

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 №9**  
**дисциплины «Основы программной инженерии»**

Выполнил:  
Матвеев Александр Иванович  
2 курс, группа ПИЖ-б-о-22-1,  
09.03.04 «Программная инженерия»,  
направленность (профиль) «Разработка и  
сопровождение программного  
обеспечения», очная форма обучения

---

(подпись)

Проверил Воронкин Роман Александрович

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2023 г.

## Тема: Работа со словарями в языке Python

**Цель работы:** приобретение навыков по работе со словарями при написании программ с помощью языка программирования Python версии 3.x.

Ход работы.

### 1. Создание нового репозитория с лицензией MIT.

**Create a new repository**

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

*Required fields are marked with an asterisk (\*).*

**Owner \*** SashkaHacker / **Repository name \*** laba3  
✅ laba3 is available.

Great repository names are short and memorable. Need inspiration? How about [automatic-happiness](#) ?

**Description** (optional)  
Выполнение лабораторной работы №3 по дисциплине: основы программной инженерии

☒ **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**  
You choose who can see and commit to this repository.

**Initialize this repository with:**

☒ **Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

**Add .gitignore**  
.gitignore template: Python  
Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

**Choose a license**  
License: MIT License  
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

ⓘ You are creating a public repository in your personal account.

**Create repository**

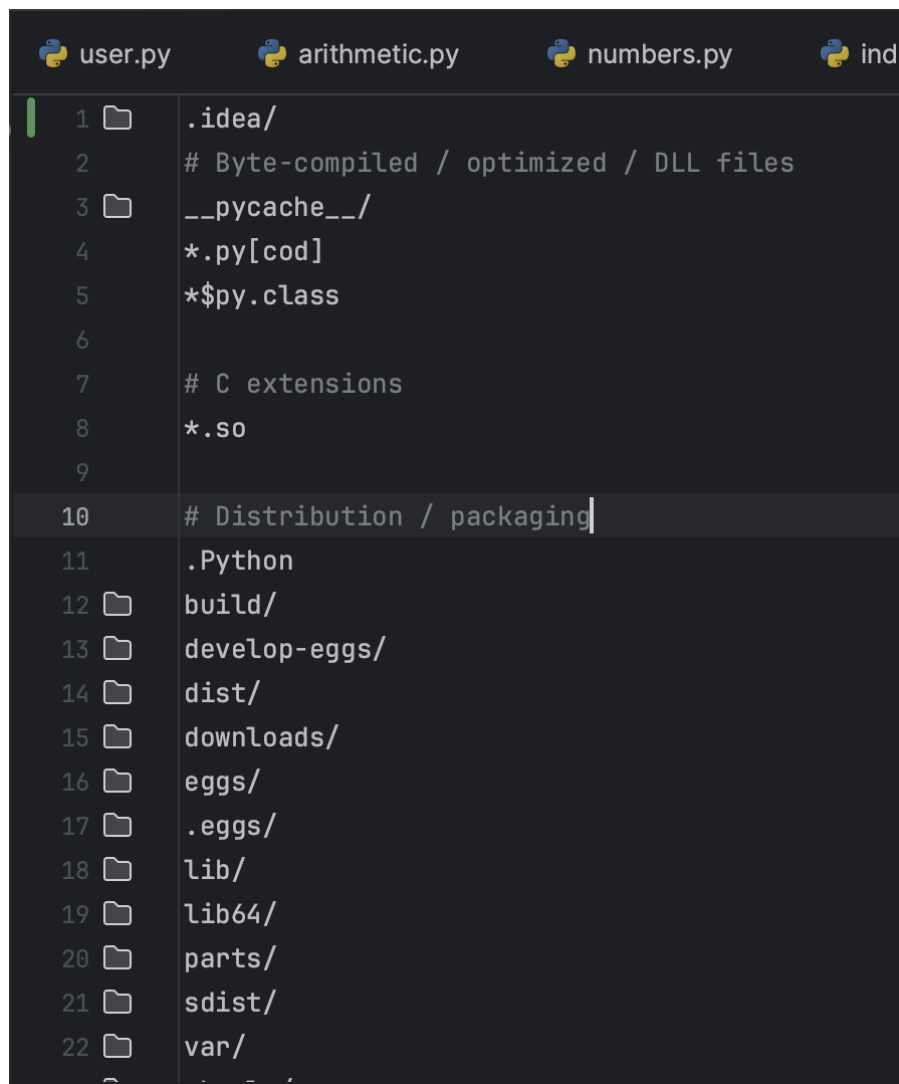
Рисунок 1 – Создание репозитория

