

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 №2.13

Дисциплина: «Программирование на Python»

Тема: «Модули и пакеты в Python»

Вариант 25

Выполнила: студентка 2
курса группы ИВТ-б-о-21-1
Яковлева Елизавета
Андреевна

Цель работы: приобретение навыков по работе с модулями и пакетами при написании программ с помощью языка программирования Python версии 3.x.

Практическая часть:

1. Создала общедоступный репозиторий на GitHub.

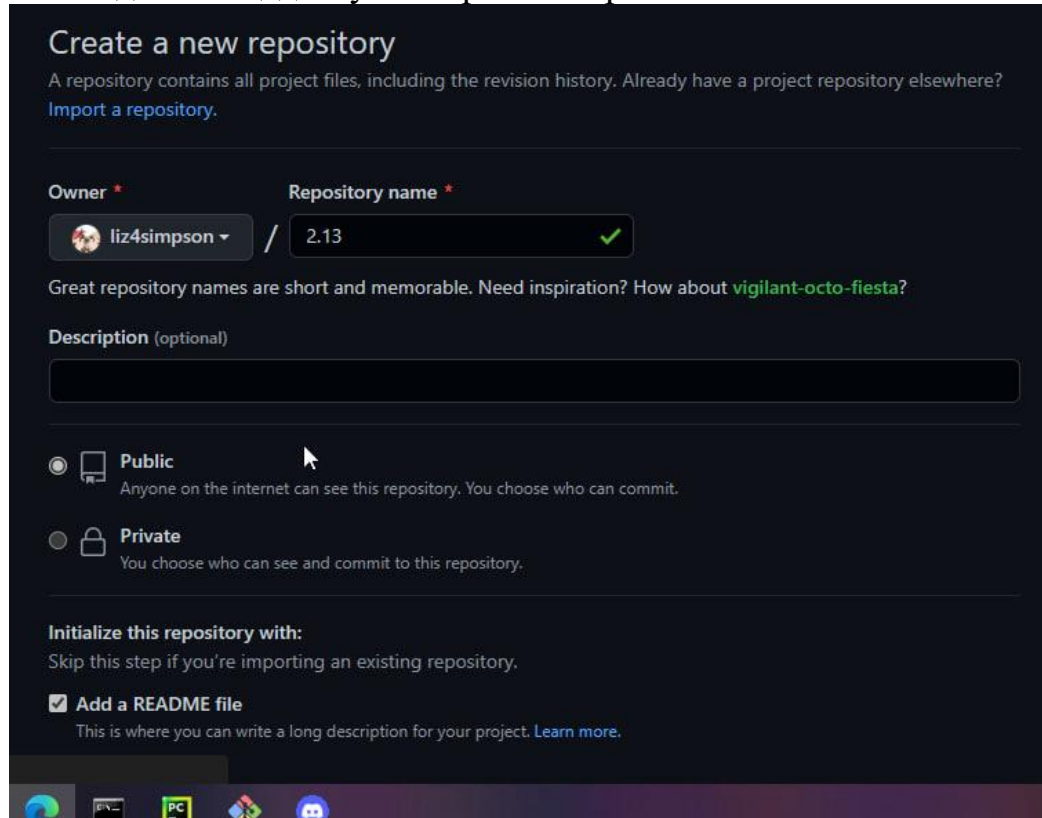


Рисунок 1. Создание репозитория

2. Выполнила клонирование созданного репозитория.

```
C:\Users\Elizaveta\Desktop\git>git clone https://github.com/liz4simpson/2.10.git
Cloning into '2.10'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 11 (delta 2), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (11/11), 5.17 KiB | 441.00 KiB/s, done.
Resolving deltas: 100% (2/2), done.
```

