

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 №2.13**  
**дисциплины «Основы кроссплатформенного программирования»**

Выполнил:  
Евдаков Евгений Владимирович  
1 курс, группа ИТС-б-о-22-1,  
11.03.02 «Инфокоммуникационные  
технологии и системы связи»,  
направленность (профиль)  
«Инфокоммуникационные системы и  
сети», очная форма обучения

---

(подпись)

Руководитель практики:  
Воронкин Р. А., доцент кафедры  
инфокоммуникаций

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2023 г.

**Тема:** Модули и пакеты

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

### **Ход работы:**

**Задание 1.** Создал общедоступный репозиторий на GitHub, в котором использована лицензий MIT и языкпрограммирования Python, также добавил файл .gitignore с необходимыми правилами. Клонировал свой репозиторий на свой компьютер.

```
C:\Users\Gaming-PC>git clone https://github.com/EvgenyEvdakov/Laba_2.13.git
Cloning into 'Laba_2.13'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (8/8), done.
Resolving deltas: 100% (1/1), done.
```

Рисунок 1. Клонирование репозитория

**Задание 2.** Организовал свой репозиторий в соответствии с моделью ветвления git-flow, появилась новая ветка develop в которой буду выполнять дальнейшие задачи.

```
C:\Users\Gaming-PC\Laba_2.13>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/Gaming-PC/Laba_2.13/.git/hooks]
```

Рисунок 2. Модель ветвления git-flow

**Задание 3.**

### **Индивидуальное задание 1.**

#### **Вариант 9 (по списку группы)**

Создал новый файл под названием individual1.py и файл модуль под названием utils.py.

**Условие задания:** оформить все функции программы в виде отдельного модуля. Разработанный модуль должен быть подключен в основную программу с помощью одного из вариантов команды `import`.

**Задание:** Используя замыкания функций, объявите внутреннюю функцию, которая принимает в качестве аргумента коллекцию (список или кортеж) и возвращает или минимальное значение, или максимальное, в зависимости от значения параметра `type` внешней функции. Если `type` равен «`max`», то возвращается максимальное значение, иначе – минимальное. По умолчанию `type` должно принимать значение «`max`». Вызовите внутреннюю функцию замыкания и отобразите на экране результат ее работы.

