

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ
ФЕДЕРАЦИИ**

**Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций
«Исследование возможностей Git для работы с
локальными репозиториями»**

**Отчет по лабораторной работе № 2.6
по дисциплине «Основы программной инженерии»**

Выполнил студент группы
ПИЖ-б-о-21-1

Трушева В. О. .« » 2022г.

Подпись студента _____

Работа защищена «
» _____ 20__ г.

Проверила Воронкин Р.А.

(подпись)

Ставрополь 2022

Методика и порядок выполнения работы

1. Изучить теоретический материал работы.
2. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.
3. Выполните клонирование созданного репозитория.
4. Дополните файл .gitignore необходимыми правилами для работы с IDE PyCharm.
5. Организуйте свой репозиторий в соответствие с моделью ветвления git-flow.
6. Создайте проект PyCharm в папке репозитория.
7. Проработайте пример лабораторной работы. Создайте для него отдельный модуль языка Python. Зафиксируйте изменения в репозитории.
8. Приведите в отчете скриншоты результатов выполнения примера при различных исходных данных вводимых с клавиатуры.

```
1_Task x
D:\9_ЛР\PyCharm\Scripts\python.exe D:\9_ЛР\9_ЛР\Tasks\1_Task.py
>>> help
Список команд:

add - добавить работника;
list - вывести список работников;
select <стаж> - запросить работников со стажем;
help - отобразить справку;
exit - завершить работу с программой.
>>> add
Фамилия и инициалы? Трушева В0
Должность? Фотограф
Год поступления? 2020
>>> дшые
>>> Неизвестная команда дшые
list
+-----+-----+-----+-----+
| № | Ф.И.О. | Должность | Год |
+-----+-----+-----+-----+
| 1 | Трушева В0 | Фотограф | 2020 |
+-----+-----+-----+-----+
>>> select 1
1: Трушева В0
>>> exit

Process finished with exit code 0
```

Рисунок 1 – Результат выполнения программы

9. Решите задачу: создайте словарь, связав его с переменной school , и наполните данными, которые бы отражали количество учащихся в разных классах (1а, 1б, 2б, 6а, 7в и т. п.). Внесите изменения в словарь согласно следующему:

- а) в одном из классов изменилось количество учащихся,
- б) в школе появился новый класс,
- с) в школе был расформирован (удален) другой класс. Вычислите общее количество учащихся в школе.

```
1_Task x
D:\9_ЛР\PyCharm\Scripts\python.exe D:\9_ЛР\9_ЛР\Ind_Tasks\1_Task.py
Первоначальный состав:
{'1a': 28, '1б': 32, '4г': 25, '9е': 25, '10б': 29, '8в': 21}

Состав после изменений:
{'1a': 28, '4г': 25, '9е': 25, '10б': 29, '8в': 21, '11г': 30}

Общее количество учащихся в школе = 158

Process finished with exit code 0
```

Рисунок 2 – Результат выполнения программы

10. Зафиксируйте сделанные изменения в репозитории.

