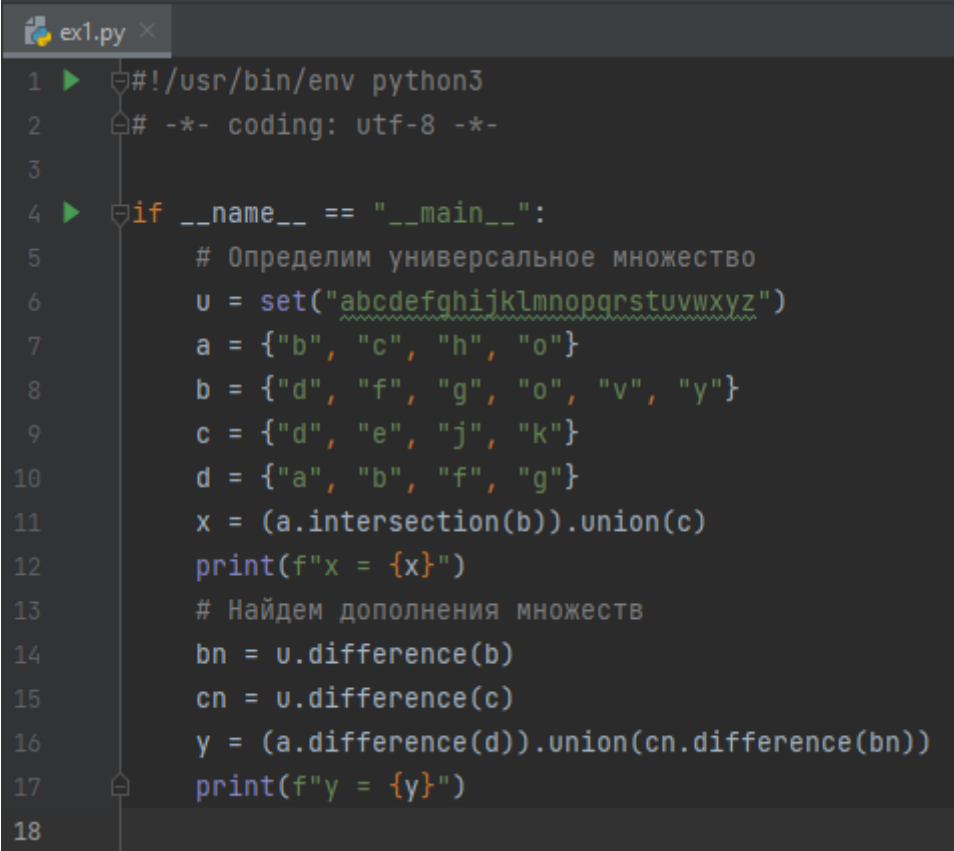
Проверил:

Доцент кафедры инфокоммуникаций,
Воронкин Р.А.

Ставрополь, 2021 г.

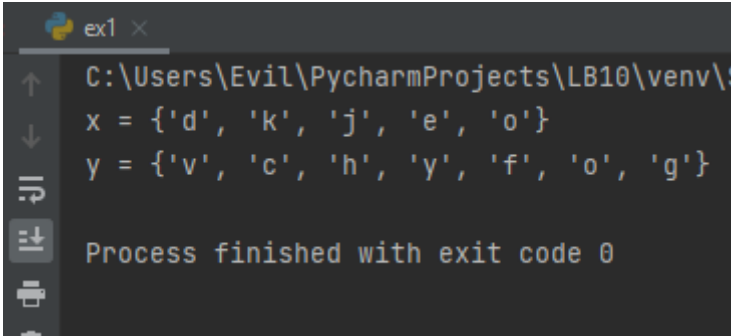
1. Работа с множествами в языке Python

1.1 Пример 1 (рис. 1, 2).



```
ex1.py x
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  if __name__ == "__main__":
5      # Определим универсальное множество
6      u = set("abcdefghijklmnopqrstuvwxyz")
7      a = {"b", "c", "h", "o"}
8      b = {"d", "f", "g", "o", "v", "y"}
9      c = {"d", "e", "j", "k"}
10     d = {"a", "b", "f", "g"}
11     x = (a.intersection(b)).union(c)
12     print(f"x = {x}")
13     # Найдем дополнения множеств
14     bn = u.difference(b)
15     cn = u.difference(c)
16     y = (a.difference(d)).union(cn.difference(bn))
17     print(f"y = {y}")
18
```