### 2. Клонировал репозиторий на рабочий ПК.

```
Sashka@DESKTOP-U4RPSBI MINGW64 ~/Documents/GitHub/laba6 (develop)
$ git push origin develop
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/SashkaHacker/laba6/pull/new/develop
remote:
To https://github.com/SashkaHacker/laba6.git
 * [new branch]      develop -> develop
```

Рисунок 2 – Клонирование репозитория

3. Дополнил файл .gitignore необходимыми инструкциями.



```
user.py arithmetic.py numbers.py ind
1  .idea/
2  # Byte-compiled / optimized / DLL files
3  __pycache__/
4  *.py[cod]
5  *$py.class
6
7  # C extensions
8  *.so
9
10 # Distribution / packaging
11 .Python
12 build/
13 develop-eggs/
14 dist/
15 downloads/
16 eggs/
17 .eggs/
18 lib/
19 lib64/
20 parts/
21 sdist/
22 var/
```

Рисунок 3 – Файл .gitignore

4. Проработка задания №1 из методических указаний.

```
Run: primer1 x
C:\Users\Sashka\AppData\Local\Programs\Python\Python310\python.exe C:/Users/Sashka/D
>>> add
Фамилия и инициалы? Яков Тишин
Должность? slave
Год поступления? 2022
>>> add
Неизвестная команда add
add
Фамилия и инициалы? Billy H.Y
Должность? men
Год поступления? 2000
>>> list
+-----+-----+-----+-----+
| No |          Ф.И.О.          |      Должность      |      Год      |
+-----+-----+-----+-----+
|  1 | Billy H.Y                |      men            |      2000      |
|  2 | Яков Тишин               |      slave          |      2022      |
+-----+-----+-----+-----+
>>> select 1
1: Billy H.Y
2: Яков Тишин
>>> help
Список команд:

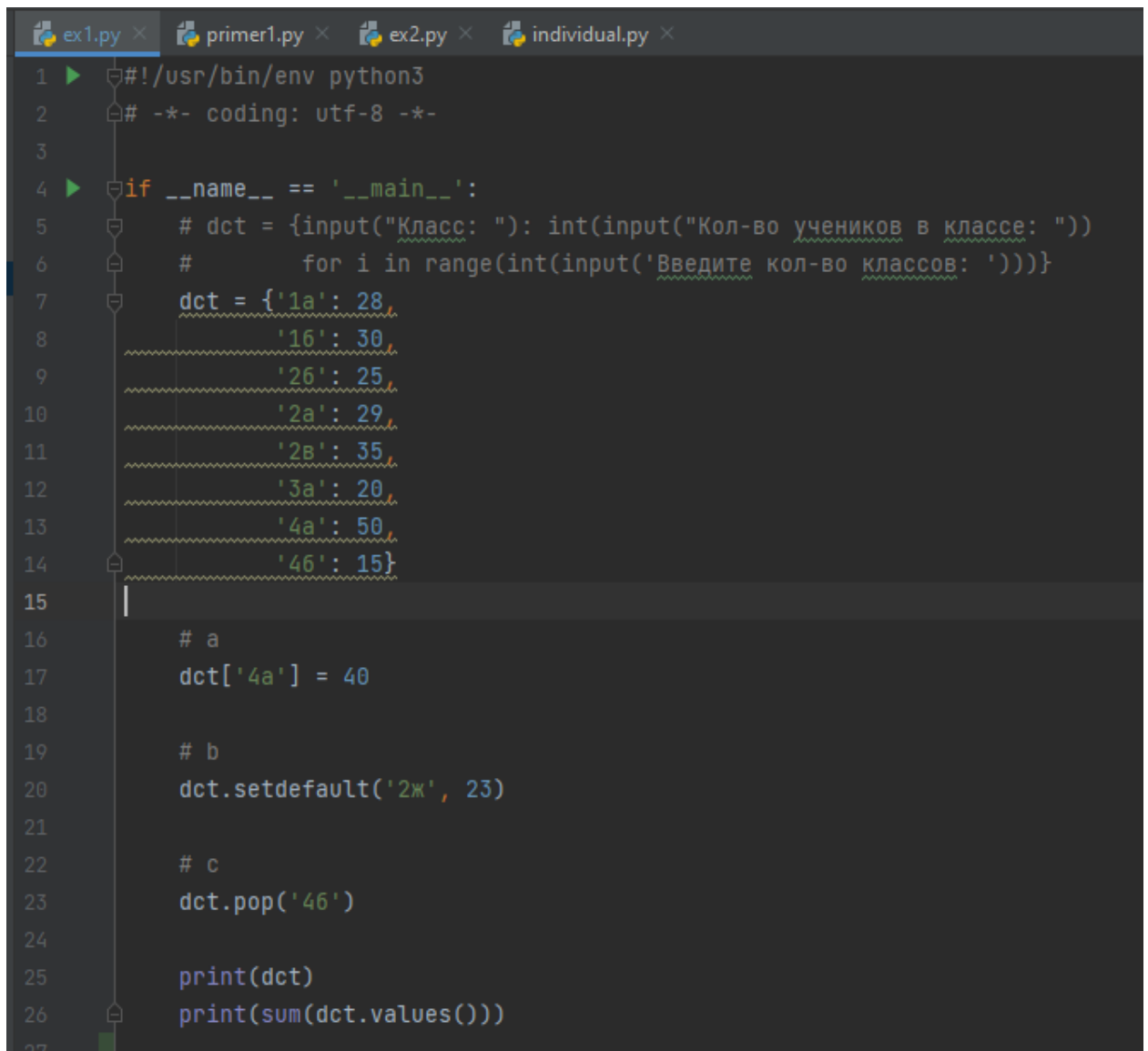
add - добавить работника;
list - вывести список работников;
select <стаж> - запросить работников со стажем;
help - отобразить справку;
exit - завершить работу с программой.
>>> exit

Process finished with exit code 0
```