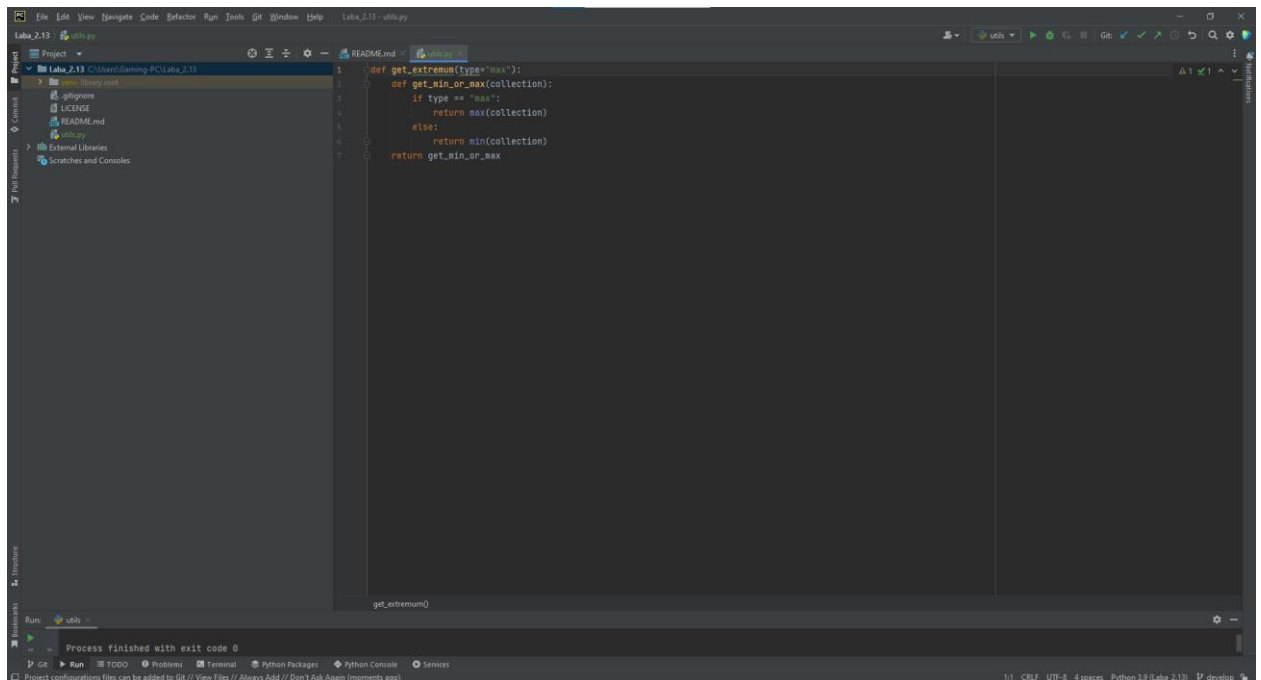


Рисунок 3. Модуль `utils.py`

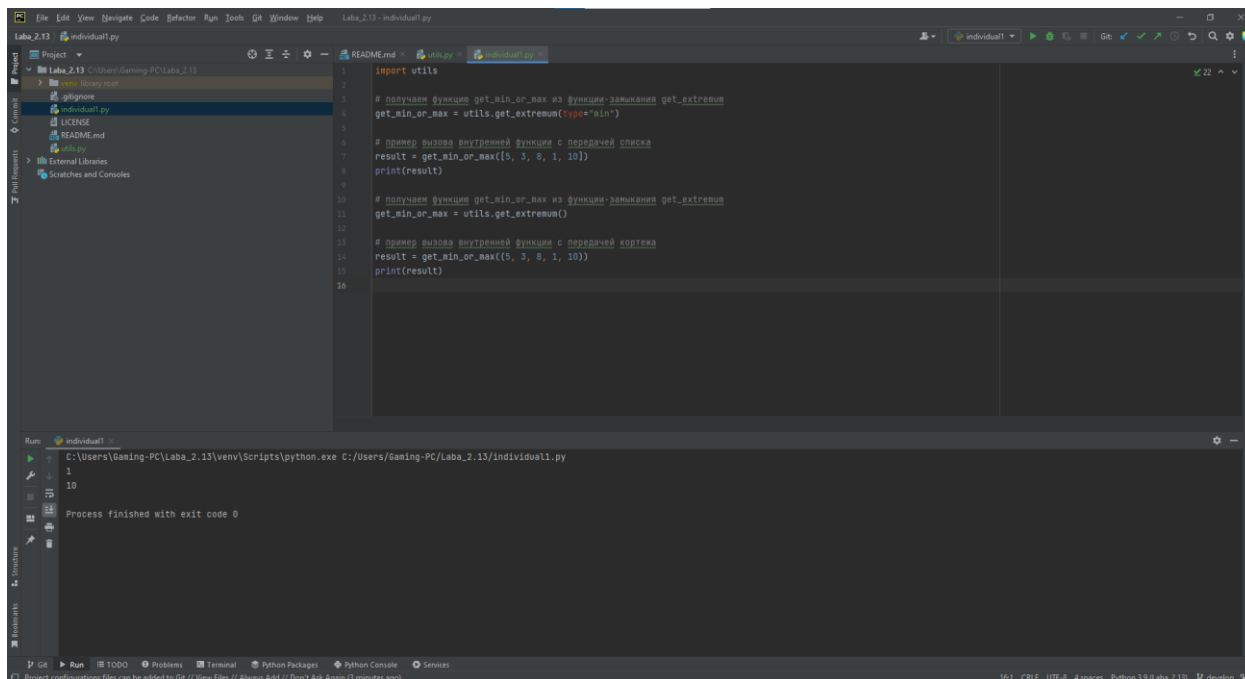


Рисунок 4. Основная программа с использованием модуля

## Задание 4.

### Индивидуальное задание 2.

#### Вариант 9 (по списку группы)

**Условие задания:** 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.

**Задание 2.8:** Решить индивидуальное задание лабораторной работы 2.6, оформив каждую команду в виде отдельной функции.

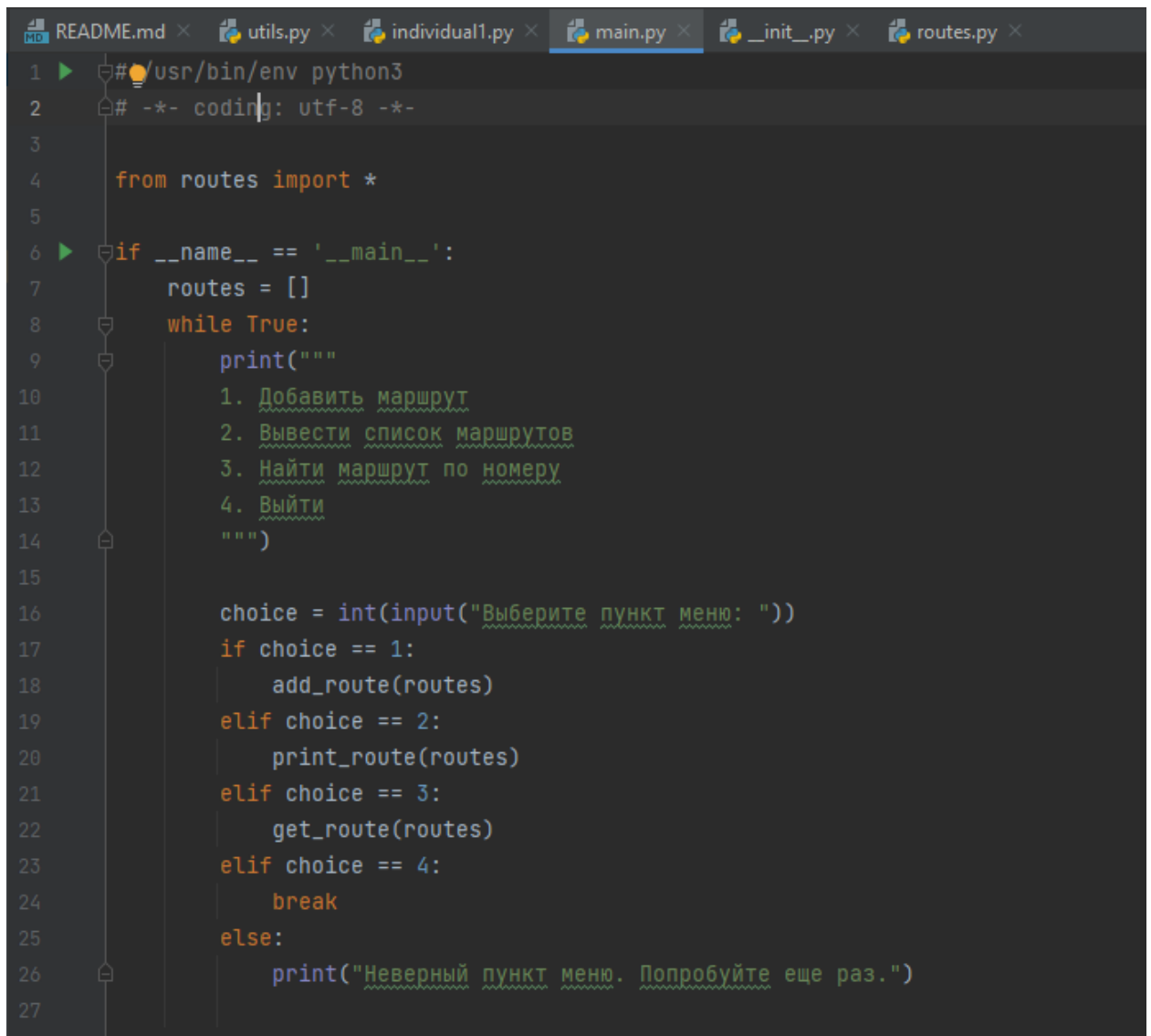
**Условие задание 2.6:** Использовать словарь, содержащий следующие ключи: название начального пункта маршрута; название конечного пункта маршрута; номер маршрута. Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в список, состоящий из словарей заданной структуры; записи должны быть упорядочены по номерам маршрутов; вывод на экран информации о маршруте, номер которого введен с клавиатуры; если таких маршрутов нет, выдать на дисплей соответствующее сообщение.

```
1 # Функция добавления маршрута в список
2 def add_route(route_list):
3     route = {}
4     route['start'] = input("Введите начальный пункт маршрута: ")
5     route['end'] = input("Введите конечный пункт маршрута: ")
6     route['number'] = int(input("Введите номер маршрута: "))
7
8     # Определяем индекс, на котором нужно вставить маршрут в список
9     index = 0
10    for i in range(len(route_list)):
11        if route['number'] > route_list[i]['number']:
12            index = i + 1
13        else:
14            break
15
16    # Добавляем маршрут в список
17    route_list.insert(index, route)
18    print("Маршрут добавлен.")
19
20
21 # Функция вывода списка маршрутов на экран
22 def print_route(route_list):
23     print("Список маршрутов:")
24     for route in route_list:
25         print(f"{route['number']}. {route['start']} - {route['end']}")
26
27
28 # Функция поиска маршрута по номеру
29 def get_route(route_list):
30     number = int(input("Введите номер маршрута: "))
31     for route in route_list:
32         if route['number'] == number:
33             print(f"Маршрут {number}: {route['start']} - {route['end']}")
34             return
35     print("Маршрут не найден.")
```

