

**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 №9 по
дисциплине: основы программной инженерии**

Выполнил:

студент группы ПИЖ-б-о-20-1

Турклиев Владимир Назирович

Проверил:

доцент кафедры инфокоммуникаций

Романкин Р.А.

Ставрополь, 2021 г.

Ход работы

Пример 1

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

import sys
from datetime import date

if __name__ == '__main__':
    # Список работников.
    workers = []

    # 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.
            name = input("Фамилия и инициалы? ")
            post = input("Должность? ")
            year = int(input("Год поступления? "))

            # Создать словарь.
            worker = {
                'name': name,
                'post': post,
                'year': year,
            }

            # Добавить словарь в список.
            workers.append(worker)
            # Отсортировать список в случае необходимости.
            if len(workers) > 1:
                workers.sort(key=lambda item: item.get('name', ''))

        elif command == 'list':
            # Заголовок таблицы.
```

```

line = '+-{}-+-{}-+-{}-+-{}-+'.format(
    '-' * 4,
    '-' * 30,
    '-' * 20,
    '-' * 8
)
print(line)
print(
    '| {:^4} | {:^30} | {:^20} | {:^8} |'.format(
        "№",
        "Ф.И.О.",
        "Должность",
        "Год"
    )
)
print(line)

# Вывести данные о всех сотрудниках.
for idx, worker in enumerate(workers, 1):
    print(
        '| {:>4} | {:<30} | {:<20} | {:>8} |'.format(
            idx,
            worker.get('name', ''),
            worker.get('post', ''),
            worker.get('year', 0)
        )
    )
    print(line)

elif command.startswith('select '):
    # Получить текущую дату.
    today = date.today()

    # Разбить команду на части для выделения номера года.
    parts = command.split(' ', maxsplit=1)
    # Получить требуемый стаж.
    period = int(parts[1])

    # Инициализировать счетчик.

```

```

count = 0
# Проверить сведения работников из списка.
for worker in workers:
    if today.year - worker.get('year', today.year) >= period:
        count += 1
        print(
            '{:>4}: {}'.format(count, worker.get('name', ''))
        )

# Если счетчик равен 0, то работники не найдены.
if count == 0:
    print("Работники с заданным стажем не найдены.")

elif command == 'help':
    # Вывести справку о работе с программой.
    print("Список команд:\n")
    print("add - добавить работника;")
    print("list - вывести список работников;")
    print("select <стаж> - запросить работников со стажем;")
    print("help - отобразить справку;")
    print("exit - завершить работу с программой.")
else:
    print(f"Неизвестная команда {command}", file=sys.stderr)

```

Фамилия и инициалы? *Турклиев В.Н*

Должность? *Дворник*

Год поступления? *2022*

>>> *add*

Фамилия и инициалы? *Путин В.В*

Должность? *Слуга народа*

Год поступления? *1999*

>>> *add*

Фамилия и инициалы? *Тарасов С.В*

Должность? *зам. дворника*

Год поступления? *2020*

>>> *list*

№	Ф.И.О.	Должность	Год
1	Путин В.В	Слуга народа	1999
2	Тарасов С.В	зам. дворника	2020
3	Турклиев В.Н	Дворник	2022

```
>>> select 24
Работники с заданным стажем не найдены.
>>> select 1
1: Путин В.В
2: Тарасов С.В
>>> select 0
1: Путин В.В
2: Тарасов С.В
3: Турклиев В.Н
```

Задание 9

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

if __name__ == '__main__':
    school = {'1a': '19', '3б': '15', '4в': '23', '5а': '18'}
    sum = 0

    for i in school:
        sum += int(school[i])

    print(f"Общее количество учащихся: {sum}")
```

9 x

C:\Users\Vova\AppData\Local\Programs\Python\Python39\python.exe "