Рисунок 4 – Пример работы программы

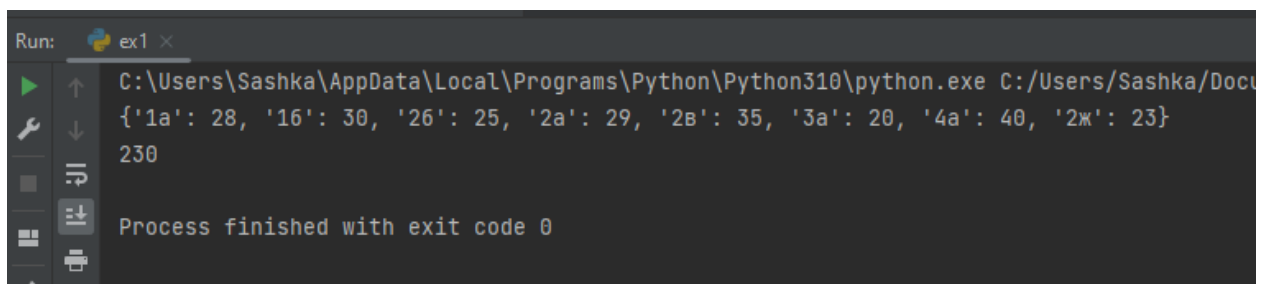
## 5. Выполнение задания №1.

