

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 №16**  
**дисциплины «Программирование на Python»**

**Вариант 7**

Выполнил:  
Кулешов Олег Иванович  
2 курс, группа ИВТ-б-о-22-1,  
09.03.01 «Информатика и  
вычислительная техника»,  
направленность (профиль)  
«Программное обеспечение средств  
вычислительной техники и  
автоматизированных систем», очная  
форма обучения

---

(подпись)

Руководитель практики:  
Воронкин Р. А.

---

(подпись)

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

Ставрополь, 2023

## Тема: Модули и пакеты

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

**Индивидуальное 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`.

### Листинг модуля `geometry.py`:

```
def calculate_area(type=0):  
    def inner_function(x, y):  
        if type == 0:  
            area = 0.5 * x * y  
        else:  
            area = x * y  
        return area  
    return inner_function
```

### Листинг программы:

```
#!/usr/bin/env python3  
# -*- coding: utf-8 -*-  
  
from geometry import calculate_area  
  
if __name__ == '__main__':  
    # Вызов функции замыкания с параметром type = 0 для вычисления площади  
    # треугольника  
    triangle_area = calculate_area(0)  
    result_triangle = triangle_area(5, 10)  
    print("Площадь треугольника:", result_triangle)  
  
    # Вызов функции замыкания с параметром type = 1 для вычисления площади  
    # прямоугольника  
    rectangle_area = calculate_area(1)  
    result_rectangle = rectangle_area(5, 10)  
    print("Площадь прямоугольника:", result_rec
```

```
C:\Users\User\PycharmProjects\Python_laba_16\venv\Scripts\python.exe  
Площадь треугольника: 25.0  
Площадь прямоугольника: 50  
  
Process finished with exit code 0
```

Рисунок 1. Результат программы

**Индивидуальное 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.

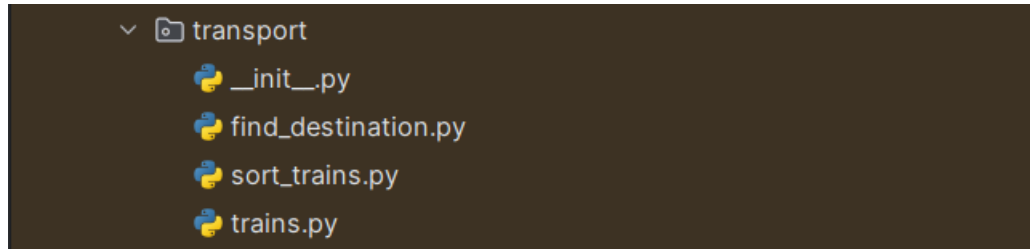


Рисунок 2. Созданный пакет

#### Листинг `find_destantion.py`:

```
def find_trains_by_destination(trains, destination):
    destination_trains = [train for train in trains if train.destination ==
destination]
    return destination_trai
```

#### Листинг `sort_trains.py`:

```
def sort_trains(trains):
    trains.sort(key=lambda x: x.departure_time)
```

#### Листинг `trains.py`:

```
class Train:
    def __init__(self, destination, train_number, departure_time):
        self.destination = destination
        self.train_number = train_number
        self.departure_time = departure_time

    def __str__(self):
        return f"Поезд номер {self.train_number} с пунктом назначения
{self.destination} отправляется в {self.departure_time}"
```

#### Листинг `__init__.py`:

```
__all__ = ['trains', 'sort_trains', 'find_destination']
```

#### Листинг программы:

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

from transport.trains import Train
from transport.sort_trains import sort_trains
from transport.find_destination import find_trains_by_destination
```

```

if __name__ == '__main__':
    # Создаем пустой список для хранения поездов
    trains = []

    # Ввод данных с клавиатуры
    n = int(input("Введите количество поездов: "))
    for i in range(n):
        destination = input("Введите название пункта назначения: ")
        train_number = input("Введите номер поезда: ")
        departure_time = input("Введите время отправления: ")
        train = Train(destination, train_number, departure_time)
        trains.append(train)

    # Сортировка списка по времени отправления поезда
    sort_trains(trains)

    # Вывод информации о поездах, направляющихся в указанный пункт назначения
    destination_input = input("Введите название пункта назначения: ")
    destination_trains = find_trains_by_destination(trains,
destination_input)
    if destination_trains:
        for train in destination_trains:
            print(train)
    else:
        print("Поездов в указанный пункт назначения нет!")

```

**Вывод:** в ходе выполнения данной лабораторной работы были приобретены навыки по взаимодействию с модулями и пакетами языка программирования Python версии 3.x.

### Ответы на контрольные вопросы

1. Модуль в языке Python - это файл, содержащий код, который может быть использован в других программах.

2. Существуют несколько способов подключения модулей в языке Python:

- Использование ключевого слова `import` для импорта модуля целиком: `import module\_name`.

- Использование ключевого слова `from` для импорта определенных элементов модуля: `from module\_name import element\_name`.

- Импорт модуля с псевдонимом: `import module\_name as alias\_name`.

- Импорт всех элементов модуля: `from module\_name import \*` (не рекомендуется из-за возможных конфликтов имен).

3. Пакет в языке Python - это специальная форма модуля, содержащая другие модули и пакеты. Пакет представляет собой директорию, содержащую

файл `__init__.py`, который указывает интерпретатору Python, что эта директория является пакетом.

4. Файл `__init__.py` в пакете языка Python выполняет несколько функций:

- Позволяет интерпретатору Python распознать директорию как пакет.
- Может содержать код инициализации пакета.
- Может определять переменные, функции и классы, которые будут доступны при импорте пакета.

5. Переменная `__all__` в файле `__init__.py` используется для определения списка импортируемых элементов при использовании `from package_name import *`. Если переменная `__all__` не определена или равна `None`, то при использовании `from package_name import *` будут импортированы только те элементы, которые перечислены в переменной `__all__` модуля.