

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

**Отчёт о лабораторной работе №4 по дисциплине основы программной
инженерии**

Выполнила:

Нестеренко Тамара Антоновна,
2 курс, группа ПИЖ-б-о-20-1,

Проверил:

Доцент кафедры инфокоммуникаций,
Воронкин Р.А.

Ставрополь, 2021 г.

ВЫПОЛНЕНИЕ

1. Практическая часть

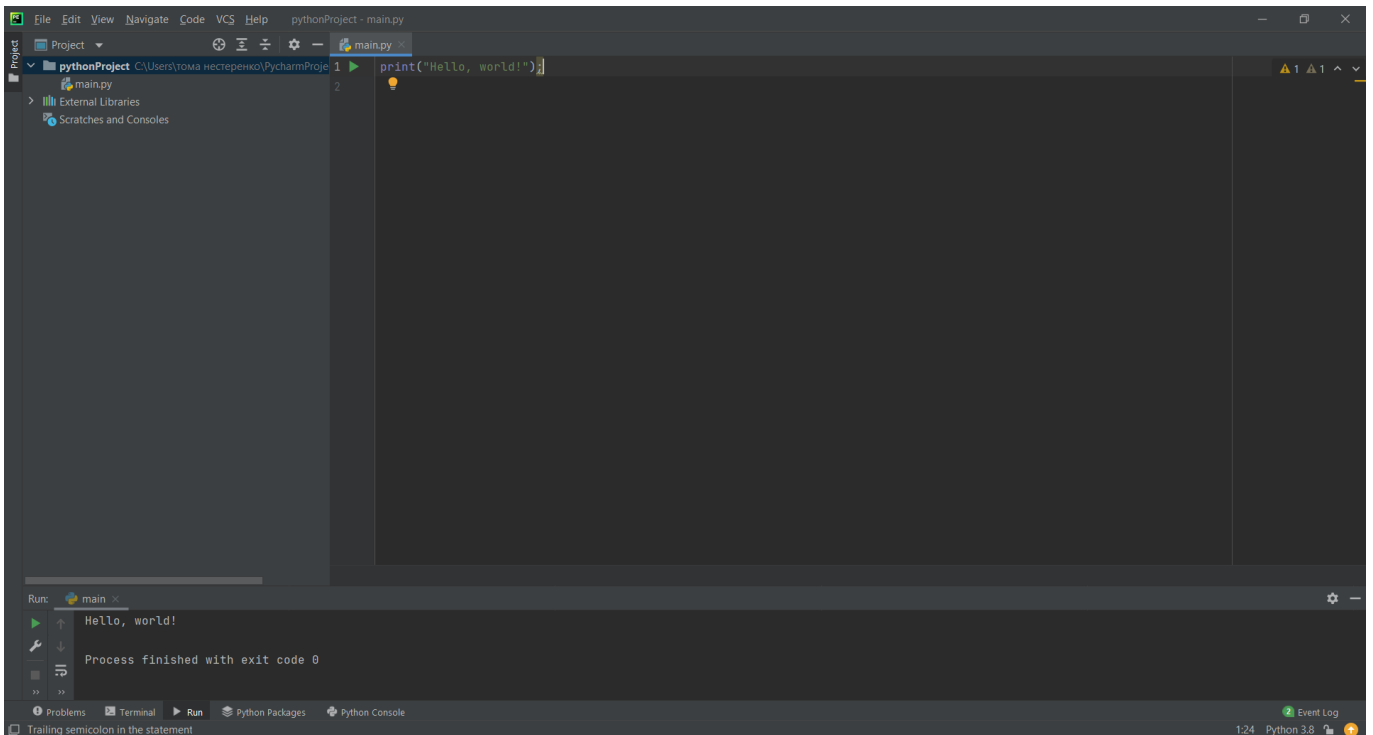


Рисунок 1 – Результат вывода строки приветствия

```
(base) C:\Users\тома нестеренко>python
Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
```

Рисунок 2 – Результат входа в интерактивный режим

```
>>> print("Hello, world!")
Hello, world!
```

Рисунок 3 – Результат вывода строки приветствия

```
>>> exit()
```

Рисунок 4 – Результат выхода из интерактивного режима

```
(base) C:\Users\тома нестеренко>python test.py
3
9
```

Рисунок 5 – Результат работы программы, которая принимает целое число на вход и выводит его квадрат

```
>>> import keyword
>>> print("Python keyword: ", keyword.kwlist)
Python keyword: ['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def',
'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal',
'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']
```

Рисунок 6 – Результат вывода ключевых слов

```
>>> keyword.iskeyword("try")
True
>>> keyword.iskeyword("b")
File "<stdin>", line 1
      keyword.iskeyword("b")
IndentationError: unexpected indent
```

