

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЁТ

по лабораторной работе №2.13

Дисциплина: «Программирование на Python»

Тема: «Модули и пакеты»

Выполнил:

Епифанов Алексей Александрович

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 Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____
Ставрополь, 2023 г.

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

Порядок выполнения работы:

1. Создал новый репозиторий, клонировал его, в нем создал ветку developer и перешел на нее.
2. Выполнил индивидуальное задание 1 вариант 10: 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 delete.py:

```
def del_items(type_param='even' ):
    """
    Выбирается тип элементов для удаления.
    Можно выбрать 'even' - удалятся четные;
    иначе не четные. По умолчанию стоит 'even'.
    """
    def delete(list_items):
        """
        Функция удалет некоторые элементы массива в зависимости от
        type_param.
        Type_param задается во внешней функции.
        """
        match type_param:
            case 'even':
                return list(filter(lambda x: x % 2 != 0, list_items))
            case _:
                return list(filter(lambda x: x % 2 == 0, list_items))
    return delete
```

Код программы ind1.py:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
from delete import del_items
if __name__ == "__main__":
    start_items = [0, 1, 2, 3, 4, 5, 23, 21, 14, 72]
    print(f"\n{start_items=}\n")
    del_even = del_items()
    print(f"{del_even(start_items)=}\n")
```

```
del_not_even = del_items('not even')
print(f"{del_not_even(start_items)=}\n")
```

```
пары/пары_3_сем/pytgit/Lab_16_py/program/ind1.py
start_items=[0, 1, 2, 3, 4, 5, 23, 21, 14, 72]
del_even(start_items)=[1, 3, 5, 23, 21]
del_not_even(start_items)=[0, 2, 4, 14, 72]
```

Рисунок 1. Вывод программы ind1.py

3. Выполнил индивидуальное задание 2 вариант 10: 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.

Код программы ind2.py:

```
from package_ind2 import *
import bisect
import re
import sys
display_routes = display.display_routes
get_route = get.get_route
select_routes = select.select_routes
def main():
    """
    Главная функция программы.
    """
    routes = []
    while True:
        command = input(">>> ").lower()
        match command:
            case 'exit':
                break
            case 'add':
                route = get_route()
                if route not in routes:
                    bisect.insort(
                        routes, route, key=lambda item: item.get(
                            'номер маршрута'
```

```

    )
    )
    else:
        print("Данный маршрут уже добавлен.")
    case 'list':
        display_routes(routes)
    case _ if (m := re.match(r'select (.+)', command)):
        name_punct = m.group(1)
        selected = select_routes(routes, name_punct)
        display_routes(selected)
    case 'help':
        print("Список команд:\n")
        print("add - добавить маршрут;")
        print("list - вывести список маршрутов;")
        print(
            "select <название пункта> - запросить маршруты, которые
начинаются\n"
            "или заканчиваются в данном пункте;"
        )
        print("help - отобразить справку;")
        print("exit - завершить работу с программой.")
    case _:
        print(f"Неизвестная команда {command}", file=sys.stderr)
if __name__ == '__main__':
    main()

```

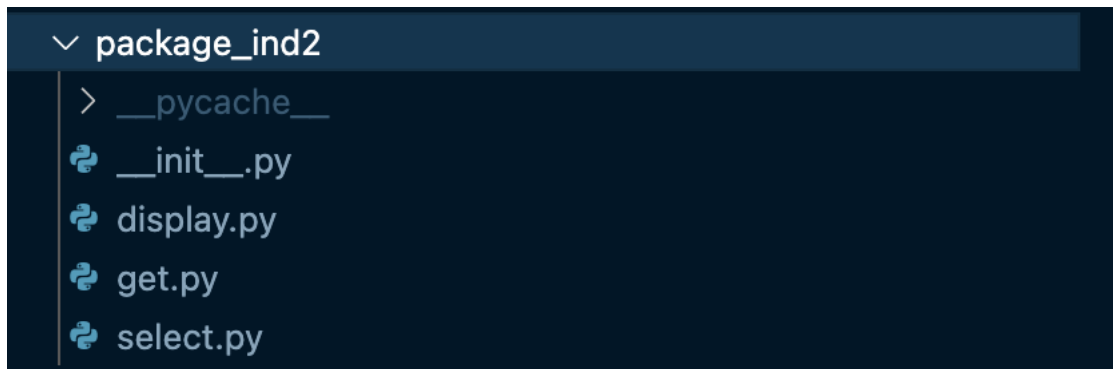


Рисунок 2. Структура пакета

Ответы на контрольные вопросы:

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.

2. Какие существуют способы подключения модулей в языке Python?

```
import имя_модуля; import имя_модуля1, имя_модуля2;  
import имя_модуля as новое_имя; from имя_модуля import имя_объекта;  
from имя_модуля import имя_объекта1, имя_объекта2;  
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`. Пакеты используются для формирования пространства имен, что позволяет работать с модулями через указание уровня вложенности (через точку).

4. Каково назначение файла `__init__.py` ?

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

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

В переменную `__all__` вносятся названия модулей, которые импортируются при `from имя_пакета import *`.

Вывод: в результате выполнения работы были приобретены навыки по работе с модулями и пакетами языка программирования Python версии 3х.