Рисунок 5. Модуль routes.py

```
1 __all__ = ['add_route', 'print_route', 'get_route']
2
3 from .routes import *
```

Рисунок 6. Модуль \_\_init\_\_.py



```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 from routes import *
5
6 if __name__ == '__main__':
7     routes = []
8     while True:
9         print("""
10             1. Добавить маршрут
11             2. Вывести список маршрутов
12             3. Найти маршрут по номеру
13             4. Выйти
14             """)
15
16         choice = int(input("Выберите пункт меню: "))
17         if choice == 1:
18             add_route(routes)
19         elif choice == 2:
20             print_route(routes)
21         elif choice == 3:
22             get_route(routes)
23         elif choice == 4:
24             break
25         else:
26             print("Неверный пункт меню. Попробуйте еще раз.")
27
```

Рисунок 7. Модуль main.py

### Задание 5.

После выполнения работы на ветке develop, слил ее с веткой main и отправил изменения на удаленный сервер.

```

C:\Users\Gaming-PC\Laba_2.13>git merge develop
Updating f8f5fb7..b00ec64
Fast-forward
 .idea/.gitignore          | 3 +++
 .idea/Laba_2.13.iml       | 10 ++++++++
 .idea/inspectionProfiles/Project_Default.xml | 12 ++++++++
 .idea/inspectionProfiles/profiles_settings.xml | 6 +++++
 .idea/misc.xml           | 4 +++
 .idea/modules.xml        | 8 ++++++
 .idea/vcs.xml            | 6 +++++
 individual1.py           | 1 +
 route_package/__init__.py | 3 +++
 route_package/main.py    | 26 ++++++++
 route_package/routes.py  | 35 ++++++++
 utils.py                 | 7 +++++
12 files changed, 121 insertions(+)
create mode 100644 .idea/.gitignore
create mode 100644 .idea/Laba_2.13.iml
create mode 100644 .idea/inspectionProfiles/Project_Default.xml
create mode 100644 .idea/inspectionProfiles/profiles_settings.xml
create mode 100644 .idea/misc.xml
create mode 100644 .idea/modules.xml
create mode 100644 .idea/vcs.xml
create mode 100644 individual1.py
create mode 100644 route_package/__init__.py
create mode 100644 route_package/main.py
create mode 100644 route_package/routes.py
create mode 100644 utils.py

```

Рисунок 5. Слияние ветки develop и main

Ссылка: [https://github.com/EvgenyEvdakov/Laba\\_2.13](https://github.com/EvgenyEvdakov/Laba_2.13)

### Ответы на контрольные вопросы:

#### 1. Что является модулем языка Python?

В языке Python модуль - это файл, содержащий Python код, который может быть импортирован и использован в других программах. Модуль предоставляет удобный способ организации кода, разделяя его на логические компоненты.

#### 2. Какие существуют способы подключения модулей в языке Python?

В языке Python существует несколько способов подключения модулей:

- 1) Использование ключевого слова `import`;
- 2) Использование ключевого слова `from`;
- 3) Использование ключевого слова `import` с псевдонимом;

#### 3. Что является пакетом языка Python?

В языке Python пакет - это способ организации и структурирования модулей в иерархическую структуру. Пакет представляет собой директорию,

которая содержит один или несколько модулей и файл `__init__.py`, который указывает на то, что директория является пакетом.

#### 4. Каково назначение файла `__init__.py` ?

Файл `__init__.py` является специальным файлом в пакете Python. Его наличие в директории указывает на то, что данная директория является пакетом. Файл `__init__.py` может быть пустым или содержать код, который будет выполнен при импорте пакета.

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

Переменная `__all__` в файле `__init__.py` пакета используется для определения списка модулей, которые будут импортированы при использовании команды `from package_name import *`.

**Вывод:** приобрел навыки по работе с модулями и пакетами языка программирования Python версии 3.x.