Рисунок 7 – Результат проверки является ли идентификатор ключевым словом

```
>>> a = 4
>>> b = 5
>>> id(a)
2348804997456
>>> id(b)
2348804997488
>>> a = b
>>> id(a)
2348804997488
```

Рисунок 8 – Результат использования функции id()

```
>>> a = 10
>>> b = "hello"
>>> c = (1, 2)
>>> type(a)
<class 'int'>
>>> type(b)
<class 'str'>
>>> type(c)
<class 'tuple'>
```

Рисунок 9 – Результат использования функции type()

```
>>> k = 15
>>> id(k)
2348804997808
>>> type(k)
<class 'int'>
```

Рисунок 10 – Результат использования функций id() и type()

```
>>> a = [1, 2]
>>> id(a)
2348809797248
>>> a[1] = 3
>>> a
[1, 3]
>>> id(a)
2348809797248
```

Рисунок 11 – Результат изменение значения объекта

```
>>> 3+2
5
>>> a = 3
>>> b = 2
>>> a + b
5
>>> c = a + b
>>> print(c)
5
>>> a = a + b
>>> print(a)
5
>>> a = 3
>>> b = 2
>>> a += b
>>> print(a)
5
```

Рисунок 12 – Пример сложения переменных

```
>>> 4-2
2
>>> a = 5
>>> b = 7
>>> a - b
-2
```

Рисунок 13 – Пример вычитания переменных

```
>>> 5 * 8
40
>>> a = 4
>>> a *= 10
>>> print(a)
40
```

Рисунок 14 – Пример умножения переменных

```
>>> 9 / 3
3.0
>>> a = 7
>>> b = 4
>>> a / b
1.75
```

Рисунок 15 – Пример деления переменных

```
>>> 9 // 3
3
>>> a = 7
>>> b = 4
>>> a // b
1
```

Рисунок 16 – Пример получения целой части от деления переменных

```
>>> 9 % 5
4
>>> a = 7
>>> b = 4
>>> a % b
3
```

Рисунок 17 – Пример получения остатка от деления переменных

```
>>> 5 ** 4
625
>>> a = 4
>>> b = 3
>>> a ** b
64
```

Рисунок 18 – Пример возведения в степень переменных

```
>>> z = 1 + 2j
>>> print(z)
(1+2j)
>>> x = complex(3,2)
>>> print(x)
(3+2j)
```

Рисунок 19 – Пример создания комплексного числа

```
>>> x + z
(4+4j)
>>> x - z
(2+0j)
>>> x * z
(-1+8j)
>>> x / z
(1.4-0.8j)
>>> x ** z
(-1.1122722036363393-0.012635185355335208j)
>>> x ** 3
(-9+46j)
```

Рисунок 20 – Пример арифметических действий с комплексными числами

```
>>> x = 3 + 2j
>>> x.real
3.0
>>> x.imag
2.0
```

Рисунок 21 – Пример извлечения действительной и мнгоой части комплексного числа

```
>>> x.conjugate()
(3-2j)
```

Рисунок 22 – Пример получения комплексносопряженного числа, используя метод conjugate()

```
>>> p = 9
>>> q = 3
>>> p & q
1
>>> p | q
11
>>> p ^ q
10
>>> ~p
-10
>>> p << 1
18
>>> p >> 1
4
```

Рисунок 23 – Пример работы с битовыми операциями

```
>>> m = 124504
>>> hex(m)
'0x1e658'
>>> oct(m)
'0o363130'
>>> bin(m)
'0b11110011001011000'
```

Рисунок 24 – Пример представления чисел в другой системе счисления

```
>>> import math
>>> math.ceil(3.2)
4
>>> math.fabs(-7)
7.0
>>> math.factorial(5)
120
>>> math.floor(3.2)
3
>>> math.exp(3)
20.085536923187668
>>> math.log2(2)
1.0
>>> math.log10(15)
1.1760912590556813
>>> math.log(9)
2.1972245773362196
>>> math.log(4,8)
0.6666666666666667
>>> math.pow(3,4)
81.0
>>> math.sqrt(25)
5.0
>>> math.cos(0)
1.0
>>> math.sin(math.pi / 6)
0.49999999999999994
>>> math.tan(math.pi / 4)
0.9999999999999999
>>> math.acos(-1)
3.141592653589793
>>> math.asin(1)
1.5707963267948966
>>> math.atan(0.5)
0.4636476090008061
>>> math.pi
3.141592653589793
>>> math.e
2.718281828459045
```