Условие: имеются данные о сумме очков, набранных в чемпионате каждой из футбольных команд. Определить, перечислены ли команды в списке в соответствии с занятыми ими местами в чемпионате.



```
1  #!/usr/bin/env python3
2  #- coding: utf-8 -*-
3
4  if __name__ == '__main__':
5      # dct = {input("Класс: "): int(input("Кол-во учеников в классе: "))
6      #         for i in range(int(input('Введите кол-во классов: ')))}
7      dct = {'1a': 28,
8             '16': 30,
9             '26': 25,
10            '2a': 29,
11            '2в': 35,
12            '3a': 20,
13            '4a': 50,
14            '46': 15}
15
16      # a
17      dct['4a'] = 40
18
19      # b
20      dct.setdefault('2ж', 23)
21
22      # c
23      dct.pop('46')
24
25      print(dct)
26      print(sum(dct.values()))
27
```

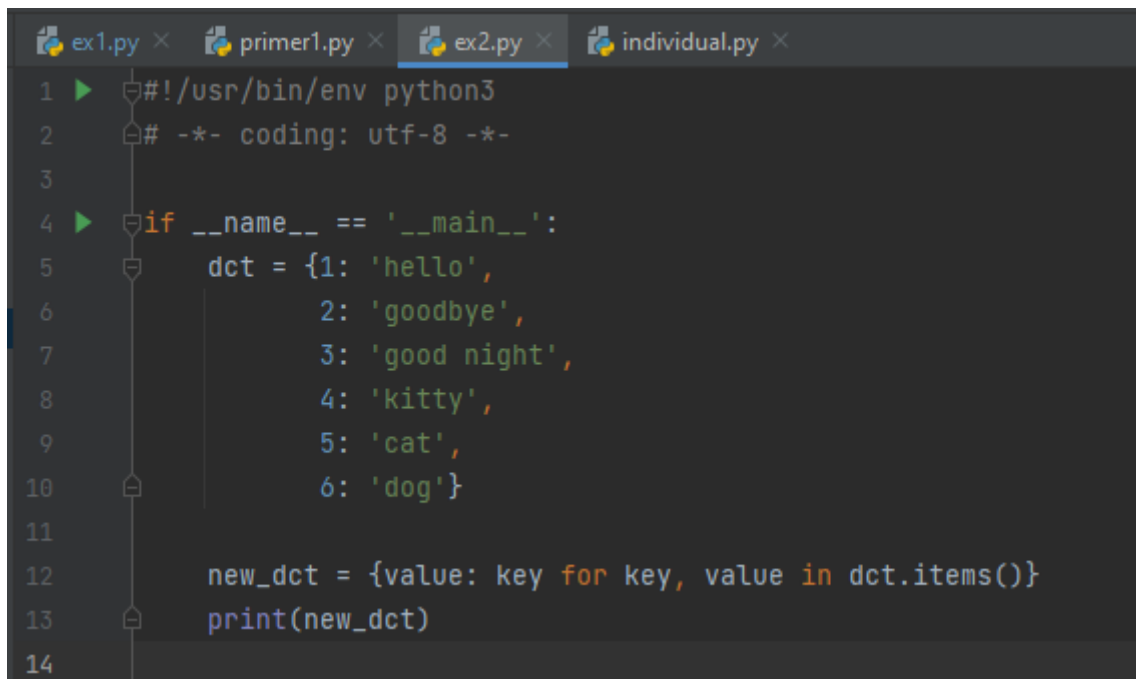
Рисунок 5 – Код программы



```
Run: ex1 x
C:\Users\Sashka\AppData\Local\Programs\Python\Python310\python.exe C:/Users/Sashka/Docu
{'1a': 28, '16': 30, '26': 25, '2a': 29, '2в': 35, '3a': 20, '4a': 40, '2ж': 23}
230
Process finished with exit code 0
```

Рисунок 7 – Пример выполнения программы

## 6. Выполнение задания №2.

A screenshot of a code editor with four tabs: ex1.py, primer1.py, ex2.py (selected), and individual.py. The code in ex2.py is as follows:

