

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

**Кафедра инфокоммуникаций
«Модули и пакеты в Python»**

**Отчет по лабораторной работе № 2.13
по дисциплине «Программирование на Python»**

Выполнил студент группы ИВТ-б-о-21-1

Шайдеров Дмитрий Викторович.

«3 » ноября 2022г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

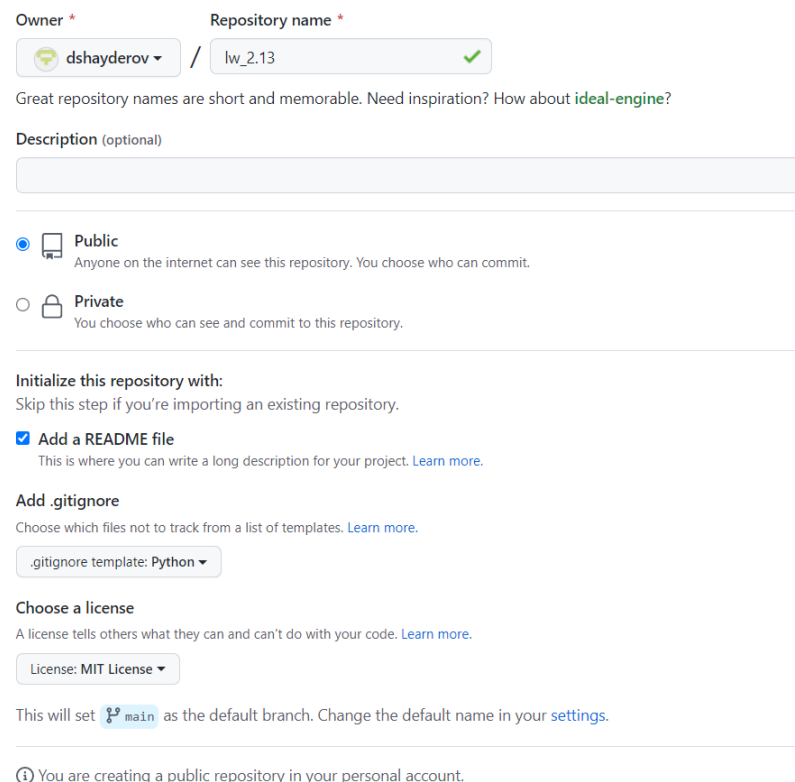
Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2022

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

Порядок выполнения работы:

1. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python.



Owner * / Repository name *

Great repository names are short and memorable. Need inspiration? How about [ideal-engine](#)?

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

This will set `main` as the default branch. Change the default name in your [settings](#).

① You are creating a public repository in your personal account.

Рисунок 1 - Создание репозитория

2. Выполните клонирование созданного репозитория.

```
Asus@LAPTOP-09A994CG MINGW64 /c/my projects/3
$ git clone https://github.com/dshayderov/lw_2.13.git
Cloning into 'lw_2.13'...
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), done.
Resolving deltas: 100% (2/2), done.
```

Рисунок 2 - Клонирование репозитория

3. Организуйте свой репозиторий в соответствии с моделью ветвления git-flow.

```

Asus@LAPTOP-09A994CG MINGW64 /c/my projects/3/lw_2.13 (main)
$ git checkout -b develop
Switched to a new branch 'develop'

Asus@LAPTOP-09A994CG MINGW64 /c/my projects/3/lw_2.13 (develop)
$

```

Рисунок 3 - Ветвление по модели git-flow

4. Создайте проект PyCharm в папке репозитория.

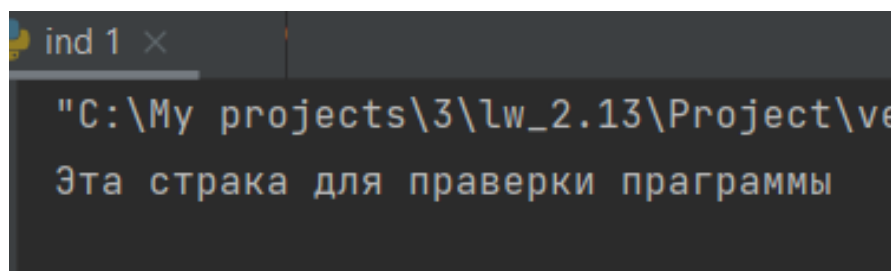
Этот компьютер > OS (C:) > My projects > 3 > lw_2.13				
Имя	Дата изменения	Тип	Размер	
Project	11.11.2022 11:35	Папка с файлами		
.gitignore	11.11.2022 11:34	Файл "GITIGNORE"	2 КБ	
LICENSE	11.11.2022 11:34	Файл	2 КБ	
README.md	11.11.2022 11:34	Файл "MD"	1 КБ	

Рисунок 4 - Проект PyCharm

5. 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`.

