

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ  
ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего  
образования  
«СЕВЕРОКАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра  
инфокоммуникаций  
Институт цифрового  
развития**

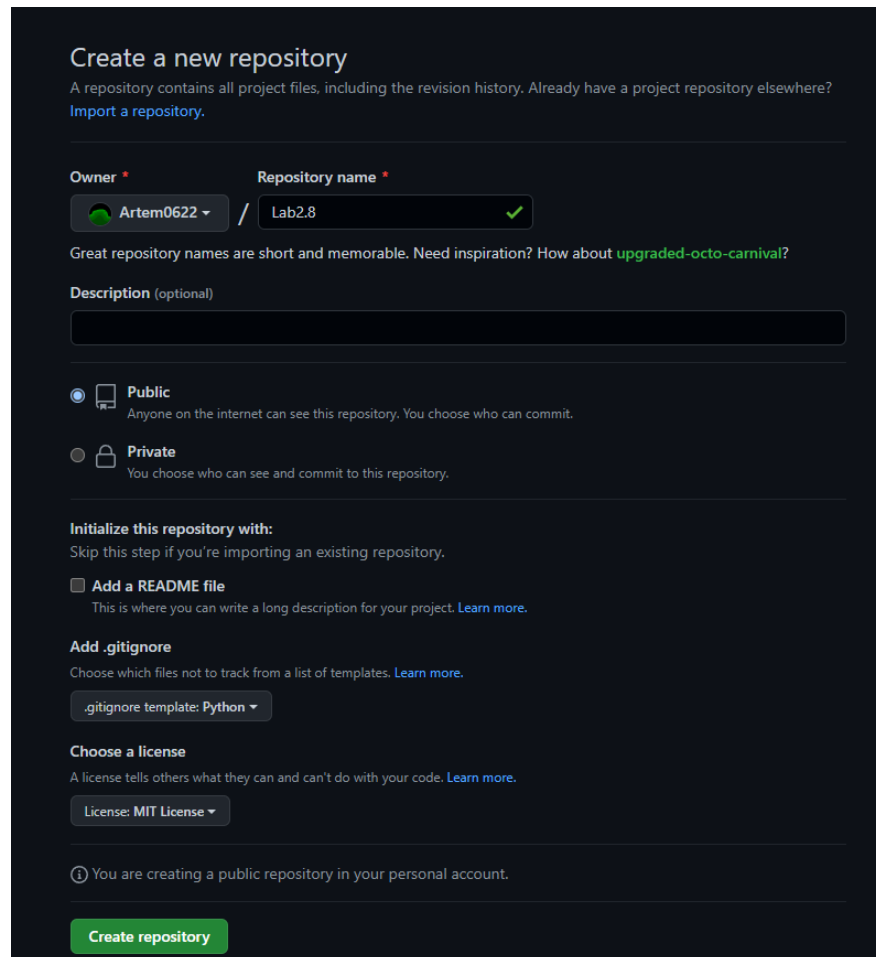
**ОТЧЁТ  
по лабораторной работе №2.8  
Дисциплина: «Программирование на  
Python» Тема: «Работа с функциями в  
языке Python»**

Выполнил: студент 2 курса  
группы ИВТ-б-о-21-1  
Богдашов Артём  
Владимирович

Ставрополь 2022

## Выполнение работы:

1. Создал репозиторий в GitHub «rep 2.6» в который добавил .gitignore, который дополнил правила для работы с IDE PyCharm с ЯП Python, выбрал лицензию MIT, клонировал его на лок. сервер и организовал в соответствии с моделью ветвления git-flow.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner \* Artem0622 / Repository name \* Lab2.8 ✓

Great repository names are short and memorable. Need inspiration? How about [upgraded-octo-carnival?](#)

Description (optional)

☒ Public  
Anyone on the internet can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

Initialize this repository with:  
Skip this step if you're importing an existing repository.

☐ Add a README file  
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore  
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: Python

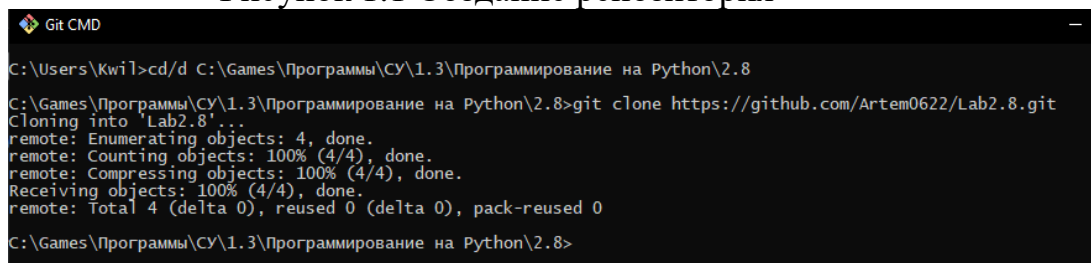
Choose a license  
A license tells others what they can and can't do with your code. [Learn more.](#)

License: MIT License

📘 You are creating a public repository in your personal account.

