

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

Институт цифрового развития
Кафедра инфокоммуникаций

**«Лабораторная работа 2.13 Модули и
пакеты»**

**ОТЧЕТ
по лабораторной работе №16
дисциплины
«Основы программной инженерии»**

Выполнил:

Луценко Дмитрий Андреевич
2 курс, группа ПИЖ-б-о-21-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка
и сопровождение программного
обеспечения», очная форма обучения

(подпись)

Проверил:

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2022 г.

Лабораторная работа 2.13 Модули и пакеты

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

Ход работы:

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.

```
1  ▶  #!/usr/bin/env python3
2      # -*- coding: utf-8 -*-
3
4      import function
5
6
7  ▶  if __name__ == "__main__":
8      test_fun = function.fun1(2)
9      print(test_fun(3))
```

Рисунок 1 – Код программы для задания 1

```
1      import math
2
3
4      def fun1(a):
5          def fun2(b):
6              f = a * math.pow(b, a)
7              l = f"Для значений {a}, {b} функция f(a,b) = {f}"
8              return l
9
10     return fun2
```

```
E:\GitHub\laba15\user\Scripts\pyth
Площадь круга равна: 28.27

Process finished with exit code 0
```

Рисунок 2 – Результат работы программы

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.

```
1  ▶ #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  from package import main
5
6  ⚡
7  ▶ if __name__ == '__main__':
8      main.main()
9
```

Рисунок 3 – Основной код программы

```
1task.py × function.py × 2task.py × __init__.py × add.py × list.py × select.py × help.py × error.py × main.py ×
1  __all__ = ["add", "list", "select", "help", "error", "main"]
    ⚡
```

Рисунок 4 – Содержание файла `__init__.py`

```

1  def add_product():
2      """
3      Ввод информации о товарах.
4      """
5      prod = input("Введите название товара: ")
6      shop = input("Введите название магазина: ")
7      cost = float(input("Введите стоимость товара: "))
8
9      return {
10         'product': prod,
11         'shop': shop,
12         'cost': cost
13     }
14

```

Рисунок 5 – Модуль add

```

1  def product_list(products):
2      """
3      Вывод списка товаров
4      """
5      line = '+-{}-+-{}-+-{}-+'.format(
6          '-' * 4,
7          '-' * 30,
8          '-' * 20
9      )
10     print(line)
11     print(
12         '| {:^25} | {:^15} | {:^14} |'.format(
13             "Товар",
14             "Магазин",
15             "Стоимость"
16         )
17     )
18     print(line)
19
20     for product in products:
21         print(
22             '| {:^25} | {:^15} | {:^14} |'.format(
23                 product.get('product', ''),
24                 product.get('shop', ''),
25                 product.get('cost', 0)
26             )
27         )
28     print(line)

```

Рисунок 6 – Модуль list

```

1  def select(products, shop):
2      """
3      Выбрать товары из конкретного магазина.
4      """
5      result = []
6      for product in products:
7          if product.get('shop', '') == shop:
8              result.append(product)
9      return result
10

```

Рисунок 7 – Модуль select

```

1  def get_help():
2      print("Список команд:\n")
3      print("add - добавить информацию о товаре;")
4      print("list - вывести список товаров;")
5      print("select - запросить товары из одного магазина;")
6      print("help - отобразить справку;")
7      print("exit - завершить работу с программой.")
8

```

Рисунок 8 – Модуль help

```

1  def error(command):
2      print(f"Неизвестная команда {command}", file=sys.stderr)

```

Рисунок 9 – Модуль error

```

9      """
10     Главная функция программы.
11     """
12     products = []
13
14     while True:
15         command = input(">>> ").lower()
16
17         if command == 'exit':
18             break
19
20         elif command == 'add':
21             product = add.add_product()
22             products.append(product)
23             if len(products) > 1:
24                 products.sort(key=lambda item: item.get('shop', ''))
25
26         elif command == 'list':
27             list.product_list(products)
28
29         elif command == 'select':
30             sel_shop = input("Введите магазин: ")
31             selected = select.select(products, sel_shop)
32             list.product_list(selected)
33
34         elif command == 'help':
35             help.get_help()
36
37     else:

```

Рисунок 10 – Модуль main

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

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.

2. Какие существуют способы подключения модулей в языке Python? Есть несколько способов подключения модулей:

`import имя_модуля`

`import имя_модуля as новое_имя` если хотим заменить название модуля псевдонимом внутри программы

`from имя_модуля import имя_объекта` если хотим импортировать отдельный объект.

`from имя_модуля import *` если хотим импортировать всё

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`.

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` ? Переменная `__all__` хранит список модулей, импортирующихся при загрузке через конструкцию