Рисунок 25 – Пример основных функций библиотеки math


```

>>> print(1032)
1032
>>> print(2.34)
2.34
>>> print("Hello")
Hello
>>> print("a:", 1)
a: 1
>>> one = 1
>>> two = 2
>>> three = 3
>>> print(one, two, three)
1 2 3
>>> print("hello" + " " + "world")
hello world
>>> print(10 - 2.5/2)
8.75
>>> print("Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun", sep="-")
Mon-Tue-Wed-Thu-Fri-Sat-Sun
>>> print(1, 2, 3, sep="//")
1//2//3
>>> print(10, end="")
10>>>
>>> print(10, end='\n')
10
>>> print(10, end='\n\n')
10

```

Рисунок 26 – Пример вывода данных. Функция print()

```

>>> pupil = "Ben"
>>> old = 16
>>> grade = 9.2
>>> print("It's %s, %d. Level: %f" % (pupil, old, grade))
It's Ben, 16. Level: 9.200000
>>> print("It's %s, %d. Level: %.1f" % (pupil, old, grade))
It's Ben, 16. Level: 9.2
>>> print("This is a {0}, It's {1}.".format("ball", "red"))
This is a ball, It's red.
>>> print("This is a {0}, It's {1}.".format("cat", "white"))
This is a cat, It's white.
>>> print("This is a {0}, It's {1}{2}.".format(1,"a", "number"))
This is a 1, It's anumber.

```

Рисунок 27 – Пример использования метода format

```
>>> input()
YES!
'YES!'
>>> answer = input()
No, it is not.
>>> answer
'No, it is not.'
>>> print(answer)
No, it is not.
>>>
```

Рисунок 28 – Пример вывода данных. Функция input()

