

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №4
дисциплины «Основы программной инженерии»

Выполнил:
Яблоновский Дмитрий Николаевич
2 курс, группа ПИЖ-б-о-22-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка и
сопровождение программного
обеспечения», очная форма обучения

(подпись)

Руководитель практики:
Богданов С.С., ассистент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: Основы языка Python.

Цель работы: исследование процесса установки и базовых возможностей языка Python версии 3.x.

Порядок выполнения работы

1. Создал репозиторий GitHub.

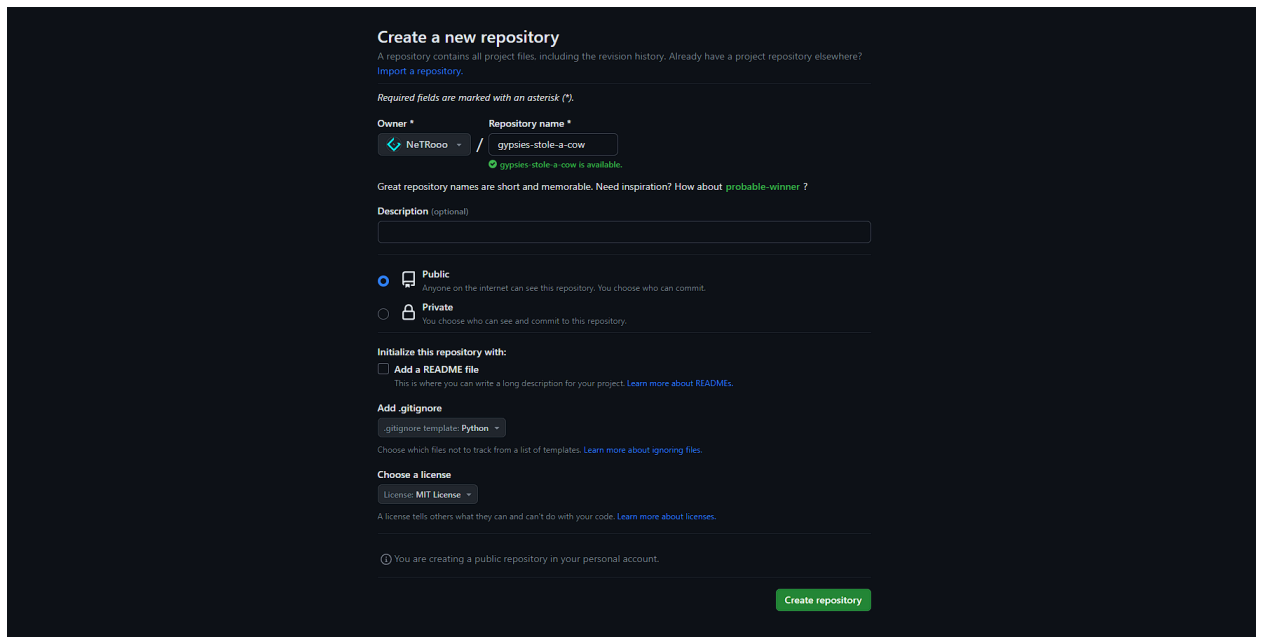
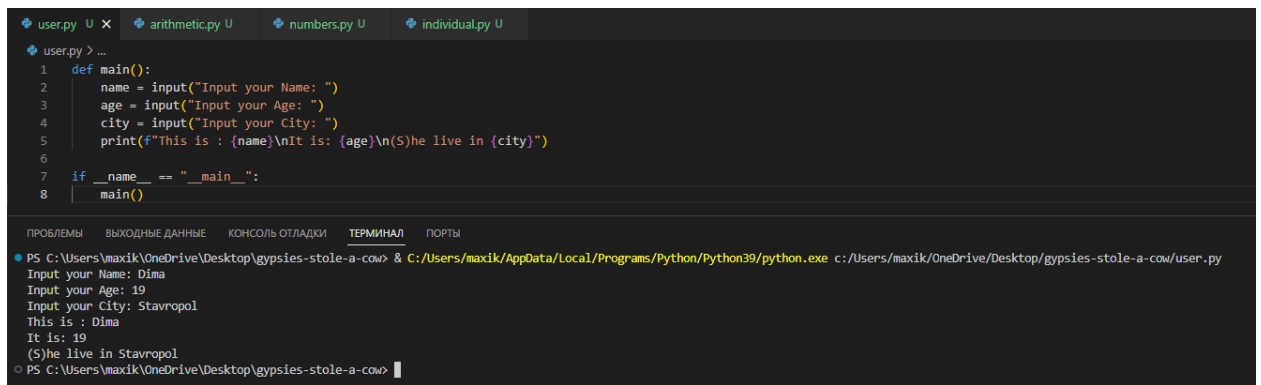


Рисунок 1- Создание репозитория

2. Проработал примеры из лабораторной работы.