Create repository

Рисунок 1.1 Создание репозитория



```
Git CMD
C:\Users\Kwil>cd/d C:\Games\Программы\СУ\1.3\Программирование на Python\2.8
C:\Games\Программы\СУ\1.3\Программирование на Python\2.8>git clone https://github.com/Artem0622/Lab2.8.git
Cloning into 'Lab2.8'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
Receiving objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
C:\Games\Программы\СУ\1.3\Программирование на Python\2.8>
```

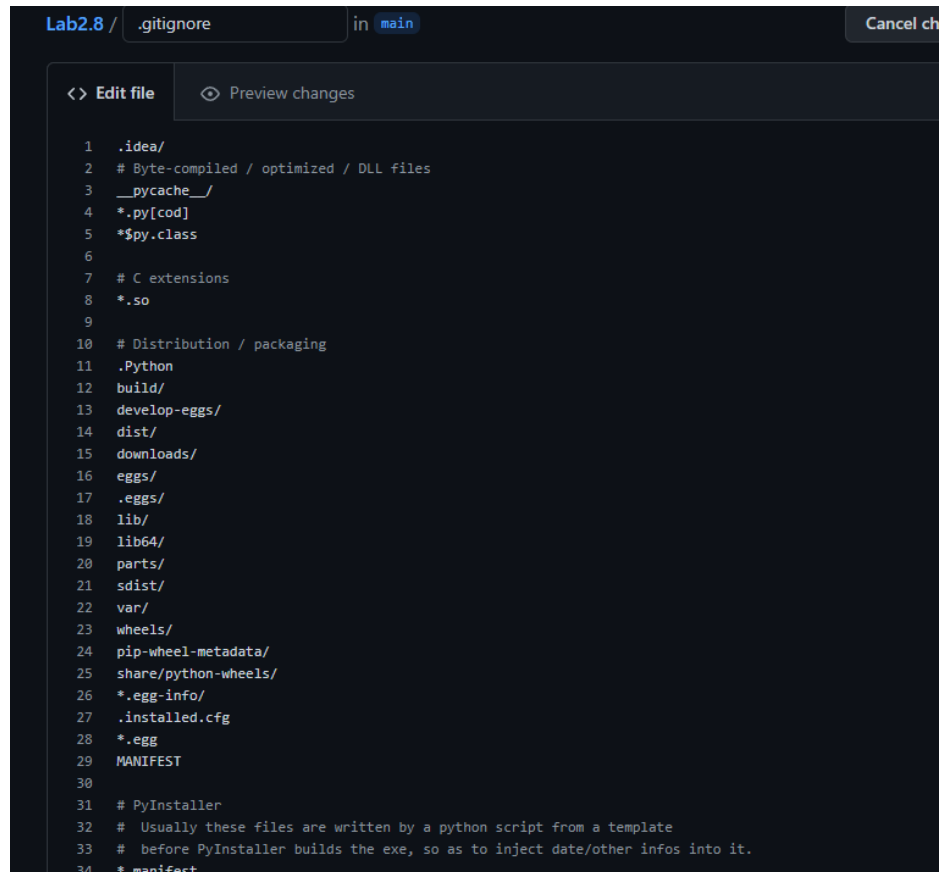
Рисунок 1.2 Клонирование репозитория

```

C:\Games\Программы\CV\1.3\Программирование на Python\2.8\Lab2.8>git flow init
Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]
How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Games/Программы/CV/1.3/Программирование на Python/2.8/Lab2.8/.git/hooks]

```

Рисунок 1.3 Организация репозитория в соответствии с моделью ветвления git-flow



```

1 .idea/
2 # Byte-compiled / optimized / DLL files
3 __pycache__/
4 *.py[cod]
5 *$py.class
6
7 # C extensions
8 *.so
9
10 # Distribution / packaging
11 .Python
12 build/
13 develop-eggs/
14 dist/
15 downloads/
16 eggs/
17 .eggs/
18 lib/
19 lib64/
20 parts/
21 sdist/
22 var/
23 wheels/
24 pip-wheel-metadata/
25 share/python-wheels/
26 *.egg-info/
27 .installed.cfg
28 *.egg
29 MANIFEST
30
31 # PyInstaller
32 # Usually these files are written by a python script from a template
33 # before PyInstaller builds the exe, so as to inject date/other infos into it.
34 *.manifest

```

Рисунок 1.4 Изменение .gitignore

2. Создал проект PyCharm в папке репозитория и проработал примеры.

