

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ  
ФЕДЕРАЦИИ**  
**Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРОКАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Институт цифрового развития**

**ОТЧЁТ**

**по лабораторной работе №2.6**

Дисциплина: «Программирование на Python»

Тема: «Работа со словарями в языке Python»

Выполнил: студент 1 курса

группы ИВТ-б-о-21-1

Кочкаров Умар Ахматович

Ставрополь 2022

- 1) Создал репозиторий с использованием лицензии MIT и клонировал его на свой ПК

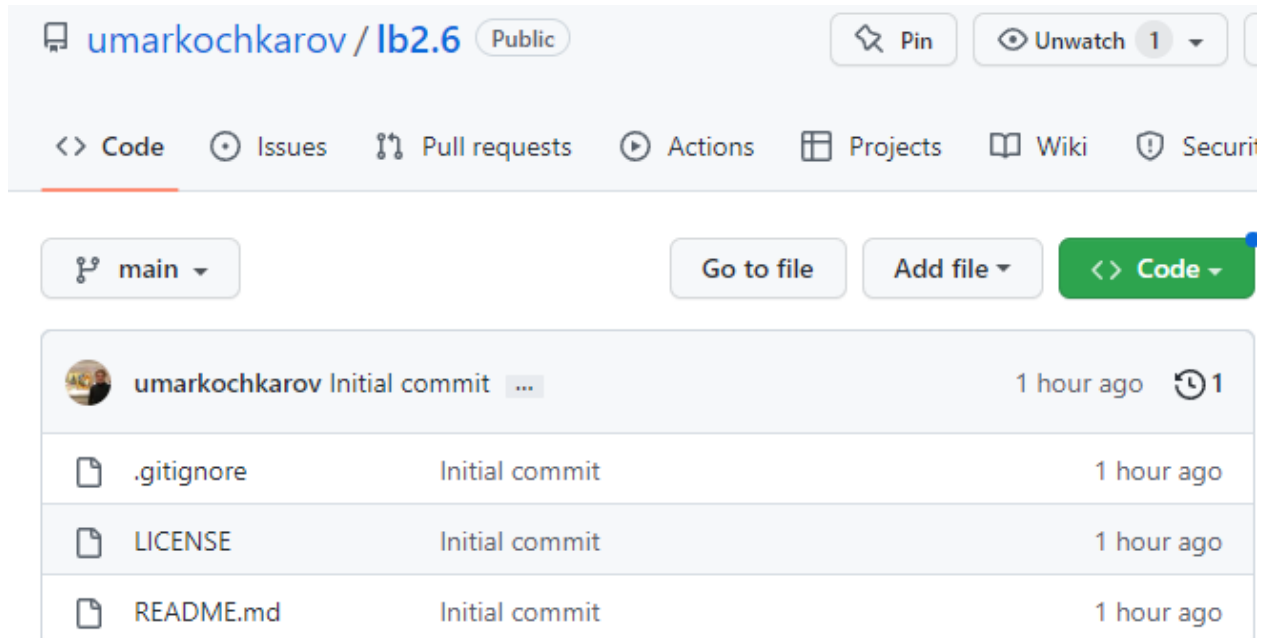
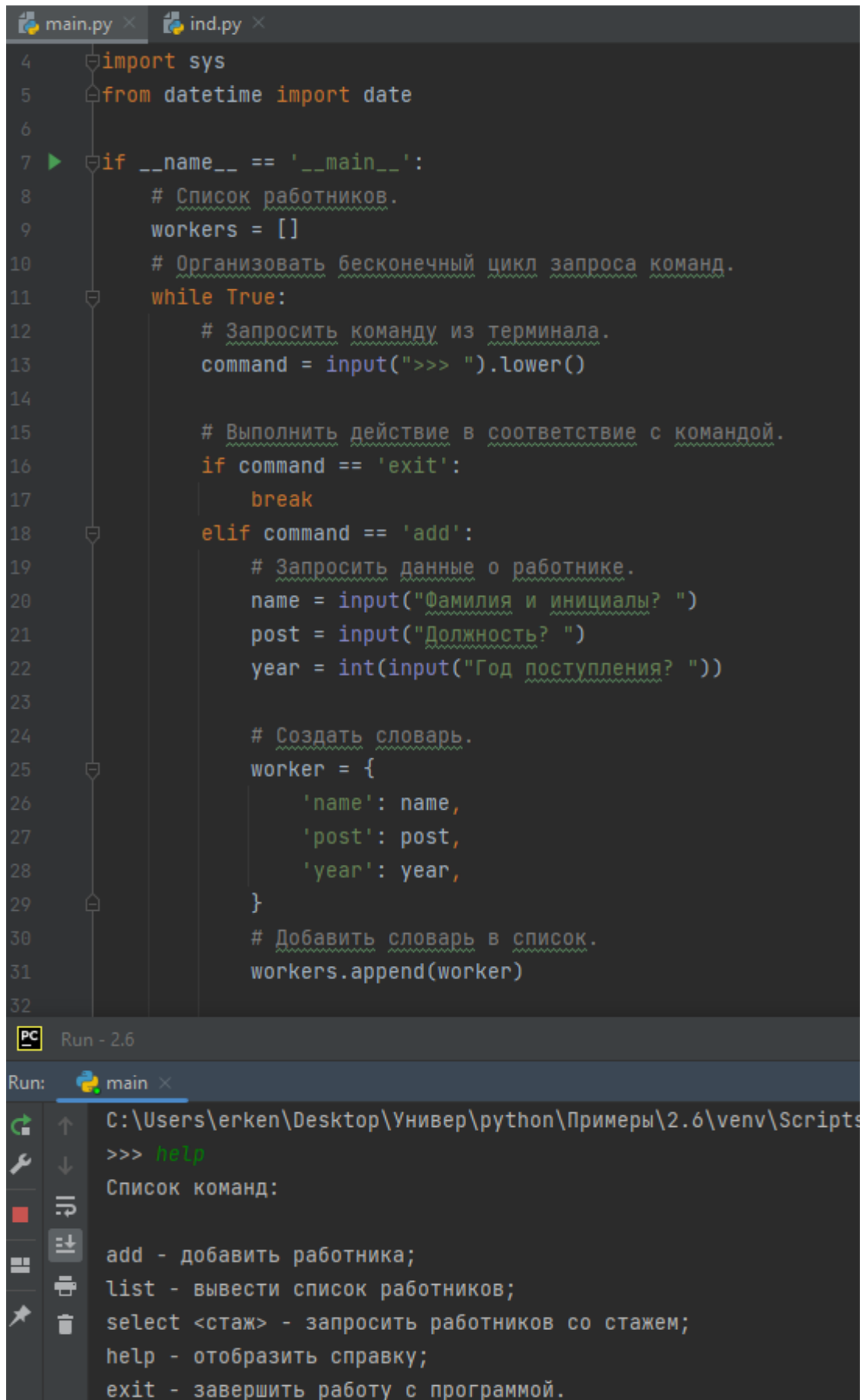