Рисунок 2. Клонирование репозитория

3. Дополнила файл .gitignore.

```
*.gitignore - Блокнот
Файл  Правка  Формат  Вид  Справка
# Created by https://www.toptal.com/developers/gitignore/api/python,pycharm
# Edit at https://www.toptal.com/developers/gitignore?templates=python,pycharm

### PyCharm ###
# Covers JetBrains IDEs: IntelliJ, RubyMine, PhpStorm, AppCode, PyCharm, CLion, Android Studio, WebStorm and Rider
# Reference: https://intellij-support.jetbrains.com/hc/en-us/articles/206544839
.idea/
.idea

# User-specific stuff
.idea/**/workspace.xml
.idea/**/tasks.xml
.idea/**/usage.statistics.xml
.idea/**/dictionaries
.idea/**/shelf

# AWS User-specific
.idea/**/aws.xml

# Generated files
.idea/**/contentModel.xml
```

Рисунок 3. Изменение файла .gitignore

4. Организовала репозиторий в соответствии git-flow.

```
C:\Users\Elizaveta\Desktop\git\2.10>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:/Users/Elizaveta/Desktop/git/2.10/.git/hooks]
```

Рисунок 4. Организация репозитория в соответствии с git-flow

5. Проработала примеры лабораторной работы.



Рисунок 5. Выполнение примеров

6. Выполнить индивидуальные задания.

Задание 1. Выполнить индивидуальное задание лабораторной работы 2.11, оформив все функции программы в виде отдельного модуля. Разработанный модуль должен быть подключен в основную программу с помощью одного из вариантов команды import.

```
C:\nana\2.13\Scripts\python.exe C:\Users\Elizaveta\Desktop\git\2.13\ind1\1.py
Уважаемая, Яковлева Елизавета! Вы делаете работу по замыканиям функций.

Process finished with exit code 0
```

Рисунок 6. Результат выполнения задания 1

Задание 2. Выполнить индивидуальное задание лабораторной работы 2.8, оформив все классы программы в виде отдельного пакета. Разработанный пакет должен быть подключен в основную программу с помощью одного из вариантов команды `import`. Настроить соответствующим образом переменную `_all_` в файле `_init_.py` пакета.

```
C:\nana\2.13\Scripts\python.exe C:\Users\Elizaveta\Desktop\git\2.13\ind2\2.py
>>> add
Пункт назначения: msk
Номер поезда: 12
Время отправления: (hh:mm) 12:00
>>> list
+-----+-----+-----+-----+
| № | Пункт назначения | Номер поезда | Время отправления |
+-----+-----+-----+-----+
| 1 | msk | 12 | 12:00:00 |
+-----+-----+-----+-----+
>>> select msk
+-----+-----+-----+-----+
| № | Пункт назначения | Номер поезда | Время отправления |
+-----+-----+-----+-----+
| 1 | msk | 12 | 12:00:00 |
+-----+-----+-----+-----+
>>> select 12
Список поездов пуст
>>>
```

Рисунок 7. Результат выполнения задания 2

1. Что является модулем языка Python?

Под модулем в Python понимается файл с расширением `.py`.

2. Какие существуют способы подключения модулей в языке Python?

`import имя_модуля`

`import имя_модуля as новое_имя from имя_модуля import имя_объекта`

`from имя_модуля import имя_объекта as псевдоним_объекта from
имя_модуля import *`

3. Что является пакетом языка Python?

Пакет в Python – это каталог, включающий в себя другие каталоги и модули, но при этом дополнительно содержащий файл `__init__.py`.

4. Каково назначение файла `init .py` ?

Если файл с именем `init .py` присутствует в каталоге пакета, то он вызывается при импорте пакета или модуля в пакете. Это может быть использовано для выполнения кода инициализации пакета, например инициализации данных уровня пакета.

5. Каково назначение переменной `all` файла `init .py` ?

`all` хранит список модулей, который импортируется при загрузке через конструкцию `from имя_пакета import *`.

Вывод: были приобретены навыки по работе с модулями и пакетами при написании программ с помощью языка программирования Python версии 3.

