

**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 №2.1 по дисциплине основы  
программной инженерии**

Выполнил:

Кожухов Филипп Денисович,  
2 курс, группа ПИЖ-б-о-20-1,

Проверил:

Доцент кафедры  
прикладной математики и  
компьютерной  
безопасности, Воронкин Р.А.

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2021 г.

## ВЫПОЛНЕНИЕ:

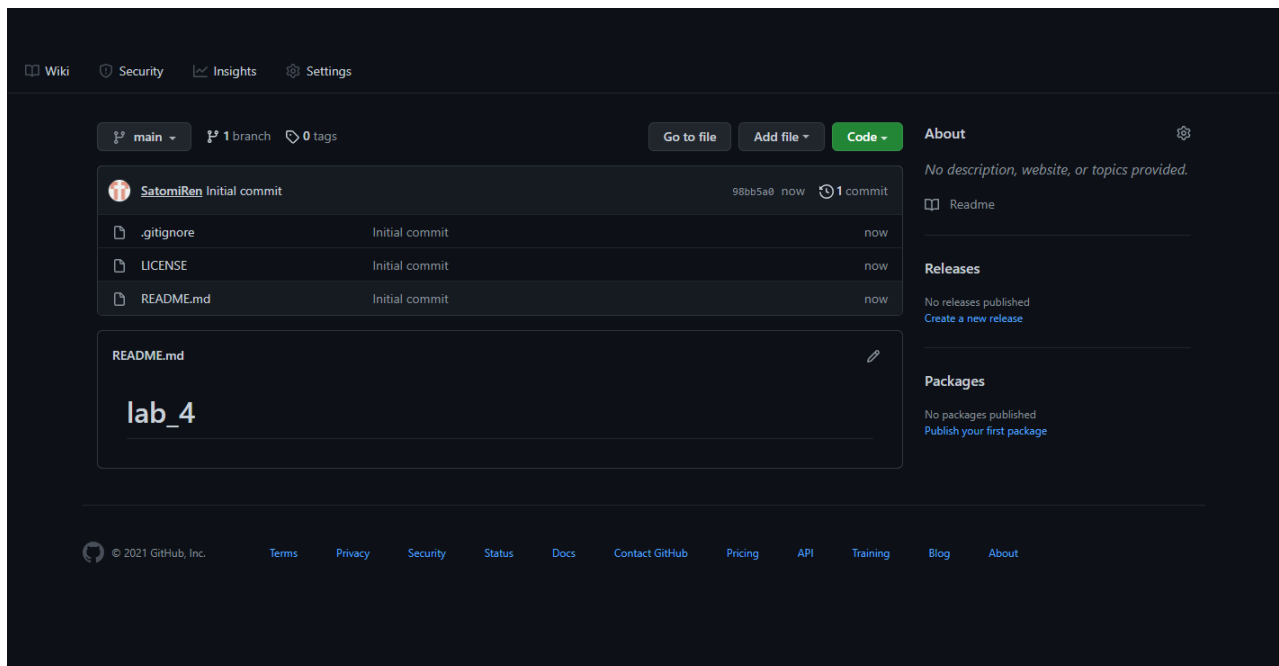


Рисунок 4.1 - Создание репозитория

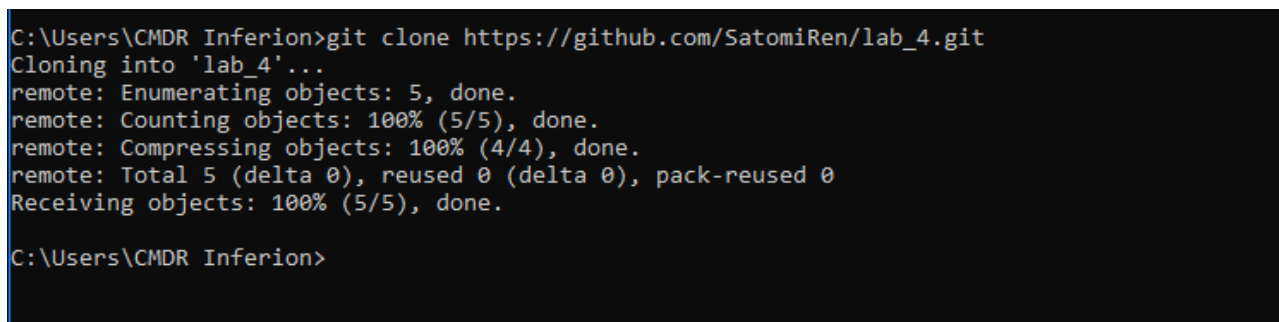


Рисунок 4.2 - Клонирование репозитория



Рисунок 4.3 - Организация репозитория в соответствии с моделью git-flow

Имя	Дата изменения	Тип	Размер
.idea	01.12.2021 10:51	Папка с файлами	
venv	01.12.2021 10:51	Папка с файлами	
.gitignore	30.11.2021 22:39	Текстовый докум...	2 КБ
LICENSE	30.11.2021 22:39	Файл	2 КБ
README.md	30.11.2021 22:39	Файл "MD"	1 КБ

Рисунок 4.5 - Создание проекта PyCharm в папке репозитория

The screenshot shows the PyCharm IDE interface. The top toolbar includes buttons for File, Edit, View, Navigate, Code, Refactor, Run, Tools, Git, Window, and Help. The main editor window displays the file `lab_4 - C:\Users\CMDR Inferion\lab_4\venv\Scripts\user.py`. The left sidebar shows the Project view with `user.py` and `arithmetic.py` open. The code in `user.py` is as follows:

```

1 name = input("What's your name? ")
2 old = input("How old are you? ")
3 address = input("Where do you live? ")
4 print("This is", name, ",")
5 print("Who is", old, "years old")
6 print("And lives in", address)

```