11. Решите задачу: создайте словарь, где ключами являются числа, а значениями – строки. Примените к нему метод `items()`, с помощью полученного объекта `dict_items` создайте новый словарь, "обратный" исходному, т. е. ключами являются строки, а значениями – числа.

```
2_Task x
D:\fgit\9_LR\PyCharm\Scripts\python.exe "D:\fgit\!ЗАГР
Исходный словарь:
{1: 'one', 2: 'two', 3: 'three', 4: 'four'}

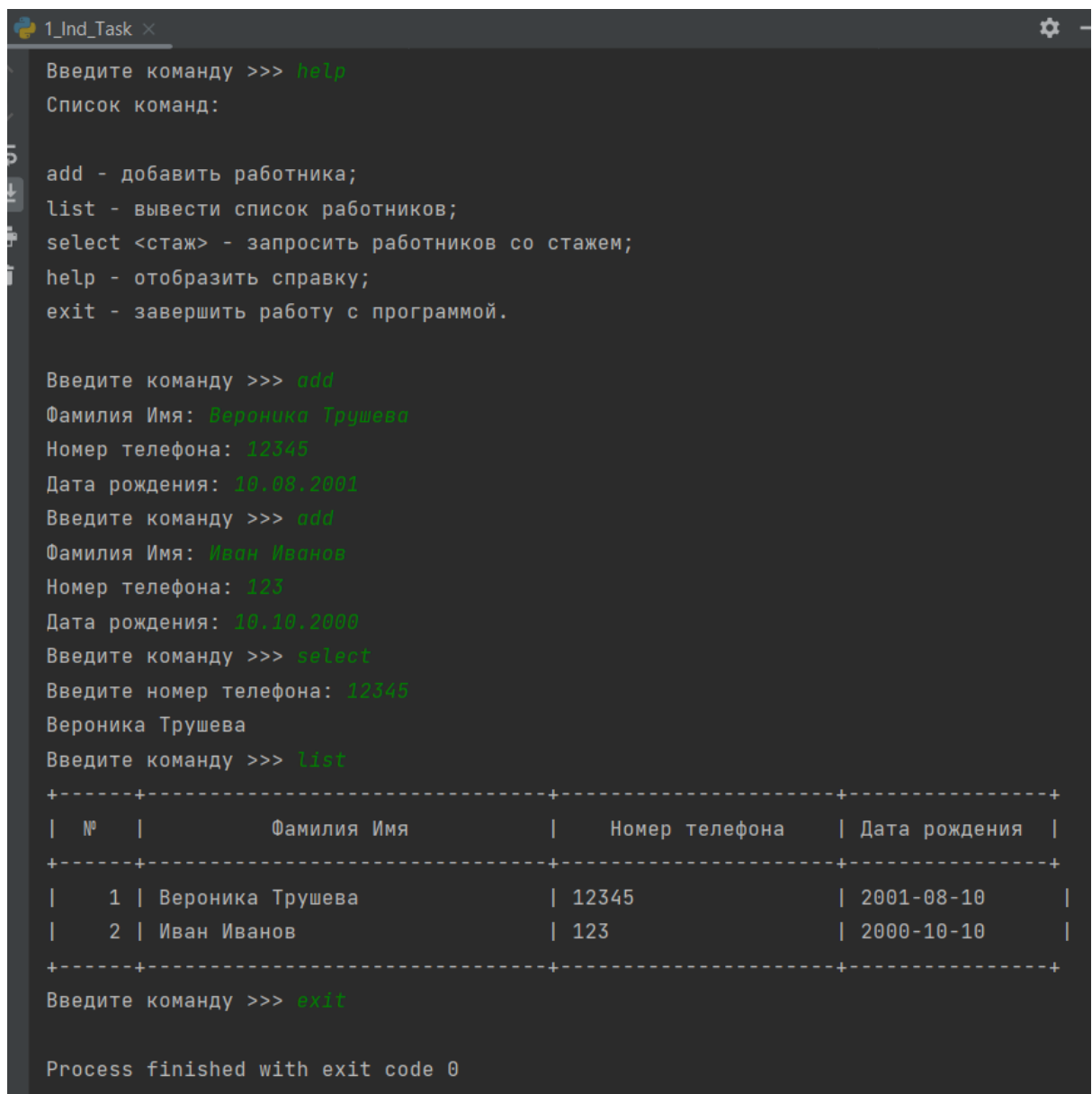
Обратный словарь:
{'one': 1, 'two': 2, 'three': 3, 'four': 4}

Process finished with exit code 0
```

Рисунок 3 – Результат выполнения программы

Индивидуальное задание. Вариант – 11

Условие. Использовать словарь, содержащий следующие ключи: фамилия, имя; номер телефона; дата рождения (список из трех чисел). Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в список, состоящий из словарей заданной структуры; записи должны быть упорядочены по датам рождения; вывод на экран информации о человеке, номер телефона которого введен с клавиатуры; если такого нет, выдать на дисплей соответствующее сообщение.



```
1_Ind_Task x
Введите команду >>> help
Список команд:

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

Введите команду >>> add
Фамилия Имя: Вероника Трушева
Номер телефона: 12345
Дата рождения: 10.08.2001
Введите команду >>> add
Фамилия Имя: Иван Иванов
Номер телефона: 123
Дата рождения: 10.10.2000
Введите команду >>> select
Введите номер телефона: 12345
Вероника Трушева
Введите команду >>> list
+-----+-----+-----+-----+
| № | Фамилия Имя | Номер телефона | Дата рождения |
+-----+-----+-----+-----+
| 1 | Вероника Трушева | 12345 | 2001-08-10 |
| 2 | Иван Иванов | 123 | 2000-10-10 |
+-----+-----+-----+-----+
Введите команду >>> exit

Process finished with exit code 0
```

Рисунок 4 – Результат выполнения программы

12. Зафиксируйте сделанные изменения в репозитории.

