

Homework_Lesson23_Report

****Цель занятия:****

Познакомиться с основами языка программирования Python, его типами данных и научиться создавать простые программы. В процессе выполнения заданий вы установите Python на свой компьютер, изучите историю Python, основные типы данных, а также научитесь работать со строками, числами и переменными.

****Задание:****

Выполните обязательные пункты (1–5), а затем выберите любое задание из пунктов 6–13.

Требование к отчёту: ваш код обязан запускаться и выполняться на любой машине и в любом окружении.

**Обязательные задания:**

1. **Изучите историю Python.**

- Ознакомьтесь с историей создания Python: кто создал язык, основные этапы его развития, и почему он стал популярным. Найти информацию можно в официальной [документации Python](<https://docs.python.org/>) или в интернете.

2. **Установите Python.**

- Скачайте и установите последнюю версию Python с [официального сайта](<https://www.python.org/downloads/>). Следуйте инструкциям, описанным в документации для вашей операционной системы.

3. **Изучите типы данных Python.**

- Ознакомьтесь с основными типами данных в Python: строки (`str`), числа (`int`, `float`), логические значения (`bool`), списки (`list`), словари (`dict`) и другие. Постарайтесь понять, как они используются.

4. **Напишите программу, которая выводит фразу «Привет, мир!».

- Это базовая программа, которая поможет вам познакомиться с функцией вывода (`print`).

5. **Напишите программу, которая вычисляет сумму $2 + 2$ и выводит результат.

- Программа должна использовать операцию сложения и выводить ответ на экран с помощью функции `print`.

6. **Программа приветствия:

- Напишите программу, которая запрашивает у пользователя его имя и выводит сообщение: «Привет, [имя]!».

7. ****Числовой вывод:****

- Создайте программу, которая выводит на экран числа от 1 до 10.

8. ****Определение возраста:****

- Напишите программу, которая запрашивает возраст пользователя и выводит сообщение: «Ваш возраст — [возраст] лет».

9. ****Произведение чисел:****

- Напишите программу, которая запрашивает у пользователя два числа и выводит их произведение.

10. ****Первая буква слова:****

- Создайте программу, которая запрашивает у пользователя слово и выводит его первую букву.

11. ****Квадрат числа:****

- Напишите программу, которая запрашивает у пользователя целое число и выводит его квадрат.

12. ****Таблица умножения:****

- Создайте программу, которая выводит таблицу умножения на число 5 (от 1 до 10).

13. ****Среднее арифметическое:****

- Напишите программу, которая запрашивает у пользователя два числа и выводит их среднее арифметическое.

Задание 1

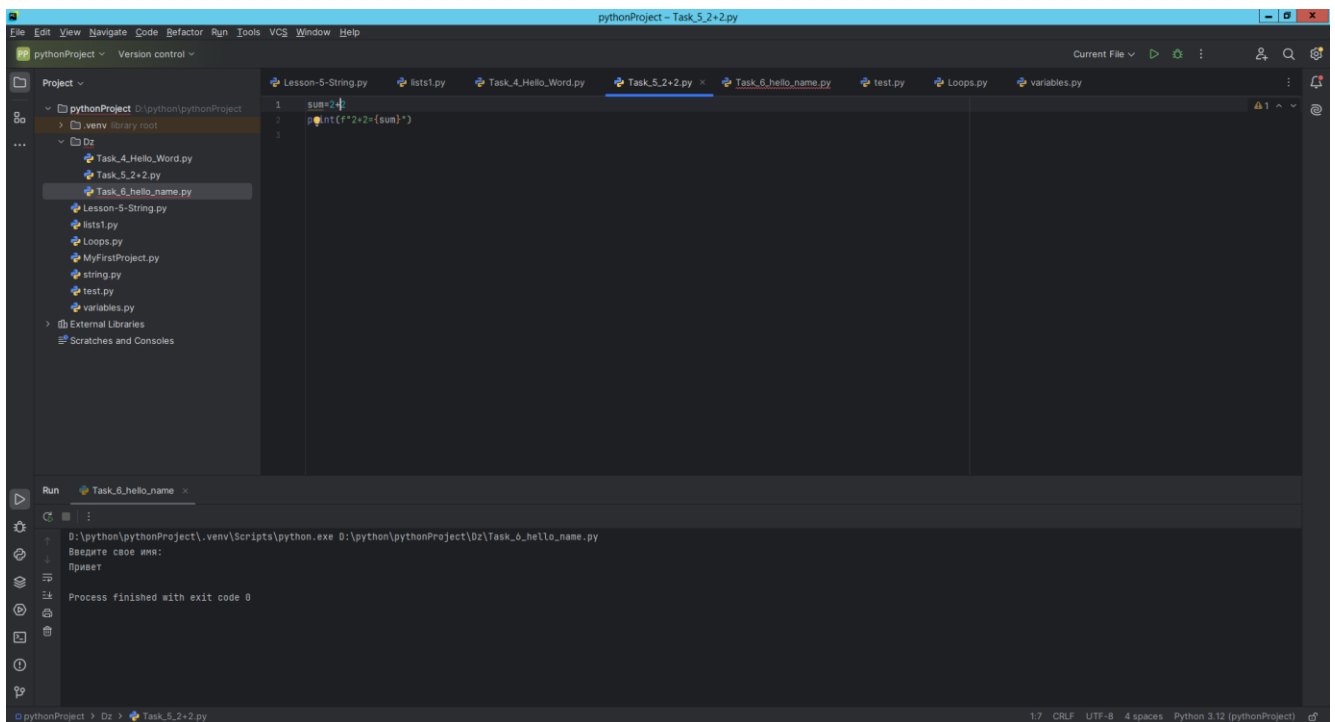
- Ознакомьтесь с историей создания Python: кто создал язык, основные этапы его развития, и почему он стал популярным. Найти информацию можно в официальной [документации Python](<https://docs.python.org/>) или в интернете. С историей ознакомился.

