

**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: «Основы программной инженерии»

Тема: «Модули и пакеты»

Выполнила:

студентка 2 курса

группы Пиж-б-о-21-1

Джолдошова Мээрим

Бекболотовна

Ставрополь 2022

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

1. Был создан репозиторий в Github в который были добавлены правила gitignore для работы IDE PyCharm, была выбрана лицензия MIT, сам репозиторий был клонирован на локальный сервер и был организован в соответствии с моделью ветвления git-flow.

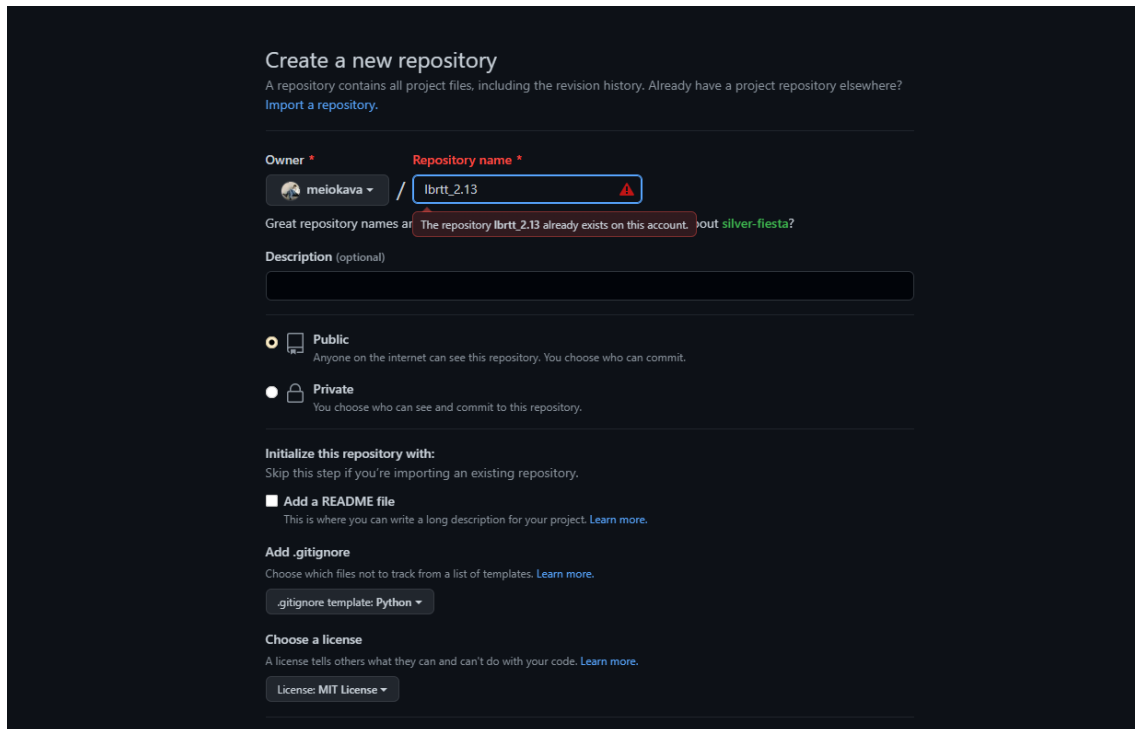


Рисунок 1 – Создание репозитория

```
C:\Users\mvideo>cd/d C:\lbrtt_2.13
C:\lbrtt_2.13>git clone https://github.com/meiokava/lbrtt_2.13.git
Cloning into 'lbrtt_2.13'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), done.
C:\lbrtt_2.13>
```

Рисунок 2 – Клонирование репозитория

Имя	Дата изменения	Тип	Ра:
.git	24.12.2022 18:13	Папка с файлами	
PyCharm	24.12.2022 18:15	Папка с файлами	
.gitignore	24.12.2022 10:52	Файл "GITIGNORE"	
LICENSE	24.12.2022 18:11	Файл	