13. Приведите в отчете скриншоты работы программ и UML-диаграммы деятельности решения индивидуального задания.

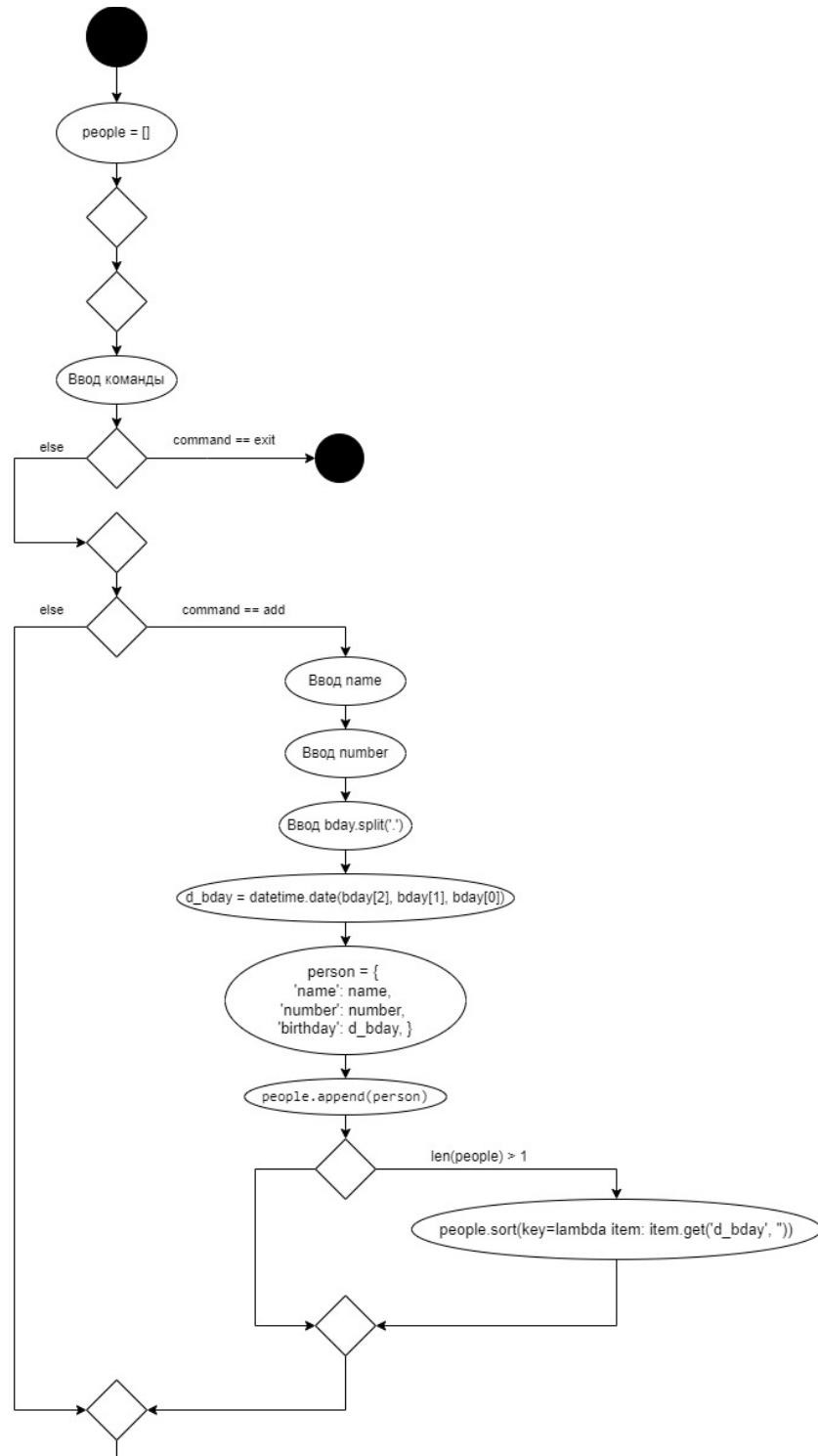


Рисунок 5 – UML-диаграмма индивидуального задания (1 часть)