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 if __name__ == '__main__':
5     dct = {1: 'hello',
6            2: 'goodbye',
7            3: 'good night',
8            4: 'kitty',
9            5: 'cat',
10           6: 'dog'}
11
12     new_dct = {value: key for key, value in dct.items()}
13     print(new_dct)
14
```

Рисунок 8 – Код программы

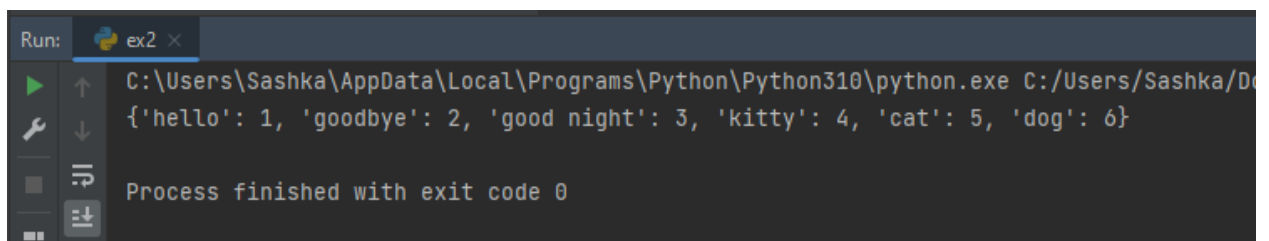