Рисунок 5 – Папка PyCharm хранящая примеры

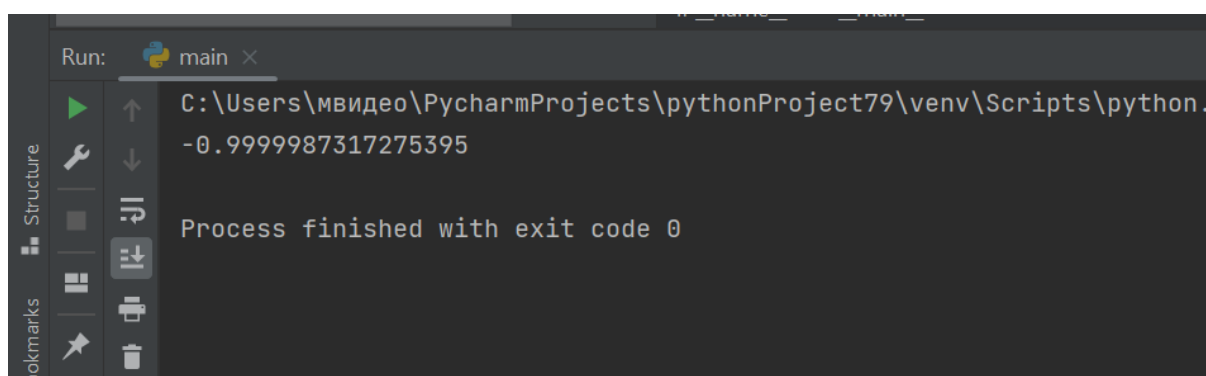


Рисунок 6 – Результат работы первого примера



Рисунок 7 – Результат работы второго примера

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`. Номер варианта уточнить у преподавателя.

Код модуля:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

def print_msg(ms):

    def printer(nm, fm):
        data = ms.format(n=nm, f=fm)
        return data
    return printer
```

Код основной программы:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

from printmsg_md import print_msg as pm

if __name__ == '__main__':
    d = "dear {f} {n} you have done a great job"
    another = pm(d)
    print(another("Gloria", "Labron"))
```

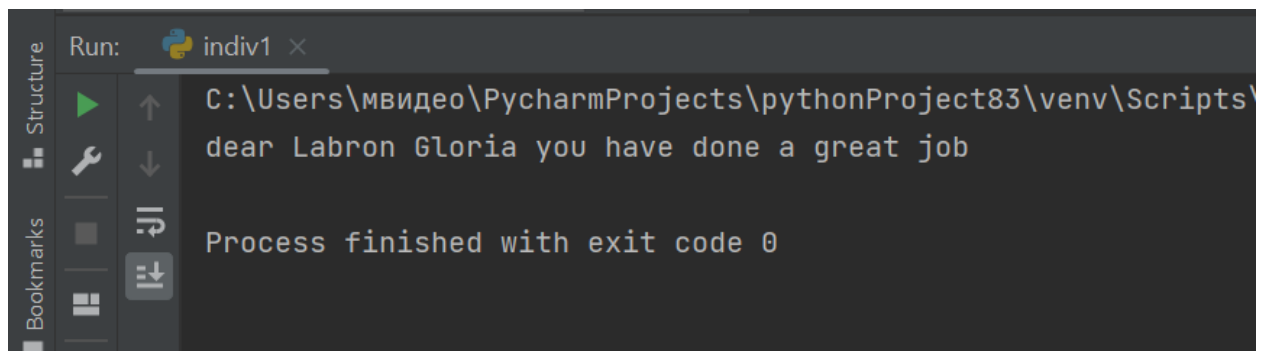


Рисунок 8 – Результат работы программы

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. Номер варианта уточнить у преподавателя.

