

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 31**

Выполнил:  
Репкин Александр Павлович  
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., канд. техн. наук,  
доцент кафедры инфокоммуникаций

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_  
Ставрополь, 2023 г.

**Тема:** Модули и пакеты в языке Python.

**Цель:** приобрести навыки работы с модулями и пакетами при написании программ с помощью языка программирования Python версии 3.x.

**Порядок выполнения работы:**

1. Выполнены примеры лабораторной работы. В них показана возможность импортирования модулей (А также обращения к импортируемому модулю по заданной переменной).

```
24
0.5000000000000001
0.8660254037844386
```

Рисунок 1. Полученный результат выполнения примеров.

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  from math import factorial as f, cos, sin, pi
5
6
7  if __name__ == '__main__':
8      print(f(4))
9      print(cos(pi/3))
10     print(sin(pi/3))
11
```

Рисунок 2. Код примеров.

2. Выполнено первое индивидуальное задание. В нём требовалось преобразовать код индивидуального задания из работы №14, переместив функции в отдельный модуль и подключив его при помощи `import`. Индивидуальное задание из работы №14: “Используя замыкания функций, определить вложенную функцию, увеличивающую значение переданного ей параметра на 3. Вычисленный результат возвращается. Задание также требовало присвоение переменной `cnt` ссылки на внутреннюю функцию при помощи вызова внешней функции. Затем, через переменную `cnt` вызывается внутренняя функция со значением `k`, введенным с клавиатуры”.

```
Good day! Please, enter value: 6
According to our calculations, answer is - 9
```

Рисунок 3. Полученный результат первого индивидуального задания.

```

1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  import calculation
5
6
7  """Внутренняя функция увеличивает полученное значение на 3."""
8
9  if __name__ == '__main__':
10     cnt = calculation.calculaion()
11     k = int(input("Good day! Please, enter value: "))
12     print("According to our calculations, answer is - ", cnt(k))
13

```

Рисунок 4. Полученный код первого индивидуального задания.

3. Выполнено второе индивидуальное задание. В нём требовалось преобразовать код индивидуального задания из работы №11, переместив функции в отдельный пакет и подключив его при помощи `import`. Индивидуальное задание из работы №11: “Преобразовать код из лабораторной работы №9, внедрив функции (В индивидуальном задании 9 лабораторной работы создавался словарь, хранящий информацию о людях – фамилия, имя, телефонный номер, дата рождения. Требовалось обрабатывать несколько команд – вывод всех людей (`list`), вывод сводки о имеющихся командах (`help`), добавление человека (`add`), вывод людей относительно требуемого месяца рождения (`select “месяц”`), выход из программы (`exit`))”.

```

Good day!, please, enter your command: ('help' will list all of them)
>>> help
List of commands:
add - adding the human;
list - list of known people;
select 'month' - list of people, born in needed month;
help - list of commands;
exit - exit this program.
>>> add
Name - Totoro
Surname - Under_Leaves
Telephone number - 83573578357
Date of birth (Day.Month.Year) - 16.04.1988
>>> list

```

№	Name	Surname	Telephone	Birthday
1	Totoro	Under_Leaves	83573578357	16.04.1988

```

>>>

```

Рисунок 5. Полученный результат второго индивидуального задания.

```

1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  from needed_functions import *
5  import sys
6
7
8  def main():
9      """Главная функция программы."""
10
11     print("Good day!, please, enter your command: ('help' will list all of them)")
12     people = []
13     while True:
14         command = input(">>> ").lower()
15         if command == "exit":
16             break
17         elif command == "add":
18             human = new_human.new_human()
19             people.append(human)
20             # Сортировка идёт по фамилиям. Если фамилии одинаковые, то рассматриваются имена
21             people.sort(key=lambda person: (person.get(
22                 "surname", ""), person.get("name", "")))
23         elif command == "list":
24             display_people.display_people(people)
25         elif command.startswith("select "):
26             which_month = command.split(" ", maxsplit=1)
27             user_month = int(which_month[1])
28             same_month = select_people.select_people(people, user_month)
29             display_people.display_people(same_month)
30         elif command == "help":
31             list_commands.list_commands()
32         else:
33             print(f"Unknown command! - {command}", file=sys.stderr)
34
35
36 if __name__ == '__main__':
37     main()
38

```

Рисунок 6. Полученный код второго индивидуального задания.

#### 4. Ответы на вопросы.

##### 1) Что такое модуль языка Python?

**Ответ:** модуль - файл с расширением .py, предназначенный для хранения часто используемых функций, классов, констант и т.д. Можно условно разделить модули и программы: программы предназначены для непосредственного запуска, а модули для импортирования их в другие программы. Модули могут быть написаны не только на Python, но и на других языках (например C).

##### 2) Какие существуют способы подключения модулей в языке Python?

**Ответ:** для импортирования модуля можно воспользоваться конструкцией `import имя_модуля`. Если необходимо установить модулю псевдоним, то используется `import имя_модуля as новое_имя`. Если нужно импортировать только конкретные объекты, то используется `from имя_модуля import имя_объекта`, а если нужно импортировать все, то `from имя_модуля import *`.

##### 3) Что является пакетом языка Python?

**Ответ:** пакет - 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). Для импортирования пакетов используется тот же синтаксис, что и для работы с модулями.

4) Каково назначение файла `__init__.py`?

**Ответ:** файл `__init__.py` может быть пустым или может содержать переменную `__all__`, хранящую список модулей, который импортируется при загрузке через конструкцию `from имя_пакета import *`.

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

**Ответ:** переменная `__all__` хранит список модулей, который импортируется при загрузке через конструкцию `from имя_пакета import *`.

**Вывод:** в ходе выполнения лабораторной работы были приобретены навыки работы с модулями и пакетами при написании программ с помощью языка программирования Python версии 3.x.