Используя замыкания функций, объявите внутреннюю функцию, которая бы все повторяющиеся символы заменяла одним другим указанным символом. Какие повторяющиеся символы искать и на что заменять, определяются параметрами внешней функции. Внутренней функции передается только строка для преобразования. Преобразованная (сформированная) строка должна возвращаться внутренней функцией.



ind 1 ×

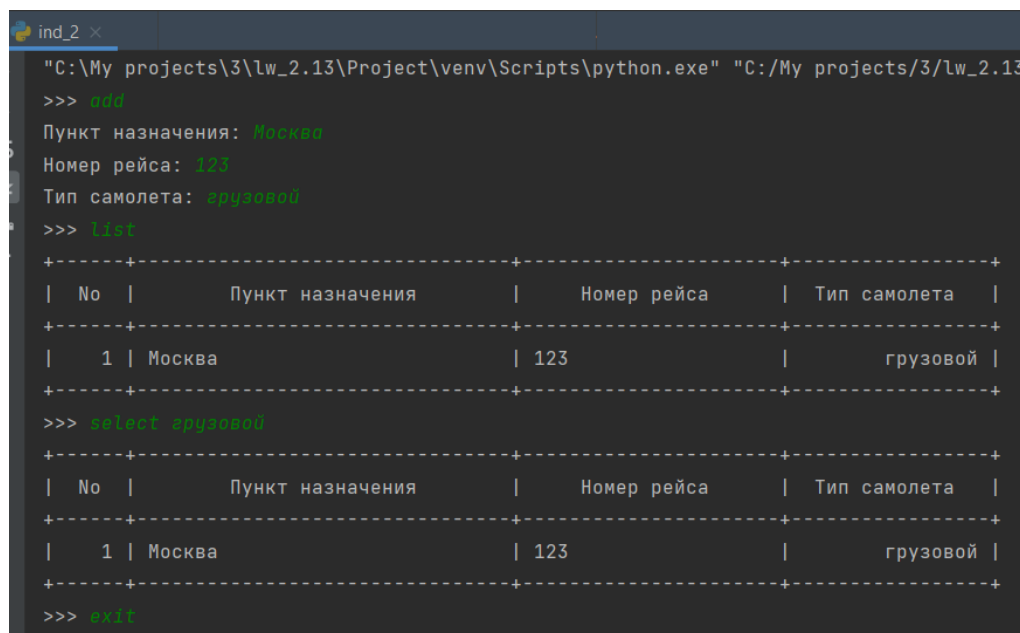
"C:\My projects\3\lw_2.13\Project\ve

Эта строка для проверки праграммы

Рисунок 5 - Результат выполнения индивидуального 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.

Использовать словарь, содержащий следующие ключи: название пункта назначения рейса; номер рейса; тип самолета. Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в список, состоящий из словарей заданной структуры; записи должны быть размещены в алфавитном порядке по названиям пунктов назначения; вывод на экран пунктов назначения и номеров рейсов, обслуживаемых самолетом, тип которого введен с клавиатуры; если таких рейсов нет, выдать на дисплей соответствующее сообщение.



```
ind_2 x
"C:\My projects\3\lw_2.13\Project\venv\Scripts\python.exe" "C:/My projects/3/lw_2.13
>>> add
Пункт назначения: Москва
Номер рейса: 123
Тип самолета: грузовой
>>> list
+-----+-----+-----+-----+
| No | Пункт назначения | Номер рейса | Тип самолета |
+-----+-----+-----+-----+
| 1 | Москва | 123 | грузовой |
+-----+-----+-----+-----+
>>> select грузовой
+-----+-----+-----+-----+
| No | Пункт назначения | Номер рейса | Тип самолета |
+-----+-----+-----+-----+
| 1 | Москва | 123 | грузовой |
+-----+-----+-----+-----+
>>> exit
```

Рисунок 6 - Результат выполнения индивидуального задания 2

Контрольные вопросы:

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.x.