

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ
УНИВЕРСИТЕТ» ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ**

**Отчет по лабораторной работе №16 по
дисциплине: основы программной инженерии**

Выполнил:

студент группы ПИЖ-б-о-21-1

Прокопов Дмитрий Владиславович

Проверил:

доцент кафедры инфокоммуникаций

Воронкин Р.А.

Ставрополь, 2022 г.

ВЫПОЛНЕНИЕ

1-я индивидуальная работа

```
def con(type: str):
    def act(nums: str):
        if type == 'list':
            return list(map(int, nums.split(' ')))
        if type == 'tuple':
            return tuple(map(int, nums.split(' ')))
    return act

ind1.py
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
import ind_2_11

print(ind_2_11.con('list')('3 7 5 9 1'))
print(ind_2_11.con('tuple')('5 7 1 3 6 8'))

ind1 x
C:\Users\Dmitriy\PycharmProjects\lb16\venv\Scripts\python.exe C:/User
[3, 7, 5, 9, 1]
(5, 7, 1, 3, 6, 8)
```

2-я индивидуальная работа

__init__.py

```
__all__ = ["main", "display_human", "find_human", "get_human"]
```

display_human.py

```
def display_human(staff):
    """
    Отобразить список работников.
    """
    # Проверить, что список работников не пуст.
    if staff:
        # Заголовок таблицы.
        line = '+-{}-+-{}-+-{}-+-{}-+'.format(
            '-' * 4,
            '-' * 30,
            '-' * 15,
            '-' * 15
        )
        print(line)
        print(
            '| {:^4} | {:^30} | {:^15} | {:^15} |'.format(
```

```

        "№",
        "Ф и И.",
        "Телефон",
        "День рождения"
    )
)
print(line)

# Вывести данные о всех сотрудниках.
for idx, human in enumerate(staff, 1):
    print(
        f'| {idx:>4} |'
        f' {human.get("name", ""):<30} |'
        f' {human.get("phone", 0):<15} |'
        f' {human.get("birthday")} |'
    )
    print(line)

else:
    print("Список пуст.")

```

find_human.py

```

def find_human(staff, fname):
    """
    Выбрать работников с заданным стажем.
    """
    # Сформировать список людей.
    result = []
    for h in staff:
        if fname in str(h.values()):
            result.append(h)

    # Проверка на наличие записей
    if len(result) == 0:
        return print("Запись не найдена")

    # Возвратить список выбранных работников.
    return result

```

get_human.py

```

import datetime

def get_human():
    """
    Запросить данные о работнике.
    """
    name = input("Фамилия и имя: ")
    phone = int(input("Номер телефона: +7"))
    bday = list(map(int, input("Дата рождения: ").split('.')))
    d_bday = datetime.date(bday[2], bday[1], bday[0])

    # Вернуть словарь.
    return {
        'name': name,
        'phone': phone,
        'birthday': d_bday
    }

```

main.py

```
import sys

from pak.find_human import find_human
from pak.get_human import get_human
from pak.display_human import display_human

def main():
    """
    Главная функция программы.
    """
    # Список работников.
    people = []

    # Организовать бесконечный цикл запроса команд.
    while True:
        # Запросить команду из терминала.
        command = input(">>> ").lower()

        # Выполнить действие в соответствие с командой.
        if command == 'exit':
            break

        elif command == 'add':
            # Запросить данные о работнике.
            human = get_human()

            # Добавить словарь в список.
            people.append(human)
            # Отсортировать список в случае необходимости.
            if len(people) > 1:
                people.sort(key=lambda item: item.get('phone', 0))

        elif command == 'list':
            # Отобразить всех работников.
            display_human(people)

        elif command == 'find':
            f = input('Введите фамилию: ')

            # Выбрать людей с заданной фамилией.
            selected = find_human(people, f)
            # Отобразить выбранных работников.
            display_human(selected)

        elif command == 'help':
            # Вывести справку о работе с программой.
            print("Список команд:\n")
            print("add - добавить человека;")
            print("list - вывести список людей;")
            print("find - найти человека по фамилии;")
            print("help - отобразить справку;")
            print("exit - завершить работу с программой.")

        else:
            print(f"Неизвестная команда {command}", file=sys.stderr)
```

Код основной программы:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
```

```
from pak.main import *
```

```
if __name__ == '__main__':
    main()
```

```
>>> help
```

Список команд:

add - добавить человека;

list - вывести список людей;

find - найти человека по фамилии;

help - отобразить справку;

exit - завершить работу с программой.

```
>>> add
```

Фамилия и имя: *Иванович Сергей*

Номер телефона: *+76568880011*

Дата рождения: *10.10.2000*

```
>>> add
```

Фамилия и имя: *Аверин Антон*

Номер телефона: *+71115250000*

Дата рождения: *3.01.1994*

```
>>> list
```

+-----+-----+-----+-----+			
№	Ф и И.	Телефон	День рождения
+-----+-----+-----+-----+			
1	Аверин Антон	1115250000	1994-01-03
+-----+-----+-----+-----+			
2	Иванович Сергей	6568880011	2000-10-10
+-----+-----+-----+-----+			

```
>>> find
```

Введите фамилию: *Аверин*

+-----+-----+-----+-----+			
№	Ф и И.	Телефон	День рождения
+-----+-----+-----+-----+			
1	Аверин Антон	1115250000	1994-01-03
+-----+-----+-----+-----+			

Вопросы для защиты работы

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. Стоит заметить, что модули могут быть написаны не только на языке Python, но и на других языках (например C).

2. Какие существуют способы подключения модулей в языке Python?

Самый простой способ импортировать модуль в Python это воспользоваться конструкцией:

```
import имя_модуля
```

За один раз можно импортировать сразу несколько модулей, для этого их нужно перечислить через запятую после слова `import`. Если вы хотите задать псевдоним для модуля в вашей программе, можно воспользоваться вот таким синтаксисом:

```
import имя_модуля as новое_имя
```

Для импортирования нескольких функций из модуля, можно перечислить их имена через запятую

```
from имя_модуля import имя_объекта1, имя_объекта2
```

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`

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

```
from имя_пакета import *
```