

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»

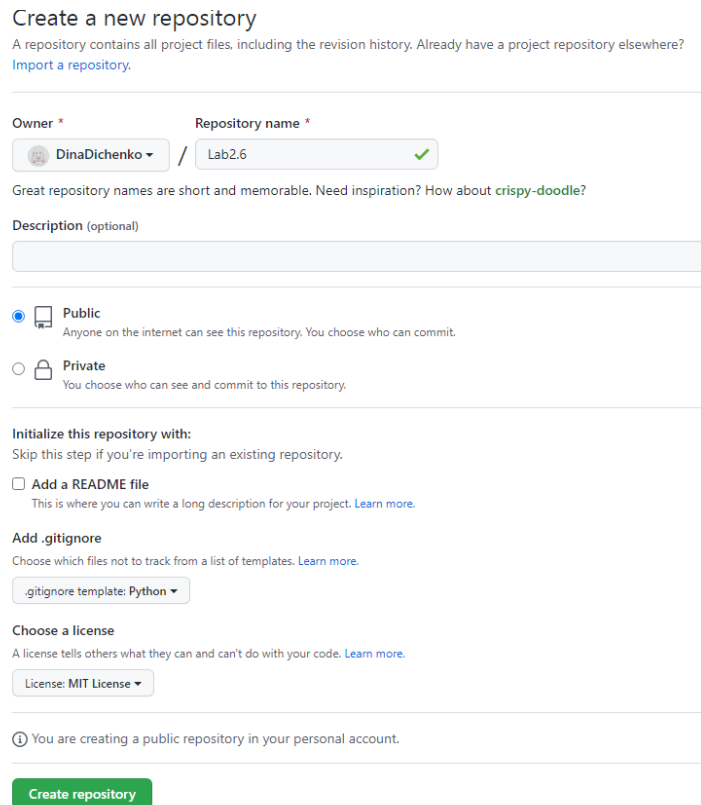
Выполнила: студентка 1 курса,
группы ИВТ-б-о-21-1
Диченко Дина Алексеевна

Ставрополь 2022

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

Практическая часть:

1. Создала общедоступный репозиторий.



Create a new repository

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

Owner * DinaDichenko / Repository name * Lab2.6 ✓

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

Description (optional)

☒ 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:
Skip this step if you're importing an existing repository.

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

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: Python

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

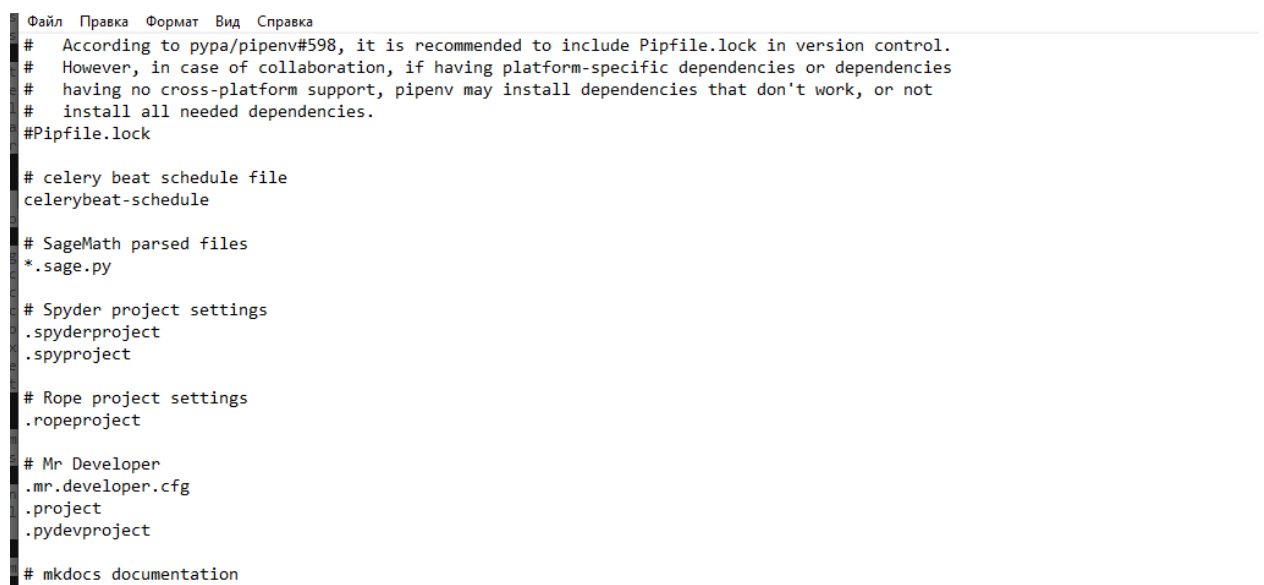
License: MIT License

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

[Create repository](#)

Рисунок 1. Создание репозитория

2. Клонировала репозиторий, дополнила файл .gitignore.



```
Файл  Правка  Формат  Вид  Справка
# According to pyup/pipenv#598, it is recommended to include Pipfile.lock in version control.
# However, in case of collaboration, if having platform-specific dependencies or dependencies
# having no cross-platform support, pipenv may install dependencies that don't work, or not
# install all needed dependencies.
#Pipfile.lock

# celery beat schedule file
celerybeat-schedule

# SageMath parsed files
*.sage.py

# Spyder project settings
.spyderproject
.spyproject

# Rope project settings
.ropeproject

# Mr Developer
.mr.developer.cfg
.project
.pydevproject

# mkdocs documentation
..
```

