

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»**

Выполнила:  
Мурашко Анастасия Юрьевна  
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:**

**Вариант №9**

**Индивидуальное задание:**

## 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`. Номер варианта уточнить у преподавателя.

Рисунок 1. Условия задания

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  from helpiz1 import delchis
5
6  f = delchis(list1=[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14])
7  f()
```

Рисунок 2. Код

```
C:\Users\MonstR\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/U
Введите even или noteven:
even
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
[1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
[1, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
[1, 3, 5, 7, 8, 9, 10, 11, 12, 13, 14]
[1, 3, 5, 7, 9, 10, 11, 12, 13, 14]
[1, 3, 5, 7, 9, 11, 12, 13, 14]
[1, 3, 5, 7, 9, 11, 13, 14]
[1, 3, 5, 7, 9, 11, 13]
```

Рисунок 3. Результат выполнения

```

1  def delchis(list1):
2      def delchetnechet():
3          type = input('Введите even или noteven:\n')
4          if type == 'even':
5              for x in range(len(list1)):
6                  if x % 2 == 0:
7                      list1.pop(list1.index(x))
8                      print(list1)
9              elif type == 'noteven':
10                 for x in range(len(list1)):
11                     if x % 2 != 0:
12                         list1.pop(list1.index(x))
13                         print(list1)
14             else:
15                 print('Введена не правильная команда')
16
17     return delchetnechet

```

Рисунок 4. Модуль 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. Номер варианта уточнить у преподавателя.

Рисунок 5. Условия задания

```

1  import sys
2  # import inf
3  from inf import table, table_name, table_name_fil
4
5  list_shop = []
6  spisok_new = []
7
8  while True:
9      command = input('>>> ').lower()
10
11      if command == 'exit':
12          break
13
14      elif command == 'add':
15          name_shop = input('Название магазина: ')
16          name_product = input('Название товара: ')
17          prise = input('Стоимость товара: ')
18
19          list_shop_new = {
20              'name_shop': name_shop,
21              'name_product': name_product,
22              'prise': prise
23          }
24
25          list_shop.append(list_shop_new)
26
27          if len(list_shop) > 1:
28              list_shop.sort(key=lambda item: item.get('name_shop', ''))
29
30      elif command == 'list':
31          print(table())
32          print(table_name())

```

```

        if len(list_shop) > 1:
            list_shop.sort(key=lambda item: item.get('name_shop', ''))

    elif command == 'list':
        print(table())
        print(table_name())
        print(table())
        for item_n in table_name_fil(list_shop):
            print(item_n)
        print(table())

    elif command == 'product':
        shop_sear = input('Введите название товара: ')
        search_shop = []
        for shop_sear_itme in list_shop:
            if shop_sear_itme.get('name_goods', '') == shop_sear:
                search_shop.append(shop_sear_itme)

        if len(search_shop) > 0:
            print(search_shop)
        else:
            print('Такого товара не найдено', file=sys.stderr)

    elif command == 'help':
        print('Список команд:\n')
        print('add - добавить магазин.')
        print('list - вывести список магазинов.')
        print('product <Название> - запросить информацию о товаре.')
        print('help - Справочник.')
        print('exit - Завершить пработу программы.')
    else:
        print(f'Команда <{command}> не существует.', file=sys.stderr)
        print('Введите <help> для просмотра доступных команд')

```

Рисунок 6, 7. Код

```

C:\Users\MonstR\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/MonstR/Desktop/3/lb3iz2.py
>>> add
Название магазина: фыаффф
Название товара: цйуцк
Стоимость товара: 1234
>>> list
+-----+-----+-----+-----+
| №      | Название магазина | Название товара | Стоимость |
+-----+-----+-----+-----+
| 1      | фыаффф           | цйуцк          | 1234      |
+-----+-----+-----+-----+
>>> |

```

Рисунок 8. Результат выполнения

```

1 from .t import *
2 from .tn import *
3 from .tnf import *

```

Рисунок 9. Пакет задания 2

```
def table():
    line = '+--{}--{}--{}--{}--+'.format(
        '-' * 8,
        '-' * 30,
        '-' * 45,
        '-' * 30
    )
    return line
```

```
def table_name():
    post = '| {:^8} | {:^30} | {:^45} | {:^30} | '.format(
        "№",
        "Название магазина",
        "Название товара",
        "Стоимость"
    )
    return post
```

```
def table_name_fil(names):
    post = []
    for idx_new, spisok_new_new in enumerate(names, 1):
        post.append(
            '| {:>8} | {:<30} | {:<45} | {:<30} | '.format(
                idx_new,
                spisok_new_new.get('name_shop', ''),
                spisok_new_new.get('name_product', ''),
                spisok_new_new.get('prise', '')
            )
        )
    return post
```

Рисунок 10, 11, 12. Модули задания 2

### 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?

Самый простой способ - `import имя_модуля`. Способ импортировать сразу несколько модулей - `import имя_модуля1, имя_модуля2`. Способ с созданием псевдонима для модуля - `import имя_модуля as новое_имя`. Способ чтобы не указывать постоянно имя модуля (или псевдоним) - `from имя_модуля import`

имя\_объекта.

### 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` ?

В `__init__.py` файл заставляет Python рассматривать каталоги, содержащие его, как модули. Кроме того, это первый файл, загружаемый в модуль, поэтому вы можете использовать его для выполнения кода, который хотите запускать каждый раз при загрузке модуля, или для указания экспортируемых подмодулей.

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

Хранит список модулей, который импортируется при загрузке через конструкцию.

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