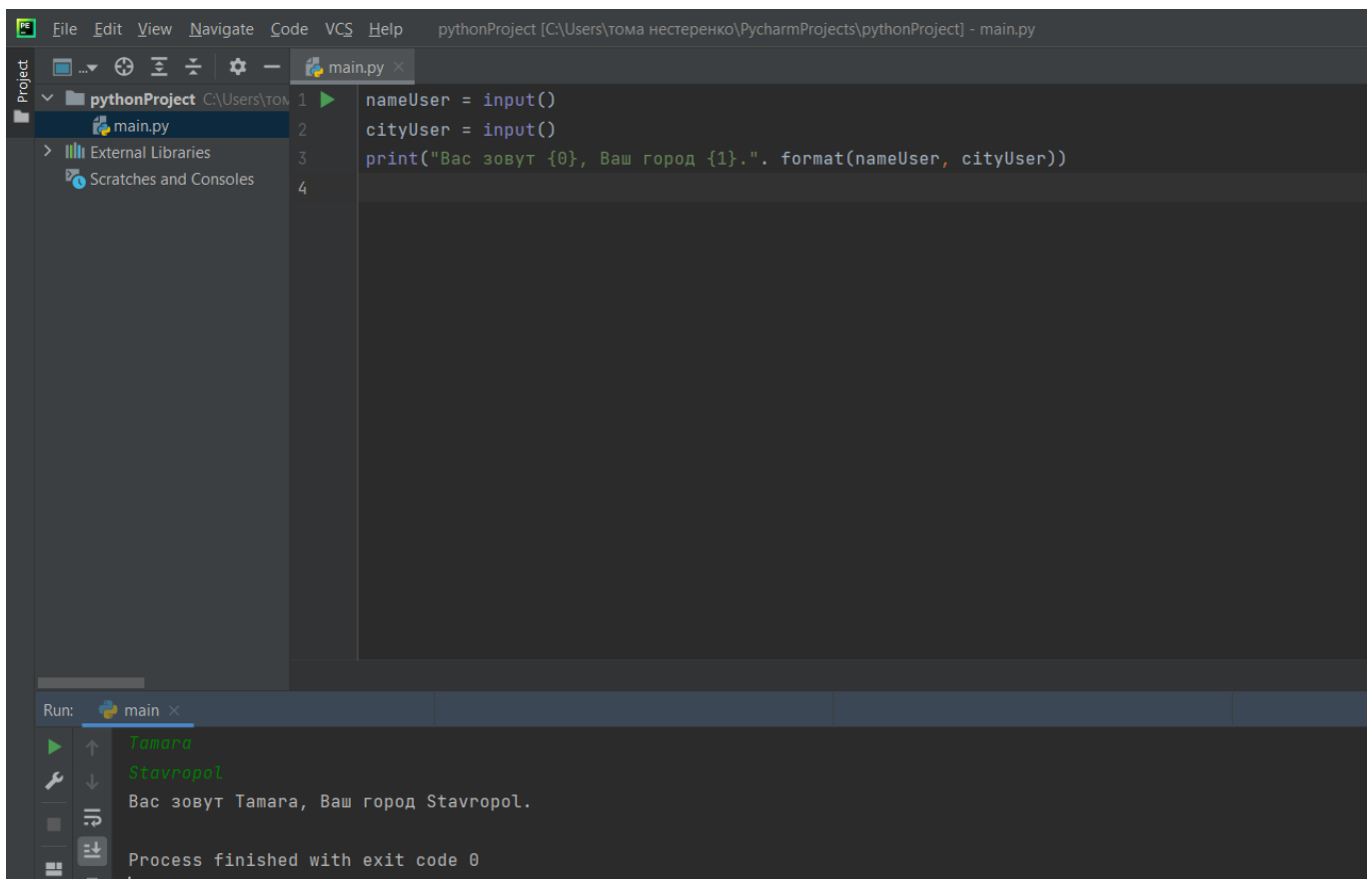


Рисунок 29 – Пример программы для ввода информации о имени и городе

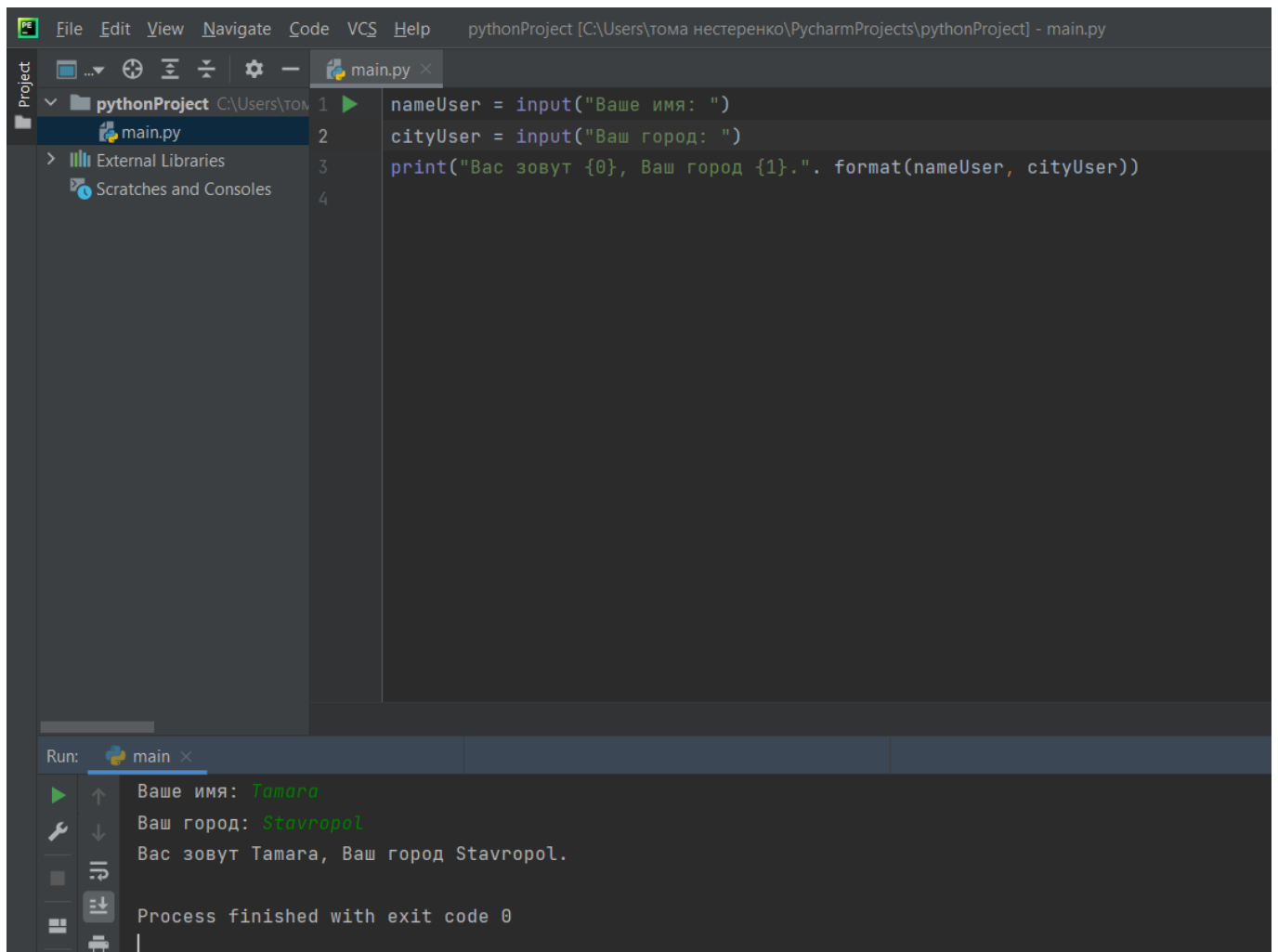


Рисунок 30 – Пример программы для ввода информации о имени и городе с приглашением к вводу

Ссылка на репозиторий: https://github.com/tamaranesterenko/Python.LR_4