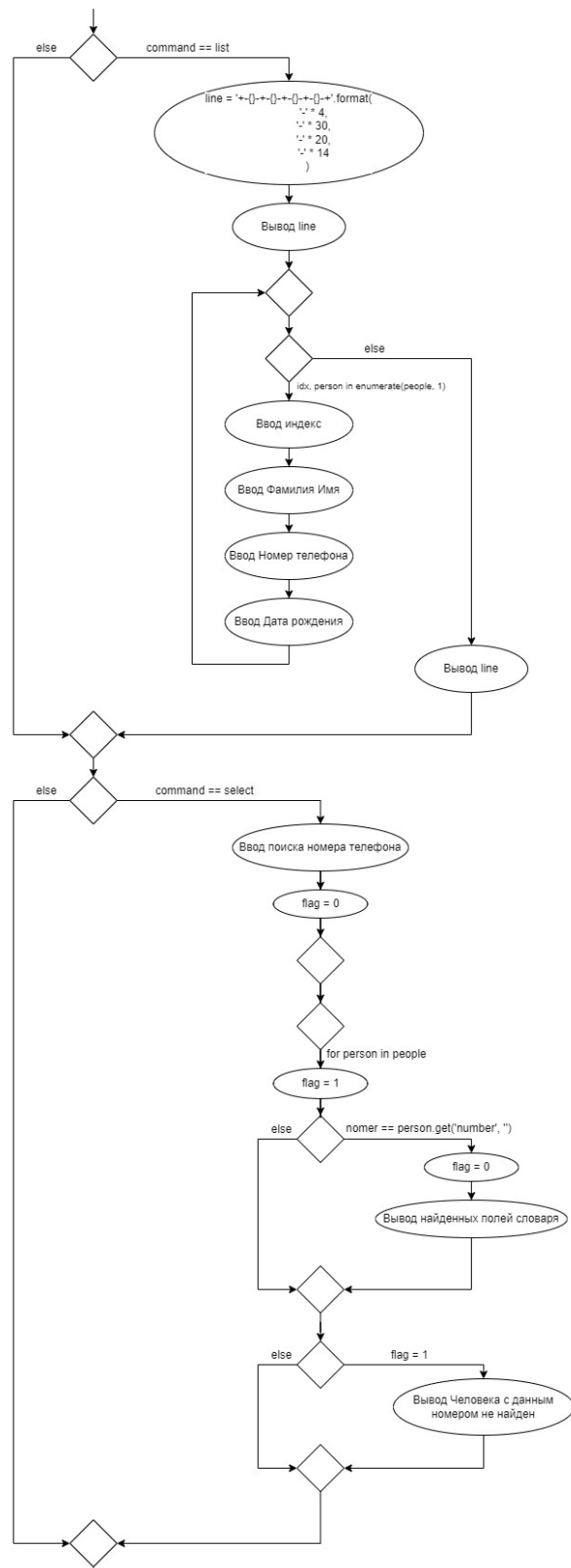


Рисунок 6 – UML-диаграмма индивидуального задания (2 часть)