Задание 2

****Установите Python.****

- Скачайте и установите последнюю версию Python с [официального сайта](<https://www.python.org/downloads/>). Следуйте инструкциям, описанным в документации для вашей операционной системы.

Одной из машин, где я буду работать с Python, это Windows с установленным PyCharm Community Edition 2024.2.3



Задание 3

****Изучите типы данных Python.****

- Ознакомьтесь с основными типами данных в Python: строки (`str`), числа (`int`, `float`), логические значения (`bool`), списки (`list`), словари (`dict`) и другие. Постарайтесь понять, как они используются.

Строки (str)

Строки — это последовательности символов. Они могут содержать текст, пробелы, числа и другие символы.

```
name = "Hello World!"
```

Числа

Целые числа (int) — это числа без десятичной точки.

```
a = 144
```

Числа с плавающей запятой (float) — это числа с десятичной точкой.

```
b = 23.5
```

Логические значения (bool)

Тип bool используется для представления логических значений. Он может быть либо True (истина), либо False (ложь).

```
bool_t = True
```

```
bool_f = False
```

Списки (list)

Список — это упорядоченная коллекция элементов, которая может содержать элементы разных типов (строки, числа, другие списки и т.д.). Списки в Python могут быть изменяемыми.

```
fruits = ['apple', 'orange', 'banana', 'mango']  
numbers = [1, 2, 3, 4, 5]
```

Списки поддерживают индексацию, срезы и могут быть изменены:

```
fruits[1] = 'pineapple '
```

Словари (dict)

Словарь представляет собой набор пар "ключ-значение". Ключи в словаре уникальны, и через них можно получить соответствующее значение.

```
person = {'name': 'Алексей', 'age': 33, 'city': 'Минск'}  
rint(person['name']) # выведет 'Алексей'
```

Множества (set)

Множества — это коллекции уникальных элементов без какого-либо порядка. Они не поддерживают дублирование.

```
numbers = {1, 2, 3, 4}
```

Если добавить в множество повторяющийся элемент, он не будет добавлен:

```
numbers.add(3) # множество не изменится
```

Кортежи (tuple)

Кортежи — это неизменяемые (immutable) коллекции элементов, как и списки, но их элементы нельзя изменить после создания.

```
coordinates = (10, 20, 30)
```

Задание 4 – 13

Скрипты готовы и находятся в репозитори под следующими названиями:

Задание 4 - Task_4_Hello_Word.py

```
avl@ubuntu-s24:~$ python3 python/Task_4_Hello_Word.py
Hello Word
```

Задание 5 - Task_5_2+2.py

```
avl@ubuntu-s24:~$ python3 python/Task_5_2+2.py
2+2=4
```

Задание 6 - Task_6_hello_name.py

```
avl@ubuntu-s24:~$ python3 python/Task_6_hello_name.py
Введите свое имя:Alex
Привет Alex
```

Задание 7 - Task_7_number_1...10.py

```
avl@ubuntu-s24:~$ python3 python/Task_7_number_1...10.py
1 2 3 4 5 6 7 8 9 10
```

Задание 8 - Task_8_Yor_Age.py

```
avl@ubuntu-s24:~$ python3 python/Task_8_Yor_Age.py
Введите ваш возраст: 88
Ваш возраст: 88
```

Задание 9 - Task_9_Product of_number.py

```
avl@ubuntu-s24:~$ python3 python/Task_9_Product\ of_number.py
Ведите два числа через пробел для вычисления произведения:2 9
Произведение 2.0 и 9.0 равно 18.0
```

Задание 10 - Task_10_First_Letter.py

```
avl@ubuntu-s24:~$ python3 python/Task_10_First_Letter.py
Введите слово:Python
Первая буква слова: P
```

Задание 11 - Task_11_Square_Number.py

```
avl@ubuntu-s24:~$ python3 python/Task_11_Square_Number.py
Введите число: 4
Корень квадратный числа 4.0 равен 16.0
```

Задание 12 - Task_12_multiplication.py

```
avl@ubuntu-s24:~$ python3 python/Task_12_multiplication.py
Введите число: 5
5.0 * 1 = 5.0
5.0 * 2 = 10.0
5.0 * 3 = 15.0
5.0 * 4 = 20.0
5.0 * 5 = 25.0
5.0 * 6 = 30.0
5.0 * 7 = 35.0
5.0 * 8 = 40.0
5.0 * 9 = 45.0
5.0 * 10 = 50.0
```

Задание 13 - Task_13_arithmetic_average.py

```
avl@ubuntu-s24:~$ python3 python/Task_13_arithmetic_average.py
```

```
Введи два числа через пробол: 7 8
```

```
Среднее арифметическое 7.0 и 8.0 равно 7.5
```