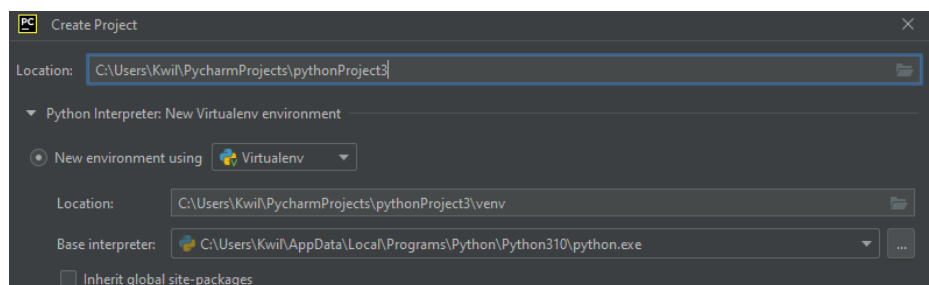


Рисунок 2.1 Создание проекта в PyCharm

```

Фамилия и инициалы: Иванов Иван
Должность: Доцент
Год поступления: 2009
>>> list
+-----+-----+-----+-----+-----+
| № |           Ф.И.О.           |      Должность      |   Год   |
+-----+-----+-----+-----+-----+
|  1 | Иванов Иван                | Доцент              |  2009   |
+-----+-----+-----+-----+-----+

```

Рисунок 2.2 Рез-т выполнения программы

3.(3 вариант). Выполнил индивидуальное задание.

```

>>> add
Фамилия и инициалы? Иванов Иван Иванович
Номер группы? ИТС
Успеваемость 5 5 5 5 5
>>> select
Таких студентов нет
>>> add
Фамилия и инициалы? Петров Пётр Гринченко
Номер группы? ППЖ
Успеваемость 5 2 3 2 5
>>> select
* Петров Пётр Гринченко группа № ППЖ

```

Рисунок 3.1 Вывод программы индивидуального задания

```

C:\Games\Программы\СУ\1.3\Программирование на Python\2.8\Lab2.8>git add .
C:\Games\Программы\СУ\1.3\Программирование на Python\2.8\Lab2.8>git commit -m "programm"
[develop 686229f] programm
2 files changed, 221 insertions(+)
create mode 100644 primer.py
create mode 100644 test.py

```

Рисунок 4.1 Коммит изменений

```

C:\Games\Программы\СУ\1.3\Программирование на Python\2.8\Lab2.8>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
C:\Games\Программы\СУ\1.3\Программирование на Python\2.8\Lab2.8>

```

Рисунок 4.2 Переход на ветку main

```
C:\Games\Программы\СУ\1.3\Программирование на Python\2.8\Lab2.8>git merge develop
Updating 881aeb6..686229f
Fast-forward
 primer.py | 123 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 test.py   | 98  +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 2 files changed, 221 insertions(+)
 create mode 100644 primer.py
 create mode 100644 test.py
```

Рисунок 4.3 Слияние ветки main с develop

```
C:\Games\Программы\СУ\1.3\Программирование на Python\2 и 8\lab2.8>git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 2.88 KiB | 2.88 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Artem0622/lab2.8.git
 8c24765..61e1d71  main -> main
```

Рисунок 4.4 Пуш изменений

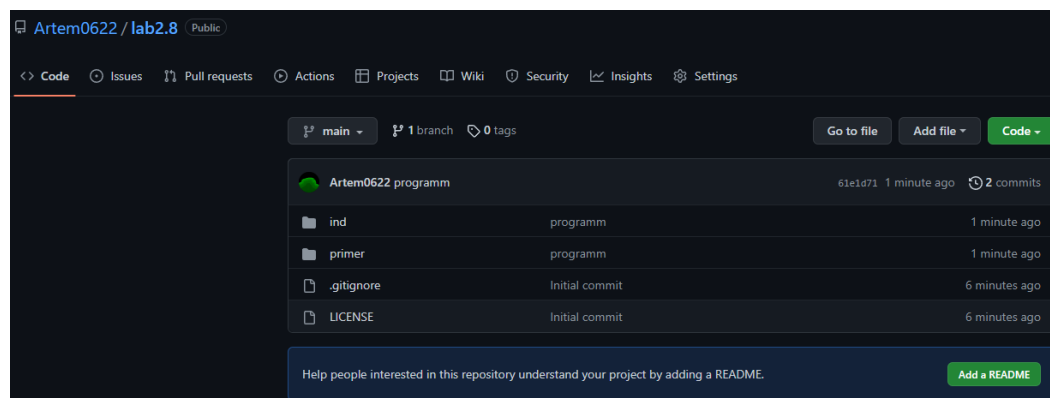


Рисунок 4.5 Изменение на уд сервере