Рисунок 1. Репозиторий

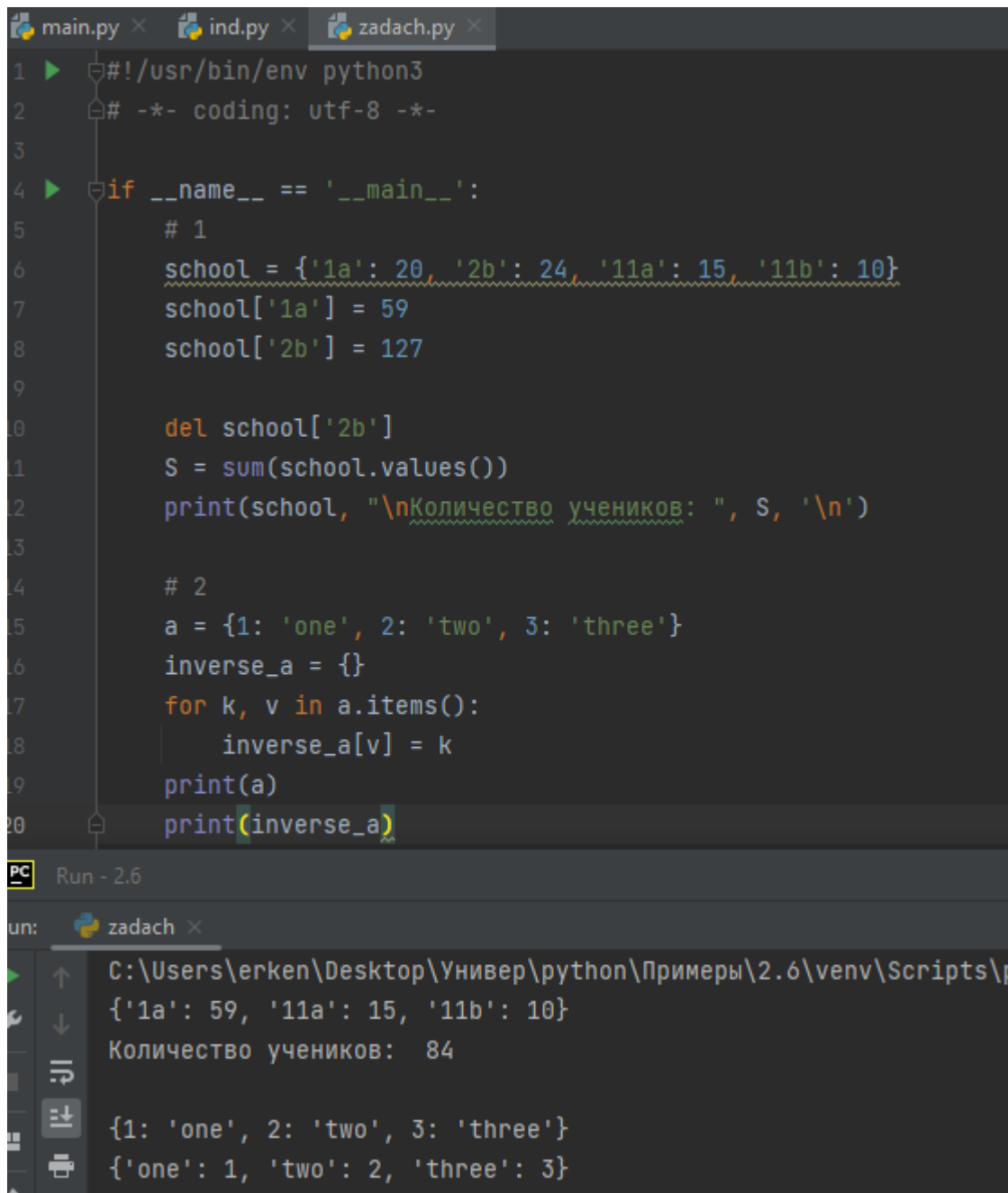
## 2) Проработал примеры:



```
main.py x ind.py x
4 import sys
5 from datetime import date
6
7 if __name__ == '__main__':
8     # Список работников.
9     workers = []
10    # Организовать бесконечный цикл запроса команд.
11    while True:
12        # Запросить команду из терминала.
13        command = input(">>> ").lower()
14
15        # Выполнить действие в соответствие с командой.
16        if command == 'exit':
17            break
18        elif command == 'add':
19            # Запросить данные о работнике.
20            name = input("Фамилия и инициалы? ")
21            post = input("Должность? ")
22            year = int(input("Год поступления? "))
23
24            # Создать словарь.
25            worker = {
26                'name': name,
27                'post': post,
28                'year': year,
29            }
30            # Добавить словарь в список.
31            workers.append(worker)
32
PC Run - 2.6
Run: main x
C:\Users\erken\Desktop\Универ\python\Примеры\2.6\venv\Scripts
>>> help
Список команд:
add - добавить работника;
list - вывести список работников;
select <стаж> - запросить работников со стажем;
help - отобразить справку;
exit - завершить работу с программой.
```