Рисунок 2 – Выполнил клонирование созданного репозитория



```
user.py x arithmetic.py U numbers.py U individual.py U
user.py > ...
1 def main():
2     name = input("Input your Name: ")
3     age = input("Input your Age: ")
4     city = input("Input your City: ")
5     print(f"This is : {name}\nIt is: {age}\n(5)he live in {city}")
6
7 if __name__ == "__main__":
8     main()

ПРОБЛЕМЫ Выходные данные КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> & C:/Users/maxik/AppData/Local/Programs/Python/Python39/python.exe c:/Users/maxik/OneDrive/Desktop/gypsies-stole-a-cow/user.py
Input your Name: Dima
Input your Age: 19
Input your City: Stavropol
This is : Dima
It is: 19
(5)he live in Stavropol
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow>
```

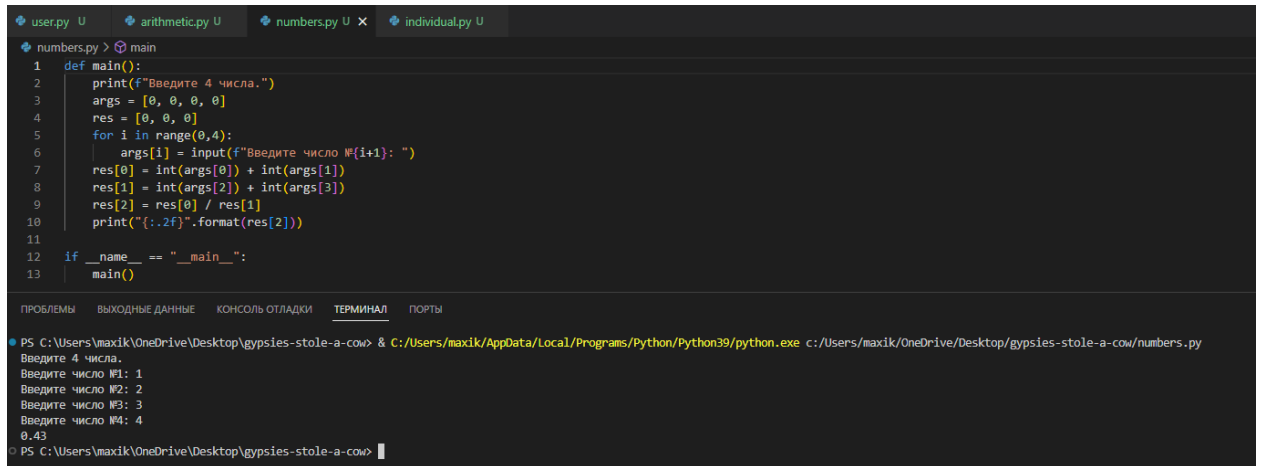
Рисунок 3 – Создал файл «user.py» и написал программу



```
user.py U arithmetic.py U x numbers.py U individual.py U
arithmetic.py > ...
1 def main():
2     solve = input("Solve the example: 4 * 100 - 54?\n")
3     if solve == '346':
4         print(f"The answer is correct")
5     else:
6         print(f"The answer is not correct\nCorrect: 346")
7
8 if __name__ == "__main__":
9     main()

ПРОБЛЕМЫ Выходные данные КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> & C:/Users/maxik/AppData/Local/Programs/Python/Python39/python.exe c:/Users/maxik/OneDrive/Desktop/gypsies-stole-a-cow/arithmetic.py
Solve the example: 4 * 100 - 54?
346
The answer is correct
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> & C:/Users/maxik/AppData/Local/Programs/Python/Python39/python.exe c:/Users/maxik/OneDrive/Desktop/gypsies-stole-a-cow/arithmetic.py
Solve the example: 4 * 100 - 54?
2134
The answer is not correct
Correct: 346
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow>
```

Рисунок 4 – Создал файл «arithmetic.py» и написал программу



```
user.py U arithmetic.py U numbers.py U x individual.py U
numbers.py > main
1 def main():
2     print(f"Введите 4 числа.")
3     args = [0, 0, 0, 0]
4     res = [0, 0, 0]
5     for i in range(0,4):
6         args[i] = input(f"Введите число №{i+1}: ")
7     res[0] = int(args[0]) + int(args[1])
8     res[1] = int(args[2]) + int(args[3])
9     res[2] = res[0] / res[1]
10    print("{:.2f}".format(res[2]))
11
12 if __name__ == "__main__":
13     main()

ПРОБЛЕМЫ Выходные данные КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> & C:/Users/maxik/AppData/Local/Programs/Python/Python39/python.exe c:/Users/maxik/OneDrive/Desktop/gypsies-stole-a-cow/numbers.py
Введите 4 числа.
Введите число №1: 1
Введите число №2: 2
Введите число №3: 3
Введите число №4: 4
0.43
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow>
```