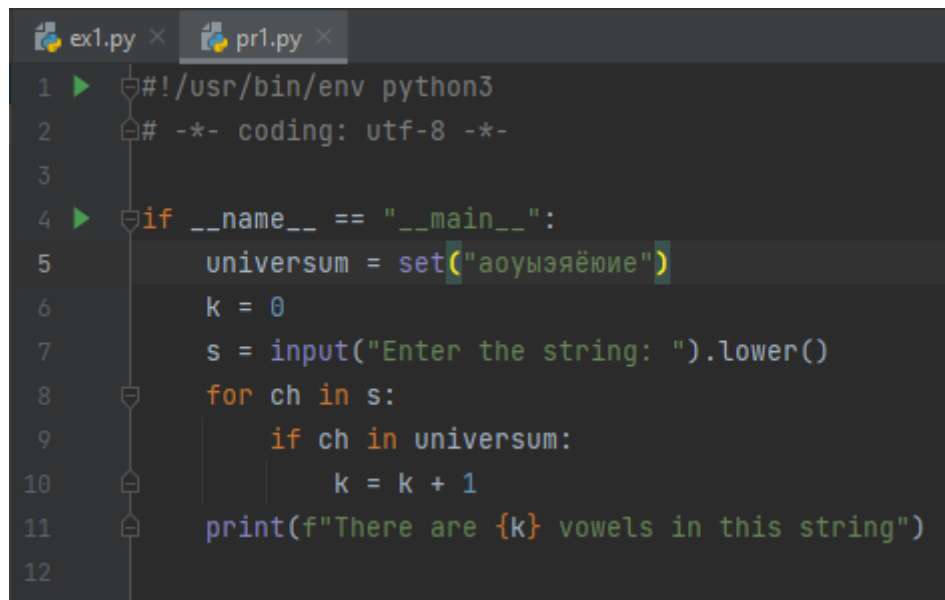
Рисунок 1 – Код примера



```
ex1 x
C:\Users\Evil\PycharmProjects\LB10\venv\Scripts\python.exe
x = {'d', 'k', 'j', 'e', 'o'}
y = {'v', 'c', 'h', 'y', 'f', 'o', 'g'}
Process finished with exit code 0
```

Рисунок 2 – Результат выполнения программы

1.2 Задание №1 (рис 3, 4 ,5).

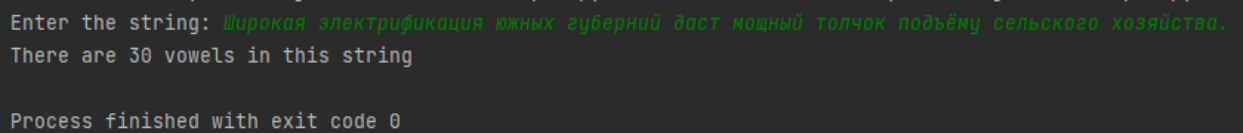


```

1  ▶  #!/usr/bin/env python3
2      # -*- coding: utf-8 -*-
3
4  ▶  if __name__ == "__main__":
5      universum = set("аоуыэяёюие")
6      k = 0
7      s = input("Enter the string: ").lower()
8      for ch in s:
9          if ch in universum:
10             k = k + 1
11      print(f"There are {k} vowels in this string")
12

```

Рисунок 3 – Код программы



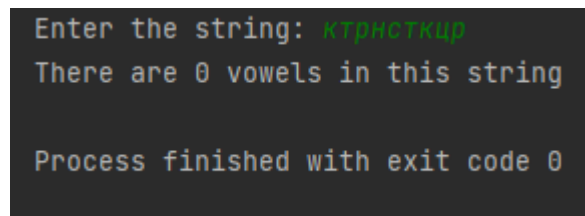
```

Enter the string: Широкая электрификация южных губерний даст мощный толчок подъёму сельского хозяйства.
There are 30 vowels in this string

Process finished with exit code 0