The bottom panel shows the Run output for the `user` configuration. The command executed is `"C:\Users\CMDR Inferion\lab_4\venv\Scripts\python.exe" "C:\Users\CMDR Inferion\lab_4\venv\Scripts\user.py"`. The output shows the program running with user input: `What's your name? Philip`, `How old are you? 19`, and `Where do you live? Stavropol, Russia`. The final output is `This is Philip ,`, `Who is 19 years old`, and `And lives in Stavropol, Russia`. The process finished with exit code 0.

Рисунок 4.6 - Программа user.py

The screenshot shows a code editor with a file named `arithmetic.py`. The code is as follows:

```
1 print("Solve the mathematical problem: 4 * 100 - 54")
2 answer = input("Your answer: ")
3 correct = 4 * 100 - 54
4 print("Correct answer is", correct)
```

Below the code editor, the 'Run' output is displayed:

```
Run: arithmetic
"C:\Users\CMDR Inferion\lab_4\venv\Scripts\python.exe" "C:/Users/CMDR Inferion/lab_4/venv/Scripts/arithmetic.py"
Solve the mathematical problem: 4 * 100 - 54
Your answer: 346
Correct answer is 346
Process finished with exit code 0
```

Рисунок 4.7 - Программа `arithmetic.py`

The screenshot shows a code editor with a file named `numbers.py`. The code is as follows:

```
1 first = int(input("Input 1st number: "))
2 second = int(input("Input 2nd number: "))
3 third = int(input("Input 3rd number: "))
4 fourth = int(input("Input 4th number: "))
5 sum1 = first + second
6 sum2 = third + fourth
7 total = float(sum1 / sum2)
8 print(round(total, 2))
```

Below the code editor, the 'Run' output is displayed:

```
Run: numbers
"C:\Users\CMDR Inferion\lab_4\venv\Scripts\python.exe" "C:/Users/CMDR Inferion/lab_4/venv/Scripts/numbers.py"
Input 1st number: 324
Input 2nd number: 7548
Input 3rd number: 7865
Input 4th number: 384
0.95
Process finished with exit code 0
```