Рисунок 5 – Создал файл «numbers.py» и написал программу

```
user.py U arithmetic.py U numbers.py U individual.py U x
individual.py > ...
1 def main():
2     print(f"Введите возраст Тани и Мити")
3     input1 = [0, 0]
4     res = [0, 0, 0]
5     for i in range(2):
6         input1[i] = int(input())
7     res[0] = ((int(input1[0]) + int(input1[1])) / 2)
8     res[1] = abs(int(input1[0]) - int(res[0]))
9     res[2] = abs(int(input1[1]) - int(res[0]))
10    print(f"Средний возраст: {res[0]}\nРазница Тани: {res[1]}\nРазница Мити: {res[2]}")
11
12 if __name__ == "__main__":
13     main()

ПРОБЛЕМЫ Выходные данные КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ ПОРТЫ
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> & C:/Users/maxik/AppData/Local/Programs/Python/Python39/python.exe c:/Users/maxik/OneDrive/Desktop/gypsies-stole-a-cow/individual.py
Введите возраст Тани и Мити
14
18
Средний возраст: 16.0
Разница Тани: 2
Разница Мити: 2
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> █
```

Рисунок 6 – Создал файл «individual.py» и написал программу

```
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> git add .
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> git commit -m "the best development of the century"
[dev 4b850df] the best development of the century
4 files changed, 43 insertions(+)
create mode 100644 arithmetic.py
create mode 100644 individual.py
create mode 100644 numbers.py
create mode 100644 user.py
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> █
```

Рисунок 7 – Выполнил коммит в ветке «dev»

```
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> git branch
dev
* main
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> git merge dev
Updating 9558c0f..4b850df
Fast-forward
 arithmetic.py | 9 ++++++++
 individual.py | 13 ++++++++
 numbers.py   | 13 ++++++++
 user.py      | 8 ++++++
4 files changed, 43 insertions(+)
create mode 100644 arithmetic.py
create mode 100644 individual.py
create mode 100644 numbers.py
create mode 100644 user.py
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> █
```

Рисунок 8 – Перешёл в ветку «main» и выполнил слияние с веткой «dev»

```
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 1.22 KiB | 1.22 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/NeTrooo/gypsies-stole-a-cow.git
9558c0f..4b850df main -> main
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> git checkout dev
Switched to branch 'dev'
Your branch is ahead of 'origin/dev' by 1 commit.
(use "git push" to publish your local commits)
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> git push
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/NeTrooo/gypsies-stole-a-cow.git
9558c0f..4b850df dev -> dev
PS C:\Users\maxik\OneDrive\Desktop\gypsies-stole-a-cow> █
```

Рисунок 9 – Отправил коммиты в удалённый репозиторий GitHub

Вывод: В данной работе мы научились использовать инструменты git для разработки в нескольких ветках, а после использовать слияние. Научились работать с языком программирования Python, а также его правилами PEP8

Ответы на контрольные вопросы

1. Как выполнить историю коммитов в Git? Какие существуют дополнительные опции для просмотра истории коммитов?

С помощью команды `git log`. Существуют множество дополнительных опций вот некоторые из них:

- `-p` или `-patch` – показывает разницу, внесенную в каждый коммит;
- `--stat` – позволяет увидеть сокращенную статистику;
- `--pretty` – эта опция меняет формат вывода;
- `--since` и `--until` – опции для ограничения вывода по времени

2. Как ограничить вывод при просмотре истории коммитов?

С помощью команды `git log` и его аргументов, например `--since`.

3. Как внести изменения в уже сделанный коммит?

С помощью команды `git commit --amend`.

4. Как отменить индексацию файла в Git?

С помощью команды `git reset HEAD`.

5. Как отменить изменения в файле?

С помощью команды `git checkout -- <file>`.

6. Что такое удаленный репозиторий Git?

Удалённые репозитории представляют собой версии вашего проекта, сохранённые в интернете или ещё где-то в сети.

7. Как выполнить просмотр удаленных репозиториях данного локального репозитория?

С помощью команды `git remote -v`.

8. Как добавить удаленный репозиторий для данного локального репозитория?

С помощью команды `git remote add <shortname> <url>`.

9. Как выполнить отправку/получение изменений с удаленного репозитория?

Получение изменение – `git fetch [remote-name]`, отправка изменений `git push <remote-name> <branch-name>`.

10. Как выполнить просмотр удаленного репозитория?
С помощью команды `git remote -v`.

11. Каково назначение тэгов Git?

Тэги Git - это ссылки на определенные коммиты в истории разработки. Они используются для пометки определенных версий или моментов в вашем проекте. Тэги обычно используются для обозначения релизов или важных этапов в разработке.

12. Как осуществляется работа с тэгами Git?

Для просмотра тэгов – `git tag`, для создания аннотированного тэга – `git tag -a v1.4 -m “сообщение”`, для отправки тэга на удаленный сервер – `git push origin <tagname>`, для удаления тэгов – `git tag -d <tagname>`.

13. . Самостоятельно изучите назначение флага `--prune` в командах `git fetch` и `git push` . Каково назначение этого флага?

Флаг `--prune` в командах `git fetch` и `git push` используется для удаления удаленных веток или тэгов, которые больше не существуют на удаленном репозитории. Если выполнить команду `git fetch --prune`, Git удалит локальные ссылки на удаленные ветки и тэги, которые были удалены на удаленном репозитории. Если выполнить команду `git push --prune`, Git удалит удаленные ветки и тэги на удаленном репозитории, которых больше нет в вашем локальном репозитории.