Рисунок 2. Код программы

- 3) 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  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  if __name__ == '__main__':
5      # 1
6      school = {'1a': 20, '2b': 24, '11a': 15, '11b': 10}
7      school['1a'] = 59
8      school['2b'] = 127
9
10     del school['2b']
11     S = sum(school.values())
12     print(school, "\nКоличество учеников: ", S, '\n')
13
14     # 2
15     a = {1: 'one', 2: 'two', 3: 'three'}
16     inverse_a = {}
17     for k, v in a.items():
18         inverse_a[v] = k
19     print(a)
20     print(inverse_a)
```

Run - 2.6

un: zadach x

C:\Users\erken\Desktop\Универ\python\Примеры\2.6\venv\Scripts\python.exe zadach.py

```
{'1a': 59, '11a': 15, '11b': 10}
Количество учеников: 84

{1: 'one', 2: 'two', 3: 'three'}
{'one': 1, 'two': 2, 'three': 3}
```

Рисунок 3. Решение задачи

#### 4) Индивидуальное задание

```
4      # Список .
5      airport = []
6
7      # Организовать бесконечный цикл запроса команд.
8      while True:
9          # Запросить команду из терминала.
10         command = input(">>> ").lower()
11
12         # Выполнить действие в соответствие с командой.
13         if command == 'exit':
14             break
15
16         elif command == 'add':
17             # Запросить данные .
18             race = input("Название пункта назначения рейса ")
19             number = input("Номер рейса ")
20             type = input("Тип самолёта ")
21
22             # Создать словарь.
23             airports = {
24                 'race': race,
25                 'number': number,
26                 'type': type,
27             }
28
29             # Добавить словарь в список.
30             airport.append(airports)
31             # Отсортировать список в случае необходимости.
32             if len(airport) > 1:
33                 airport.sort(key=lambda item: item.get('race', ''))
34
```

Run - 2.6

Run: ind ×

C:\Users\erken\Desktop\Универ\python\Примеры\2.6\venv\Scripts\python.exe C:/Users/erken/

```
>>> add
Название пункта назначения рейса Istanbul
Номер рейса 745
Тип самолёта Boeing
>>> list
```

No	Пункт	Номер	Тип самолёта.
1	Istanbul	745	Boeing

Рисунок 4. Индивидуальное задание

## **Контрольные вопросы:**

### **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 = [2, 9, 18, 28]

employee_names = ["Дима", "Марина", "Андрей", "Никита"]

zipped_values = zip(employee_names, employee_numbers)

zipped_list = list(zipped_values)

print(zipped_list)
```

Функция `zip` возвращает следующее:

```
[('Дима', 2), ('Марина', 9), ('Андрей', 18), ('Никита', 28)]
```

## **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)
```

Результат:

2022-09-11 15:43:32.249588

Получить текущую дату:

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

Результат:

2022-09-11

Получить текущее время:

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

Результат:

15:51:05.627643

**Вывод:** Изучил Словари в python