Рисунок 2. Редактирование файла .gitignore

3. Организовала репозиторий в соответствии с моделью аетвления

git flow.

```
C:\Users\aleks\Desktop\DiDi\Lab2.6>git flow init

Which branch should be used for bringing forth production releases?
- develop
- main
branch name for production releases: [main] main

Which branch should be used for integration of the "next release"?
- develop
branch name for "next release" development: [develop] develop

How to name your supporting branch prefixes?
feature branches? [feature/] fea
bugfix branches? [bugfix/] bug
release branches? [release/] rel
hotfix branches? [hotfix/] br
support branches? [support/] sup
version tag prefix? [] tag
hooks and filters directory? [C:/Users/aleks/Desktop/DiDi/Lab2.6/.git/hooks] hook

C:\Users\aleks\Desktop\DiDi\Lab2.6>
```

Рисунок 3. Организация репозитория в соответствии с git flow

4. Создала проект PyCharm и проработала пример лабораторной работы.

```
>>> add
Фамилия и инициалы? Лютик С.В.
Должность? стажер
Год поступления? 2015
>>> add
Фамилия и инициалы? Коровьев Л.А.
Должность? бухгалтер
Год поступления? 2010
>>> add
Фамилия и инициалы? Ломоносов П.У.
Должность? менеджер
Год поступления? 2020
>>> list
+-----+-----+-----+-----+
| No |          Ф.И.О.          |      Должность      |      Год      |
+-----+-----+-----+-----+
|  1 | Коровьев Л.А.          | бухгалтер          |      2010     |
|  2 | Ломоносов П.У.         | менеджер          |      2020     |
|  3 | Лютик С.В.             | стажер            |      2015     |
+-----+-----+-----+-----+
>>> select 5
1: Коровьев Л.А.
2: Лютик С.В.
```

Рисунок 4. Результат работы программы

5. Задача: 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.

```
>>> add
Класс? 1a
Количество учеников? 21
>>> add
Класс? 2a
Количество учеников? 22
>>> add
Класс? 3a
Количество учеников? 11
>>> del
Какой класс расформирован? 3a
>>> edit
В каком классе изменилось количество учеников? 1a
Сколько учеников стало? 30
>>> sum
Количество учеников в школе: 52
>>> list
```

No	Класс	Количество учеников
1	1a	30
2	2a	22

Рисунок 5. Результат работы программы

6. Задача: 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.

```
C:\Python37\python.exe C:/Users/student-09-525/Downloads/z2.py
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 4. Результат работы программы

7. Индивидуальное задание (вариант 8): Использовать словарь, содержащий следующие ключи: название пункта назначения; номер поезда; время отправления. Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в список, состоящий из словарей заданной структуры; записи должны быть упорядочены по номерам поездов; вывод на экран информации о поезде, номер которого введен с клавиатуры; если таких поездов нет, выдать на дисплей соответствующее сообщение.

```
>>> add
Название пункта назначения? Sad
Номер поезда? 1
Время отправления (ММ/ДД/ГГ ЧЧ:ММ)? 11/11/11 21:1
>>> Lay
Неизвестная команда lay>>>
add
Название пункта назначения? Lay
Номер поезда? 2
Время отправления (ММ/ДД/ГГ ЧЧ:ММ)? 01/10/21 12:30
>>> select
Введите номер поезда: 1
{'name': 'Sad', 'no': '1', 't': datetime.datetime(2011, 11, 11, 21, 1)}
>>> select
Введите номер поезда: 3
Поезда с таким номером нет
>>> lest
Неизвестная команда lest>>>
list
+-----+-----+-----+
|      No      |      Название      |      Время отправления      |
+-----+-----+-----+
|              | 1 | Sad              | 2011-11-11 21:01:00 |
|              | 2 | Lay              | 2021-01-10 12:30:00 |
+-----+-----+-----+
>>> _
```

Рисунок 5. Результат работы программы

Ответы на 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.

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

Нет.

3. Какие методы обхода словарей Вам известны?

Цикл for, методы items(), keys() и values().

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

В словаре доступ к значениям осуществляется по ключам, которые заключаются в квадратные скобки или с помощью метода get().

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

С помощью setdefault() можно добавить элемент в словарь.

6. Что такое словарь включений?

Словарь включений аналогичен списковым включениям, за исключением того, что он создаёт объект словаря вместо списка (краткая синтаксическая конструкция, предназначенная для создания словаря).

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

Функция zip() принимает итерируемый объект, например, список, кортеж, множество или словарь в качестве аргумента. Затем она генерирует список кортежей, которые содержат элементы из каждого объекта, переданного в функцию.

Пример:

```
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)
```

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.

datetime включает различные компоненты. Так, он состоит из объектов следующих типов:

- `date` — хранит дату
- `time` — хранит время
- `datetime` — хранит дату и время

Вывод: в результате выполнения работы были приобретены навыки по работе со словарями при написании программ с помощью языка программирования Python версии 3.x.