```

Рисунок 4 – Пример выполнения программы



```

Enter the string: ктрнсткцр
There are 0 vowels in this string

Process finished with exit code 0

```

Рисунок 5 – Вывод программы при отсутствии гласных букв

1.3 Задание №3 (рис. 6, 7).

```
ex1.py x pr1.py x pr2.py x
1 ▶ #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 ▶ if __name__ == "__main__":
5     a = set(i for i in input("Enter the first string: ").lower())
6     b = set(i for i in input("Enter the second string: ").lower())
7     print(f"The intersection of these strings is: {a.intersection(b)}")
8
```

Рисунок 6 – Код программы

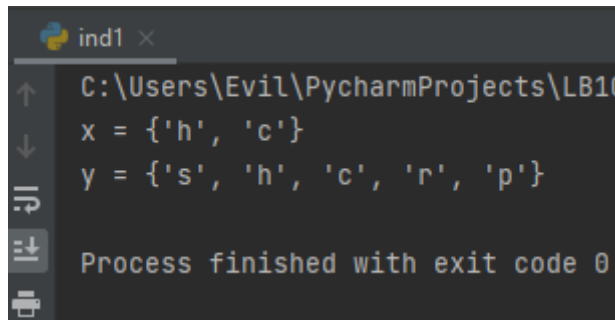
```
Enter the first string: французские
Enter the second string: электрификация
The intersection of these strings is: {'ц', 'и', 'ф', 'е', 'а', 'к', 'р'}
```

Рисунок 7 – Пример выполнения программы

1.3 Индивидуальное задание, вариант 5 (рис. 8

```
ex1.py x pr1.py x pr2.py x ind1.py x
1 ▶ #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 ▶ if __name__ == "__main__":
5     u = set("abcdefghijklmnopqrstuvwxyz")
6     a = {"c", "e", "h", "n"}
7     b = {"e", "f", "k", "n", "x"}
8     c = {"b", "c", "h", "p", "r", "s"}
9     d = {"b", "e", "g"}
10    x = (a.difference(b)).intersection(c.union(d))
11    print(f"x = {x}")
12    bn = u.difference(b)
13    y = (a.intersection(bn)).union(c.difference(d))
14    print(f"y = {y}")
15
```

Рисунок 8 – Код программы



```
ind1 x
C:\Users\Evil\PycharmProjects\LB10
x = {'h', 'c'}
y = {'s', 'h', 'c', 'r', 'p'}

Process finished with exit code 0
```

Рисунок 9 – Результат выполнения программы

2. Ответы на контрольные вопросы

1. Множеством в языке программирования Python называется неупорядоченная совокупность уникальных значений. В качестве элементов этого набора данных могут выступать любые неизменяемые объекты, такие как числа, символы, строки.

2. Пример 1: $a = \{1, 2, 0, 1, 3, 2\}$

Пример 2: $a = \text{set}(\text{'data'})$

3. С помощью оператора `in`.

4. С помощью цикла `for`.

5. Для создания множества можно в Python воспользоваться генератором, позволяющих заполнять списки, а также другие наборы данных с учетом неких условий.

6. С помощью метода `add()`, `a.add(<element>)`

7. 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.
- `clear` – удаление всех элементов множества

8. Пример объединения: `c = a.union(b)`

Пример пересечения: `c = a.intersection(b)`

Пример разности: `c = a.difference(b)`

9. Определение подмножества: `a.issubset(b)`

Определение надмножества: `a.issuperset(b)`

10. 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. Для преобразования множества в строку используется конкатенация текстовых значений, которую обеспечивает функция `join`.

Пример: `b = ','.join(a)`

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

Пример:

```
a = {('a', 2), ('b', 4)}
```

```
b = dict(a)
```

Для получения списка: используется вызов `list`, получающий в качестве аргумента множество `a`.

Пример:

```
a = {1, 2, 0, 1, 3, 2}
```

```
b = list(a)
```