

Северо-Кавказский федеральный университет
Институт математики и информационных технологий

ОТЧЕТ
о выполнении лабораторной работы №9
по дисциплине
«Основы Программной Инженерии»

Выполнил:

Ботвинкин Никита Сергеевич

студент 2 курса, ПИЖ-б-о-21-1 группы
бакалавриата «Программная инженерия»
очной формы обучения

Ставрополь, 2022

СКРИНШОТЫ РАБОТЫ ПРОГРАММ

```
>>> add
Фамилия и инициалы? Никто
Должность? Никем
Год поступления? 2001
>>> list
```

№	Ф.И.О.	Должность	Год
1	Никто	Никем	2001

Рисунок 9.1 – Пример работы из методички

```
>>> add
Фамилия и инициалы? Ботвинкин Н.С.
Номер группы? пжж
Какие оценки? 3
Какие оценки? 4
Какие оценки? 5
Какие оценки? 5
Какие оценки? 5
```

Рисунок 9.2 – Пример ввода

```
>>> list
```

№	Ф.И.О.	Группа	Успеваемость
1	Ботвинкин Н.С.	пжж	[3, 4, 5, 5, 5]

Рисунок 9.3 – Пример вывода

```
>>> list
```

№	Ф.И.О.	Группа	Успеваемость
1	Ботвинкин Н.С.	пжж	[3, 4, 5, 5, 5]
2	Никто Н.Н.	идк	[2, 3, 3, 3, 3]

```
>>> marks
```

Ф.И.О.	Группа
Никто Н.Н.	идк

Рисунок 9.4 – Пример вывода учеников с одной «2»

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?

- Циклы

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

- Через квадратные скобки или через метод get()

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

- Через квадратные скобки

6. Что такое словарь включений?

- Удобная конструкция работы со словарями, аналогичная списковым включениям

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.

```
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)
```

Рисунок 9.5 – Пример работы zip()

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

Рисунок 9.6 - Вывод

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

- Модуль `datetime` предоставляет классы для обработки времени и даты разными способами. Поддерживается и стандартный способ представления времени, однако больший упор сделан на простоту манипулирования датой, временем и их частями.