A screenshot of a terminal window titled 'Run: ex2'. It shows the command being executed and the output. The command is: `C:\Users\Sashka\AppData\Local\Programs\Python\Python310\python.exe C:/Users/Sashka/D`. The output is: `{'hello': 1, 'goodbye': 2, 'good night': 3, 'kitty': 4, 'cat': 5, 'dog': 6}`. Below the output, it says 'Process finished with exit code 0'.

Рисунок 9 – Пример выполнения программы

7. Выполнение индивидуального задания.

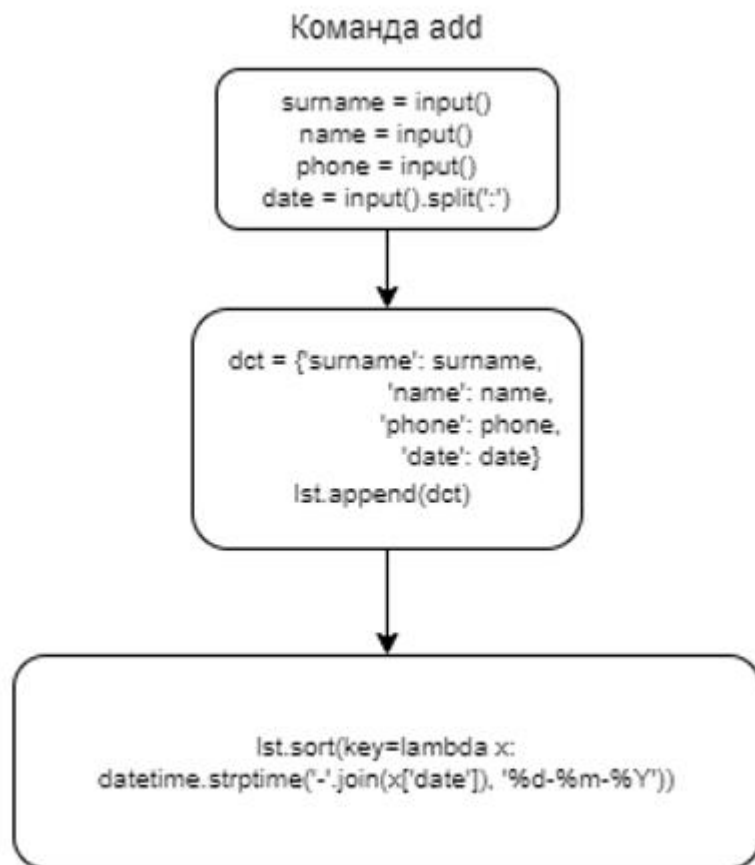


Рисунок 10 – UML-диаграмма для команды (add)

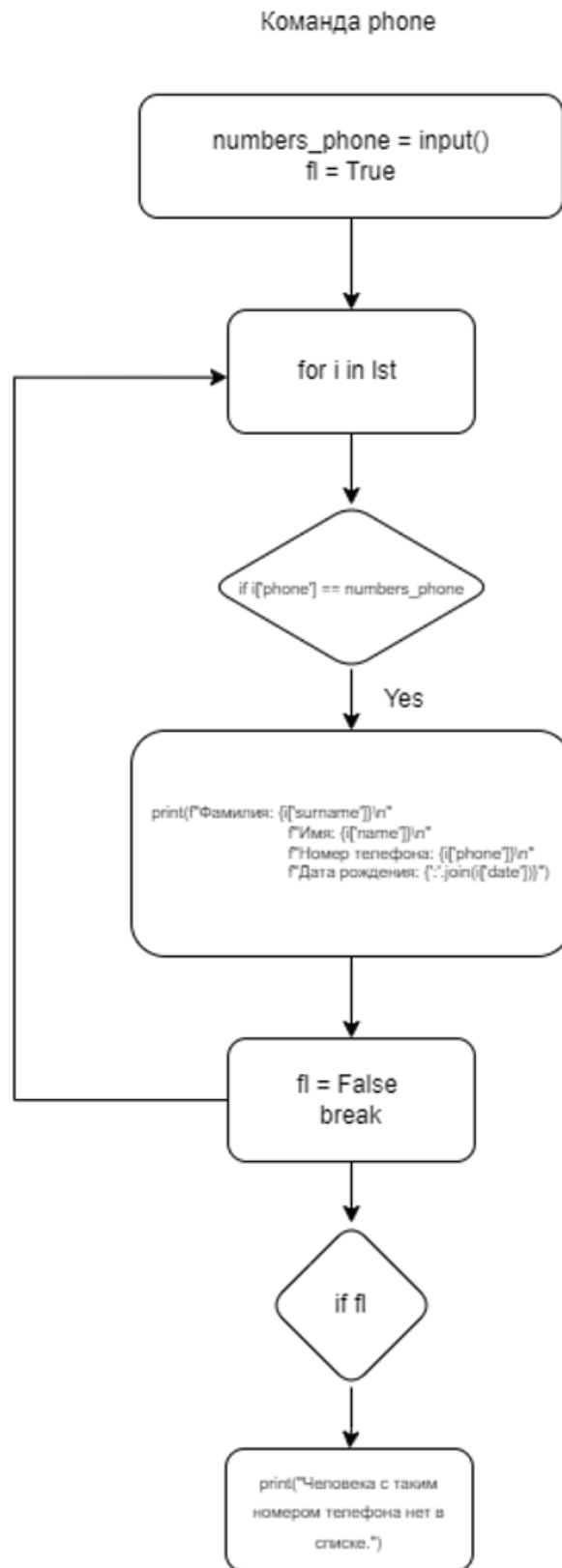


Рисунок 11 – UML-диаграмма для команды (phone)



### Логика программы

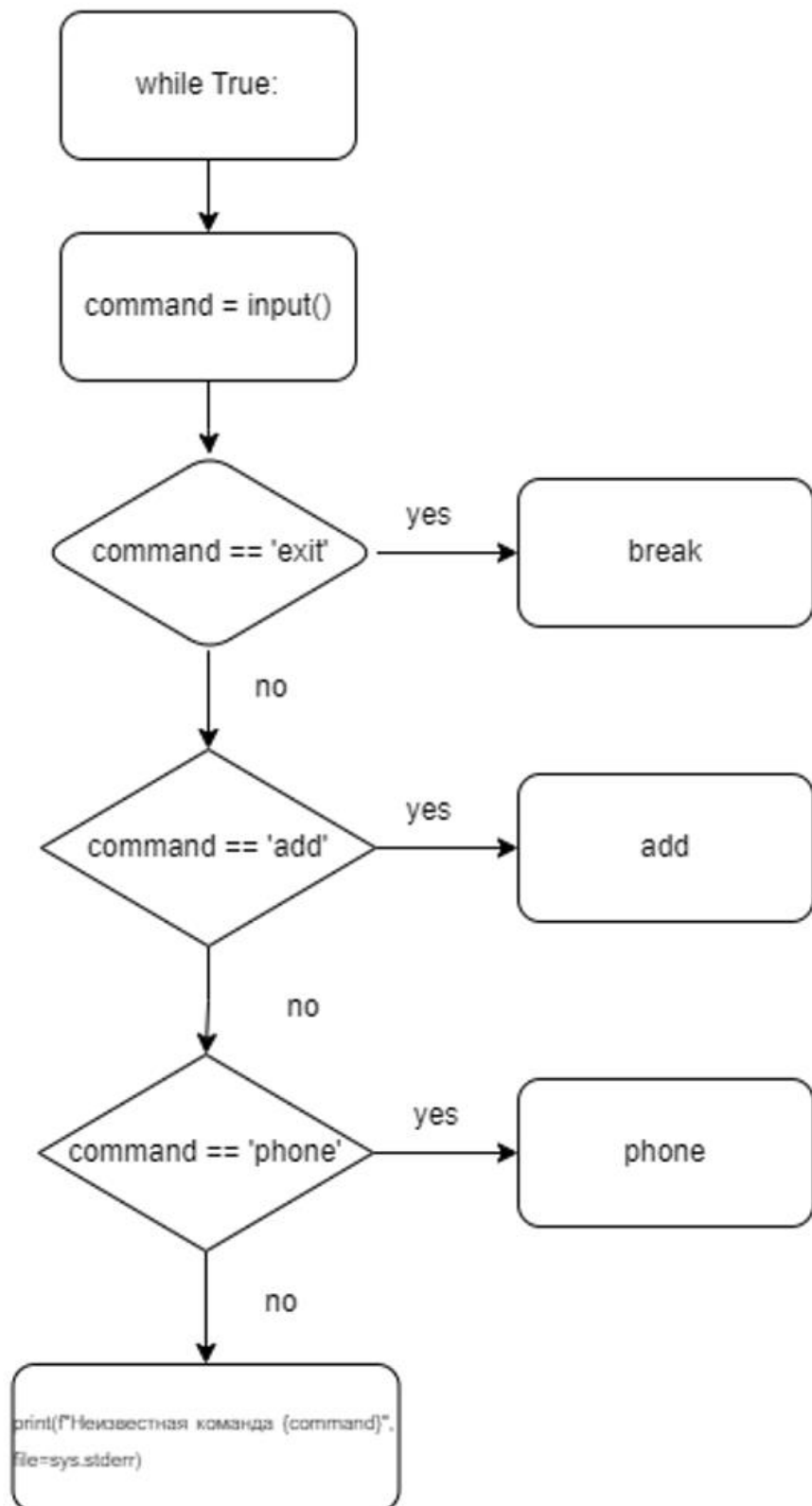
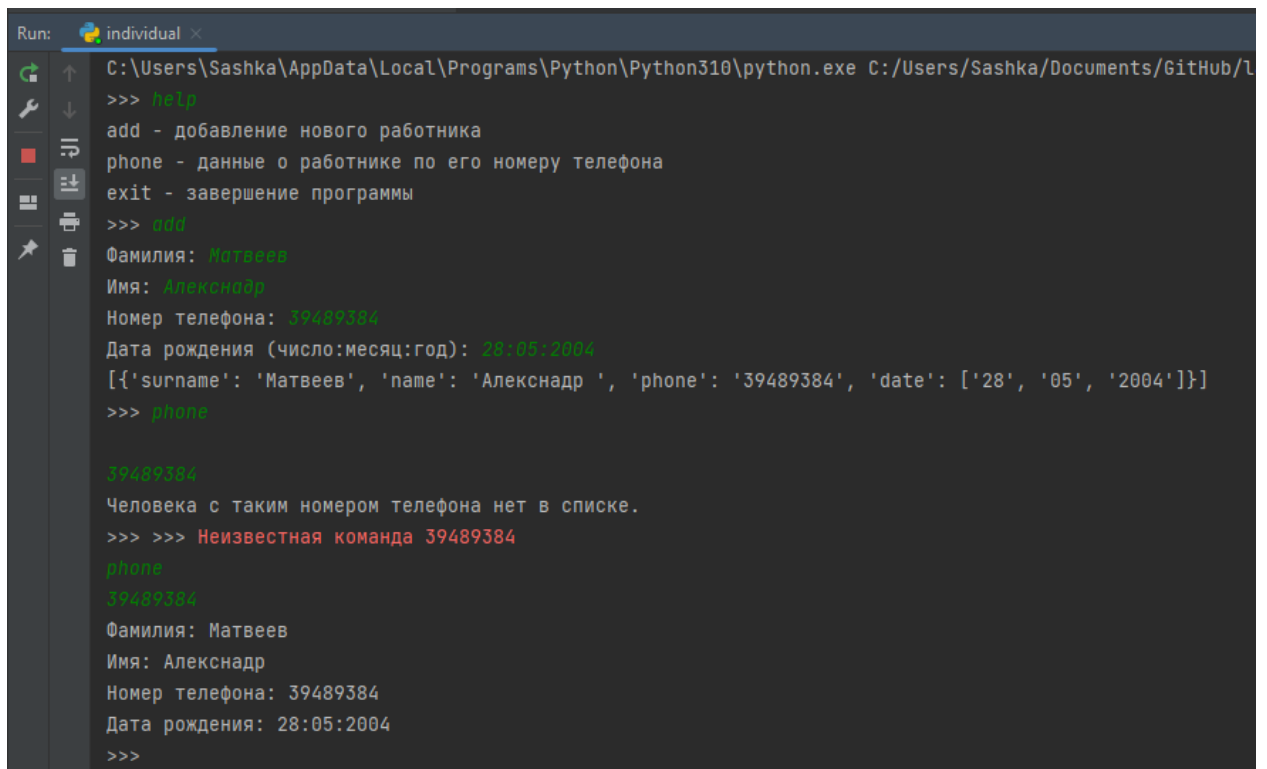


Рисунок 12 – UML-диаграмма (логика программы)



