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 23**

|  |  |  |
| --- | --- | --- |
|  | Выполнил:  Омонкулов Исомиддин Валижон угли  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, 2024

**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

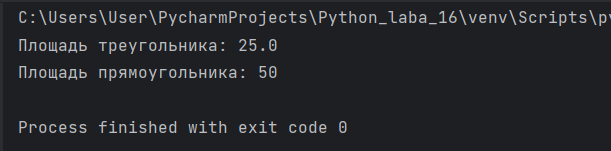


Рисунок 1. Результат программы

**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.

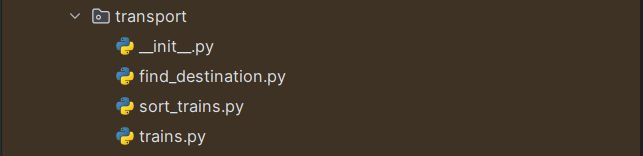


Рисунок 2. Созданный пакет

**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.