Общее количество учащихся: 75

Задание 11

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

if __name__ == '__main__':
    dict1 = {
        1: 'one',
        2: 'two',
        3: 'three',
        4: 'four'
    }
    dict2 = {}
    for x, y in dict1.items():
        dict2.setdefault(y, x)
    print(f"Dict1: {dict1}")
    print(f"Dict2: {dict2}")
```

11 x

C:\Users\Vova\AppData\Local\Programs\Python\Python39\python.exe

Dict1: {1: 'one', 2: 'two', 3: 'three', 4: 'four'}

Dict2: {'one': 1, 'two': 2, 'three': 3, 'four': 4}

Process finished with exit code 0

Индивидуальное задание

. Использовать словарь, содержащий следующие ключи: название пункта назначения рейса; номер рейса; тип самолета. Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в список, состоящий из словарей заданной структуры; записи должны быть упорядочены по возрастанию номера рейса; вывод на экран номеров рейсов и типов самолетов, вылетающих в пункт назначения, название которого совпало с названием, введенным с клавиатуры; если таких рейсов нет, выдать на дисплей соответствующее сообщение.

```

1  ▶  #!/usr/bin/env python3
2      # -*- coding: utf-8 -*-
3
4
5      shedule = []
6      while True:
7          shedule.append({
8              'название пункта назначения рейса': input('Пункт назначения? - '),
9              'номер рейса': int(input('Номер рейса? - ')),
10             'тип самолета': input('Тип самолета? - ')
11         })
12         if input('Напишите "д" чтобы продолжить ввод данных, "н" для завершения ввода.\n') == 'д':
13             pass
14         else:
15             break
16     shedule = sorted(shedule, key=lambda row: row['номер рейса'])
17
18     for i in shedule:
19         print(i)
20
21     destination = input('Рейс в какой город вас интересует?')
22     approved = []
23     for i in shedule:
24         if i['название пункта назначения рейса'] == destination:
25             approved.append(i)
26
27     for i in approved:
28         print(i)
29
30
Пункт назначения? - Рим
Номер рейса? - 123
Тип самолета? - Грузовой
Напишите "д" чтобы продолжить ввод данных, "н" для завершения ввода.
н
{'название пункта назначения рейса': 'Рим ', 'номер рейса': 123, 'тип самолета': 'Грузовой'}
{'название пункта назначения рейса': 'Берлин', 'номер рейса': 333, 'тип самолета': 'Грузовой'}
{'название пункта назначения рейса': 'Токио', 'номер рейса': 555, 'тип самолета': 'Пассажирский '}
Рейс в какой город вас интересует?

```

Ссылка - <https://github.com/vegas007gof/lab9>.

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

- 1) Что такое словари в языке Python?
Словарь (dict) представляет собой структуру данных (которая ещё называется ассоциативный массив), предназначенную для хранения произвольных объектов с доступом по ключу. Данные в словаре хранятся в формате ключ – значение.
- 2) Может ли функция len() быть использована при работе со словарями?
Да
- 3) Какие методы обхода словарей Вам известны?
for i in nums: print(nums[i]) for key, value in nums.items(): print(key, 'is', value)
Методы словаря keys() и values() позволяют получить отдельно перечни ключей и значений.
- 4) Какими способами можно получить значения из словаря по ключу?
dict["key"] dict.get("key")
- 5) Какими способами можно установить значение в словаре по ключу?

```
dict.setdefault("key", "value") dict["key"] = "value"
```

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. У функции `zip()` множество сценариев применения.

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

`datetime.fromordinal(ordinal)` - дата из числа, представляющего собой количество дней, прошедших с 01.01.1970.

`datetime.now(tz=None)` - объект `datetime` из текущей даты и времени.

`datetime.combine(date, time)` - объект `datetime` из комбинации объектов `date` и `time`.

`datetime.strptime(date_string, format)` - преобразует строку в `datetime` (так же, как и функция `strptime` из модуля `time`).

`datetime.strftime(format)` - см. функцию `strftime` из модуля `time`.

`datetime.date()` - объект даты (с отсечением времени).

`datetime.time()` - объект времени (с отсечением даты).

`datetime.toordinal()` - количество дней, прошедших с 01.01.1970.

`datetime.timestamp()` - возвращает время в секундах с начала эпохи.

`datetime.weekday()` - день недели в виде числа, понедельник - 0, воскресенье - 6.

`datetime.isocalendar()` - кортеж (год в формате ISO, ISO номер недели, ISO день недели).

`datetime.isoformat(sep="T")` - красивая строка вида "YYYYMMDDTHH:MM:SS.mmmmmm" или, если `microsecond == 0`, "YYYYMMDDTHH:MM:SS"