```
Run: individual x
C:\Users\Sashka\AppData\Local\Programs\Python\Python310\python.exe C:/Users/Sashka/Documents/GitHub/L
>>> help
add - добавление нового работника
phone - данные о работнике по его номеру телефона
exit - завершение программы
>>> add
Фамилия: Матвеев
Имя: Алекснадр
Номер телефона: 39489384
Дата рождения (число:месяц:год): 28:05:2004
[{'surname': 'Матвеев', 'name': 'Алекснадр ', 'phone': '39489384', 'date': ['28', '05', '2004']}]
>>> phone
39489384
Человека с таким номером телефона нет в списке.
>>> >>> Известная команда 39489384
phone
39489384
Фамилия: Матвеев
Имя: Алекснадр
Номер телефона: 39489384
Дата рождения: 28:05:2004
>>>
```

Рисунок 13 – Пример выполнения программы

### Контрольные вопросы:

1. Что такое словари в языке Python?

О: словари в Python - это структуры данных, которые хранят пары ключ-значение, причем в качестве ключа могут быть использованы только неизменяемые обьккты.

2. Может ли функция len() быть использована при работе со словарями?

О: да, она вернет количество пар (ключ: значение).

3. Какие методы обхода словарей Вам известны?

О: При помощи цикла for, используя методы словаря dict.values(), dict.items(), dict.keys(). А также при помощи словарных включений.

4. Какими способами можно получить значения из словаря по ключу?

О: при помощи метода словаря `dict.get('key')` или при непосредственно обратившись по ключу (`dict['key']`).

5. Какими способами можно установить значение в словаре по ключу?

О: при помощи метода `dict.setdefault('key', 'value')` или `dict['key'] = value`.

6. Что такое словарь включений?

О: способ создания словаря при помощи пары ключ: значения. Имеет следующий синтаксис: `{key: value for _ in range(10)}`

7. Самостоятельно изучите возможности функции `zip()` приведите примеры ее использования.

О: `zip()` – это функция, которая создает итератор, способный объединять несколько итерируемых объектов, возвращая кортежи из элементов каждого из них. Пример:

```
>>> list(zip(range(3), ['fee', 'fi', 'fo', 'fum']))
[(0, 'fee'), (1, 'fi'), (2, 'fo')]
```

8. Самостоятельно изучите возможности модуля `datetime`. Каким функционалом по работе с датой и временем обладает этот модуль?

О: Модуль `datetime` предоставляет классы для манипулирования датами и временем. Для примера он позволяет вычислять разницу во времени, датами, форматировать дату и время, а также сравнивать их и т.п. Например в своем индивидуальном задании я использовал следующую функцию этого модуля. Класс `datetime` обладает методом `strptime`, которая преобразует строку в объект `datetime`, который впоследствии было легко сравнивать с другими объектами, тем самым я смог отсортировать список словарей по датам, указанным в них.

```
lst.sort(key=lambda x: datetime.strptime('-'.join(x['date']), '%d-%m-%Y'))
```