Рисунок 4.8 - Программа `numbers.py`

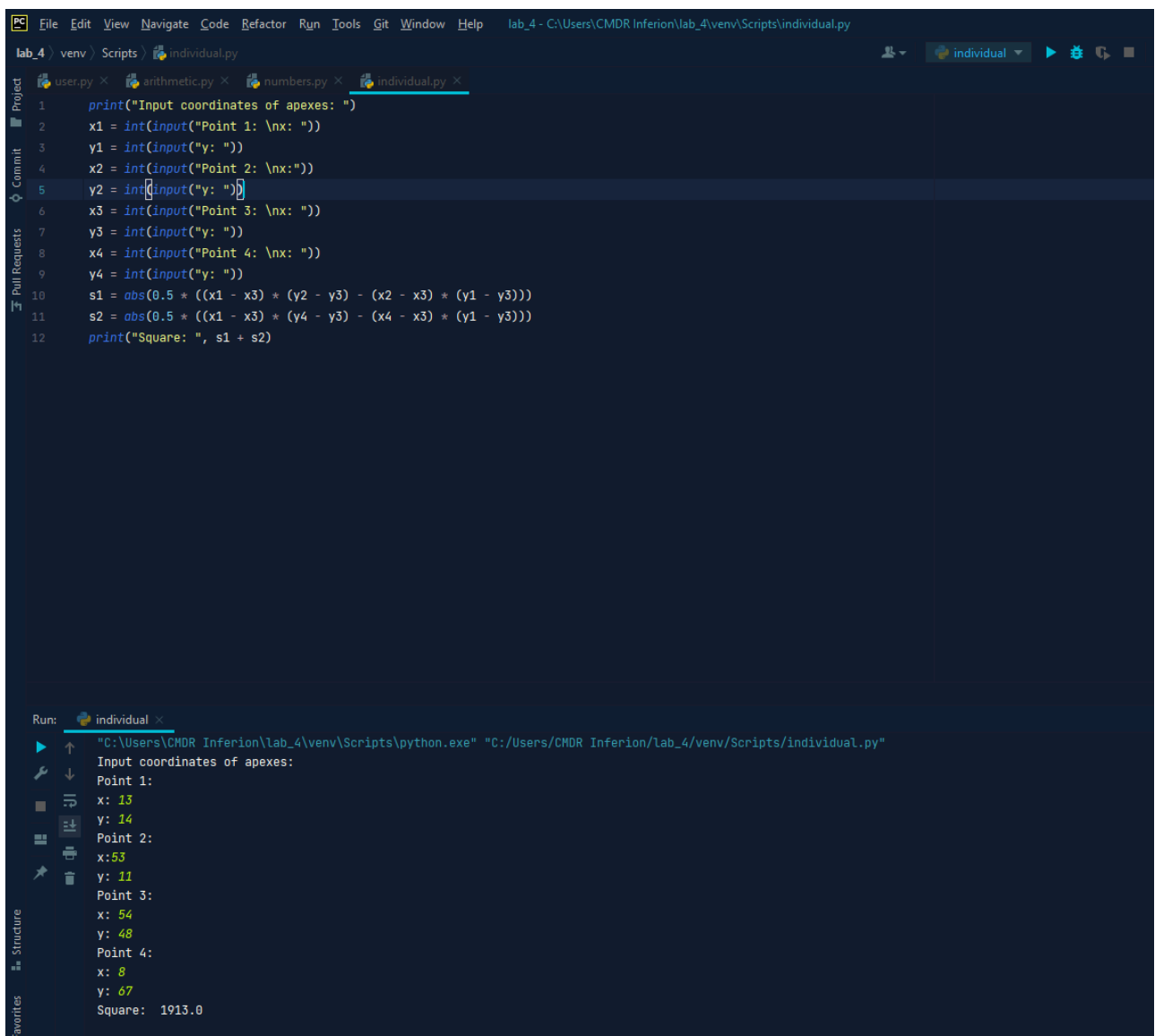


Рисунок 4.9 - Программа individual.py

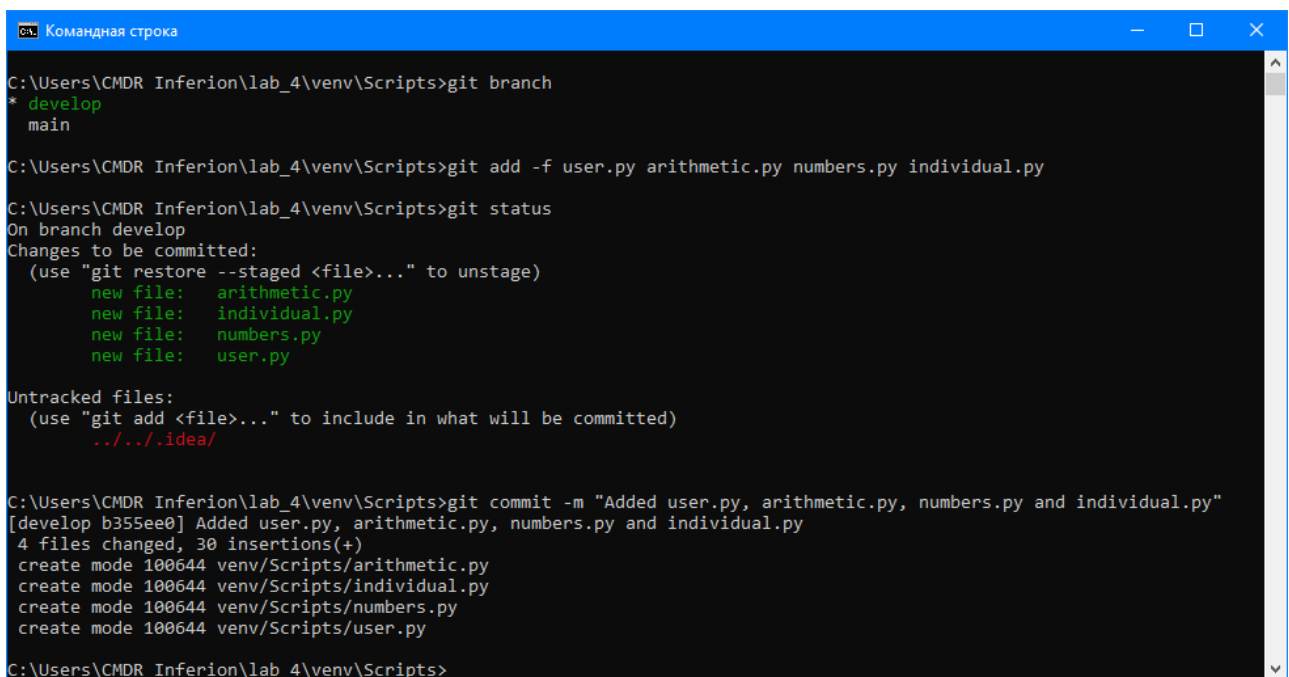


Рисунок 4.10 - Коммит вышеперечисленных файлов

```

C:\Users\CMDR Inferion\lab_4\venv\Scripts>git branch
* develop
  main

C:\Users\CMDR Inferion\lab_4\venv\Scripts>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

C:\Users\CMDR Inferion\lab_4\venv\Scripts>git merge develop
Updating 98bb5a0..b355ee0
Fast-forward
 venv/Scripts/arithmetic.py | 4 ++++
 venv/Scripts/individual.py | 12 ++++++++
 venv/Scripts/numbers.py    | 8 ++++++
 venv/Scripts/user.py       | 6 +++++
 4 files changed, 30 insertions(+)
 create mode 100644 venv/Scripts/arithmetic.py
 create mode 100644 venv/Scripts/individual.py
 create mode 100644 venv/Scripts/numbers.py
 create mode 100644 venv/Scripts/user.py

```

Рисунок 4.11 - Слияние веток develop и main

```

C:\Users\CMDR Inferion\lab_4\venv\Scripts>git push --set-upstream origin develop
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (8/8), 1.04 KiB | 1.04 MiB/s, done.
Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/SatomiRen/lab_4/pull/new/develop
remote:
To https://github.com/SatomiRen/lab_4.git
 * [new branch]      develop -> develop
Branch 'develop' set up to track remote branch 'develop' from 'origin'.

C:\Users\CMDR Inferion\lab_4\venv\Scripts>