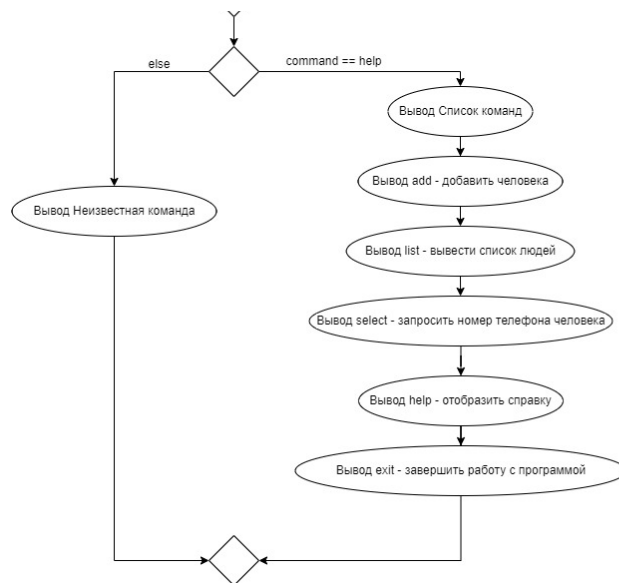


Рисунок 7 – UML-диаграмма индивидуального задания (3 часть)

14. Зафиксируйте сделанные изменения в репозитории.

15. Добавьте отчет по лабораторной работе в формате PDF в папку doc репозитория. Зафиксируйте изменения.

16. Выполните слияние ветки для разработки с веткой main/master.

17. Отправьте сделанные изменения на сервер GitHub.

18. Отправьте адрес репозитория GitHub на электронный адрес преподавателя.

Вопросы для защиты работы

1. Что такое словари в языке Python?

Словари в Python – это изменяемые отображения ссылок на объекты, доступные по ключу.

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

Функция `len()` возвращает длину (количество элементов) в объекте. Аргумент может быть последовательностью, такой как строка, байты, кортеж, список или диапазон или коллекцией (такой как словарь, множество или неизменяемое множество).

3. Какие методы обхода словарей Вам известны?

Самый очевидный вариант обхода словаря — это попытаться напрямую запустить цикл `for` по объекту словаря, так же как мы делаем это со списками, кортежами, строками и любыми другими итерируемыми объектами. `for something in currencies:`

```
print(something)
```

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

С помощью метода `.get()`

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

С помощью функции `dict.update()`

6. Что такое словарь включений?

Словарь включений аналогичен списковым включениям, за исключением того, что он создаёт объект словаря вместо списка.

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

Функция `zip()` в Python создает итератор, который объединяет элементы из нескольких источников данных. Эта функция работает со списками, кортежами, множествами и словарями для создания списков или кортежей, включающих все эти данные. Предположим, что есть список имен и номером сотрудников, и их нужно объединить в массив кортежей. Для этого можно использовать функцию `zip()`. Вот пример программы, которая делает именно это:

```
employee_numbers = [3,9, 5, 1, 8]
employee_names = ["Дима", "Саша", "Катя", "Анна"]
zipped_values = zip(employee_names, employee_numbers)
zipped_list = list(zipped_values)
print(zipped_list)
```

Функция `zip` возвращает следующее:

```
>>> employee_numbers = [3,9, 5, 1, 8]
>>> employee_names = ["Дима", "Саша", "Катя", "Анна"]
>>> zipped_values = zip(employee_names, employee_numbers)
>>> zipped_list = list(zipped_values)
>>> print(zipped_list)
[('Дима', 3), ('Саша', 9), ('Катя', 5), ('Анна', 1)]
>>>
```

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

`Datetime` — важный элемент любой программы, написанной на Python. Этот модуль позволяет управлять датами и временем, представляя их в таком виде, в котором пользователи смогут их понимать. `datetime` включает различные компоненты. Так, он состоит из объектов следующих типов:

`date` — хранит дату

`time` — хранит время

`datetime` — хранит дату и время Как получить текущие дату и время?

```
import datetime
dt_now = datetime.datetime.now()
print(dt_now)
```

```
>>> import datetime
>>> dt_now = datetime.datetime.now()
>>> print(dt_now)
2022-12-06 23:32:55.102400
```

Получить текущую дату:

```
from datetime import date
current_date = date.today()
print(current_date)
```

```
>>> from datetime import date
>>> current_date = date.today()
>>> print(current_date)
2022-12-06
>>>
```

Получить текущее время:

```
import datetime
current_date_time = datetime.datetime.now()
current_time = current_date_time.time()
print(current_time)
```

```
>>> import datetime
>>> current_date_time = datetime.datetime.now()
>>> current_time = current_date_time.time()
>>> print(current_time)
23:34:31.333160
>>>
```