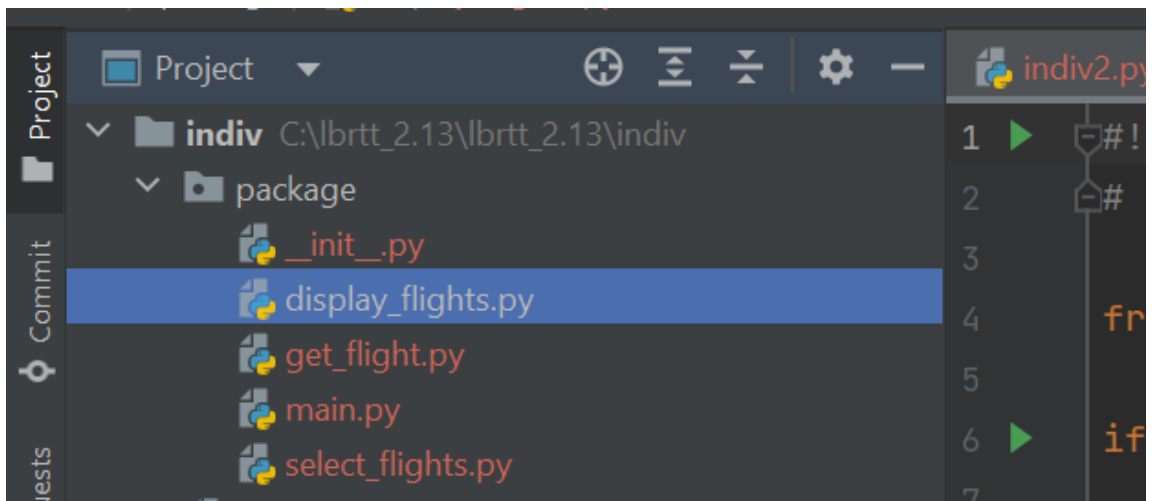


Рисунок 9 – Содержание пакета

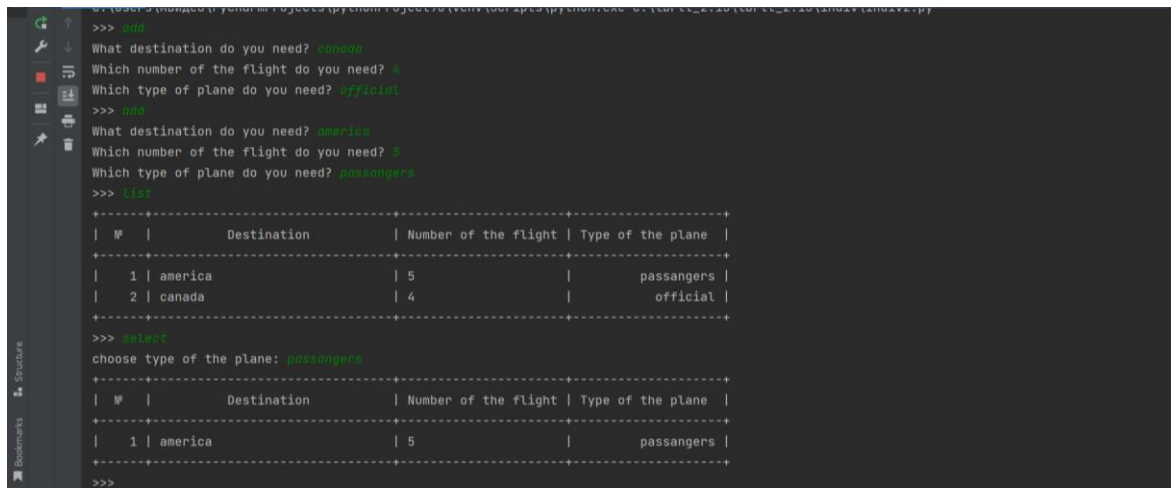


Рисунок 10 – Результат работы программы

Вывод: в результате выполнения лабораторной работы были приобретены навыки по работе декораторами функций при написании программ с помощью языка программирования Python.

Вопросы для защиты работы

1. Что является модулем языка Python?

Под модулем в Python понимается файл с расширением .py. Модули предназначены для того, чтобы в них хранить часто используемые функции, классы, константы и т. п. Можно условно разделить модули и программы: программы предназначены для непосредственного запуска, а модули для

импортирования их в другие программы. Стоит заметить, что модули могут быть написаны не только на языке Python, но и на других языках (например C).

2. Какие существуют способы подключения модулей в языке Python?

Самый простой способ импортировать модуль в Python это воспользоваться конструкцией:

```
import имя_модуля
```

За один раз можно импортировать сразу несколько модулей, для этого их нужно перечислить через запятую после слова `import`. Если вы хотите задать псевдоним для модуля в вашей программе, можно воспользоваться вот таким синтаксисом:

```
import имя_модуля as новое_имя
```

Для импортирования нескольких функций из модуля, можно перечислить их имена через запятую

```
from имя_модуля import имя_объекта1, имя_объекта2
```

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`. Пакеты используются для формирования пространства имен, что позволяет работать с модулями через указание уровня вложенности (через точку).

4. Каково назначение файла `__init__.py`?

В `__init__.py` файл заставляет Python рассматривать каталоги, содержащие его, как модули. Кроме того, это первый файл, загружаемый в модуль, поэтому вы можете использовать его для выполнения кода, который хотите запускать каждый раз при загрузке модуля, или для указания экспортируемых подмодулей.

5. Каково назначение переменной `__all__` файла `__init__.py`

Файл `__init__.py` может быть пустым или может содержать переменную `__all__`, хранящую список модулей, который импортируется при загрузке через конструкцию

```
from имя_пакета import *
```