```

Рисунок 4.12 - Отправка изменений в репозиторий GitHub

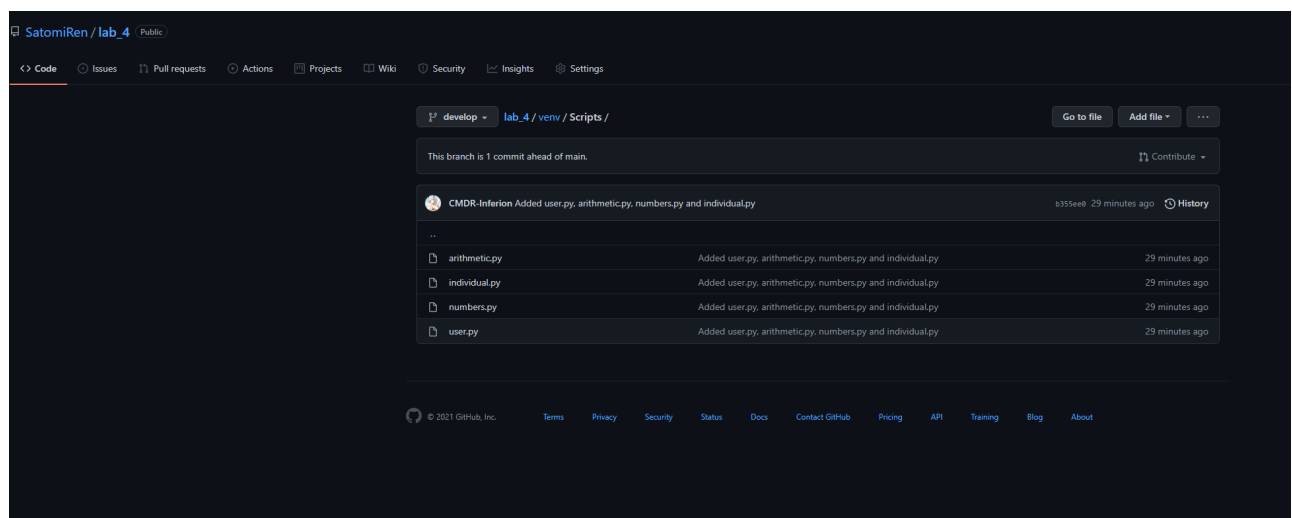


Рисунок 4.13 - Добавленные изменения в ветке develop на GitHub

## ОТВЕТЫ НА ВОПРОСЫ:

### 1. Установка в Windows:

- Запустите скачанный установочный файл.
- Выберите способ установки.
- Отметьте необходимые опции установки
- Выберите место установки

### Установка в Linux:

- `$ sudo apt-get install python3`

2. Anaconda кроме интерпретатора содержит в себе еще и набор дополнительных библиотек и среду разработки

3. Для выполнения проверки работоспособности Anaconda необходимо вначале запустить командный процессор с поддержкой виртуальных окружений Anaconda. В Windows это можно сделать выбрав следующий пункт главного меню системы Пуск -> Anaconda3(64-bit) -> Anaconda Prompt. В появившейся командной строке необходимо ввести «jupyter notebook» в результате чего отобразится процесс загрузки веб-среды Jupyter Notebook, после чего запустится веб-сервер и среда разработки в браузере.

4. При создании нового проекта в PyCharm укажите интерпретатор, который будет использоваться для запуска и отладки.

5. При запуске PyCharm выберите Open и укажите путь до файла программы, которую нужно запустить.

6. В интерактивном режиме интерпретатор Python считывает код построчно, а в пакетном режиме можно вписать весь код программы и интерпретатор считает всю программу полностью

7. Потому что тип переменной присваивается не при создании переменной, а при присваивании ей значения

8. Основные типы: int, float, complex, list, tuple, range, str, dict, bytes, bytearray, memoryview, set, frozenset

9. В Python существуют объекты - они являются абстракцией для представления данных, данные - это числа, списки, строки и т.п. При этом, под данными понимаются непосредственно сами объекты. Каждый объект имеет три атрибута - это идентификатор, значение и тип. Идентификатор помогает отличать объекты друг от друга. При инициализации переменной создается целочисленный объект, данный объект имеет некоторый идентификатор, значение 5, и тип: целое число.

10. Чтобы вывести список ключевых слов, нужно подключить модуль keyword и воспользоваться командой keyword.kwlist

11. Для того, чтобы посмотреть на объект с каким идентификатором ссылается данная переменная, можно использовать функцию id(). Функция type показывает тип данных, внесенных в переменную.

12. Неизменяемыми типами являются типы, которые не могут изменяться. Если тип данных изменяемый, то можно менять значение объекта.

13. При обычном делении вы получите число с его мнимой частью. При целочисленном делении же вы получите только целое число.

14. В Python входят функции создания комплексных чисел `complex(a,b)`, их сложения, вычитания, умножения, деления, возведения в степень, а также функция извлечения действительной и мнимой части.
15. В библиотеке `math` содержится большое количество часто используемых математических функций.
16. `sep` ставит отличный от пробела разделитель строк. Параметр `end` позволяет указать, что делать после вывода строки. По умолчанию происходит переход на новую строку.
17. Метод `format()` позволяет подставлять определенные данные в строку.
18. Для целочисленной переменной нужно ввести «A=17», а для вещественной «b=182.7»