

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.17
дисциплины
«Программирование на языке Python»

Выполнил:
Быковская Стефания Станиславовна
2 курс, группа ИТС-б-о-22-1,
11.03.02 «Инфокоммуникационные
технологии и системы связи», очная
форма обучения

(подпись)

Проверил:

Воронкин Р. А., доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

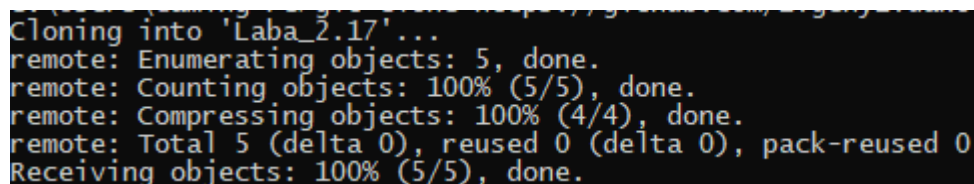
Ставрополь, 2023 г.

Тема: Разработка приложений с интерфейсом командной строки (CLI) в Python3

Цель: приобретение построения приложений с интерфейсом командной строки с помощью языка программирования Python версии 3.x.

Ход работы:

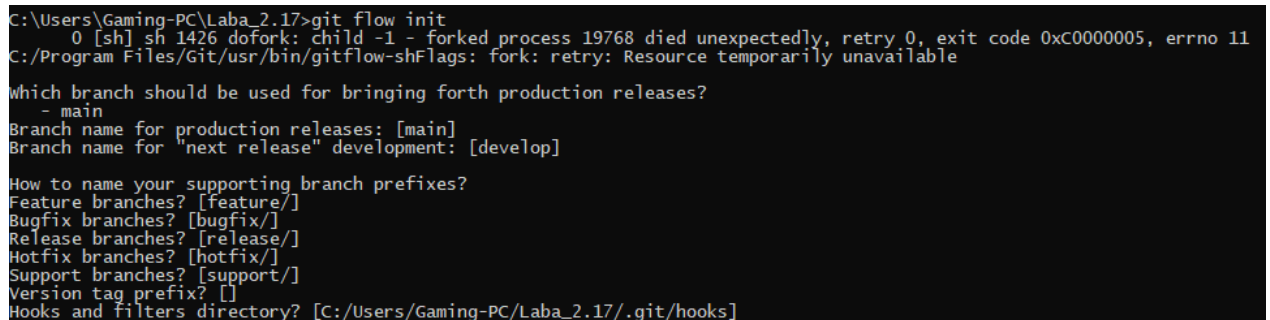
Задание 1. Создала общедоступный репозиторий на GitHub, в котором использована лицензий MIT и язык программирования Python, также добавила файл .gitignore с необходимыми правилами. Клонировала свой репозиторий на свой компьютер.



```
Cloning into 'Laba_2.17'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.
```

Рисунок 1. Клонирование репозитория

Задание 2. Организовала свой репозиторий в соответствии с моделью ветвления git-flow, появилась новая ветка develop в которой буду выполнять дальнейшие задачи.



```
C:\Users\Gaming-PC\Laba_2.17>git flow init
0 [sh] sh 1426 dofork: child -1 - forked process 19768 died unexpectedly, retry 0, exit code 0xC0000005, errno 11
C:/Program Files/Git/usr/bin/gitflow-shFlags: fork: retry: Resource temporarily unavailable

which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/Gaming-PC/Laba_2.17/.git/hooks]
```

Рисунок 2. Модель ветвления git-flow

Задание 3. Создала виртуальное окружение conda и активировала его, также установила необходимые пакеты isort, black, flake8.

```
(base) PS C:\Users\Gaming-PC> cd C:\Users\Gaming-PC\Laba_2.17
(base) PS C:\Users\Gaming-PC\Laba_2.17> conda create -n 2.17 python=3.7
Retrieving notices: ...working... done
Collecting package metadata (current_repodata.json): done
Solving environment: failed with repodata from current_repodata.json, will retry with next repodata source.
Collecting package metadata (repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 23.1.0
  latest version: 23.9.0

Please update conda by running

  $ conda update -n base -c defaults conda

Or to minimize the number of packages updated during conda update use

  conda install conda=23.9.0

## Package Plan ##

environment location: C:\Users\Gaming-PC\.conda\envs\2.17

added / updated specs:
- python=3.7

The following packages will be downloaded:
```

package	build	size
openssl-1.1.1w	h2bbff1b_0	5.5 MB
Total:		5.5 MB

Рисунок 3. Создание виртуального окружения

```
(base) PS C:\Users\Gaming-PC\Laba_2.17> conda install -c conda-forge black
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 23.1.0
  latest version: 23.9.0

Please update conda by running

  $ conda update -n base -c defaults conda

Or to minimize the number of packages updated during conda update use

  conda install conda=23.9.0
```

Рисунок 4. Добавление пакета black

```
(base) PS C:\Users\Gaming-PC\Laba_2.17> conda install -c conda-forge flake8
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Collecting package metadata (repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.

PackagesNotFoundError: The following packages are not available from current channels:

- flake8

Current channels:

- https://conda.anaconda.org/conda-forge/win-64
- https://conda.anaconda.org/conda-forge/noarch
- https://repo.anaconda.com/pkg/main/win-64
- https://repo.anaconda.com/pkg/main/noarch
- https://repo.anaconda.com/pkg/r/win-64
- https://repo.anaconda.com/pkg/r/noarch
- https://repo.anaconda.com/pkg/msys2/win-64
- https://repo.anaconda.com/pkg/msys2/noarch

To search for alternate channels that may provide the conda package you're
looking for, navigate to

  https://anaconda.org

and use the search bar at the top of the page.
```

Рисунок 5. Добавление пакета flake8

```
(base) PS C:\Users\Gaming-PC\Laba_2.17> conda install -c conda-forge isort
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 23.1.0
  latest version: 23.9.0

Please update conda by running

    $ conda update -n base -c defaults conda

Or to minimize the number of packages updated during conda update use

    conda install conda=23.9.0
```

Рисунок 6. Добавление пакета isort

Задание 4. Создала проект PyCharm в папке репозитория. Приступила к работе с примером. Добавила новый файл primer1.py.

Условие примера:

```

68 # Получить текущую дату.
69 today = date.today()
70
71 # Сформировать список работников.
72 result = []
73 for employee in staff:
74     if today.year - employee.get('year', today.year) >= period:
75         result.append(employee)
76
77 # Возвратить список выбранных работников.
78 return result
79
80
81 def save_workers(file_name, staff):
82     """
83     Сохранить всех работников в файл JSON.
84     """
85     # Открыть файл с заданным именем для записи.
86     with open(file_name, "w", encoding="utf-8") as fout:
87         # Выполнить сериализацию данных в формат JSON.
88         # Для поддержки кириллицы установим ensure_ascii=False
89         json.dump(staff, fout, ensure_ascii=False, indent=4)
90
91
92 def load_workers(file_name):
93     """
94     Загрузить всех работников из файла JSON.
95     """
96     # Открыть файл с заданным именем для чтения.
97     with open(file_name, "r", encoding="utf-8") as fin:
98         return json.load(fin)

```

Рисунок 7. Пример 1

Индивидуальное задание

Вариант 2

Создала новый файл под названием idz.py.

Условие задания: Для своего варианта лабораторной работы 2.16 необходимо дополнительно реализовать интерфейс командной строки (CLI).

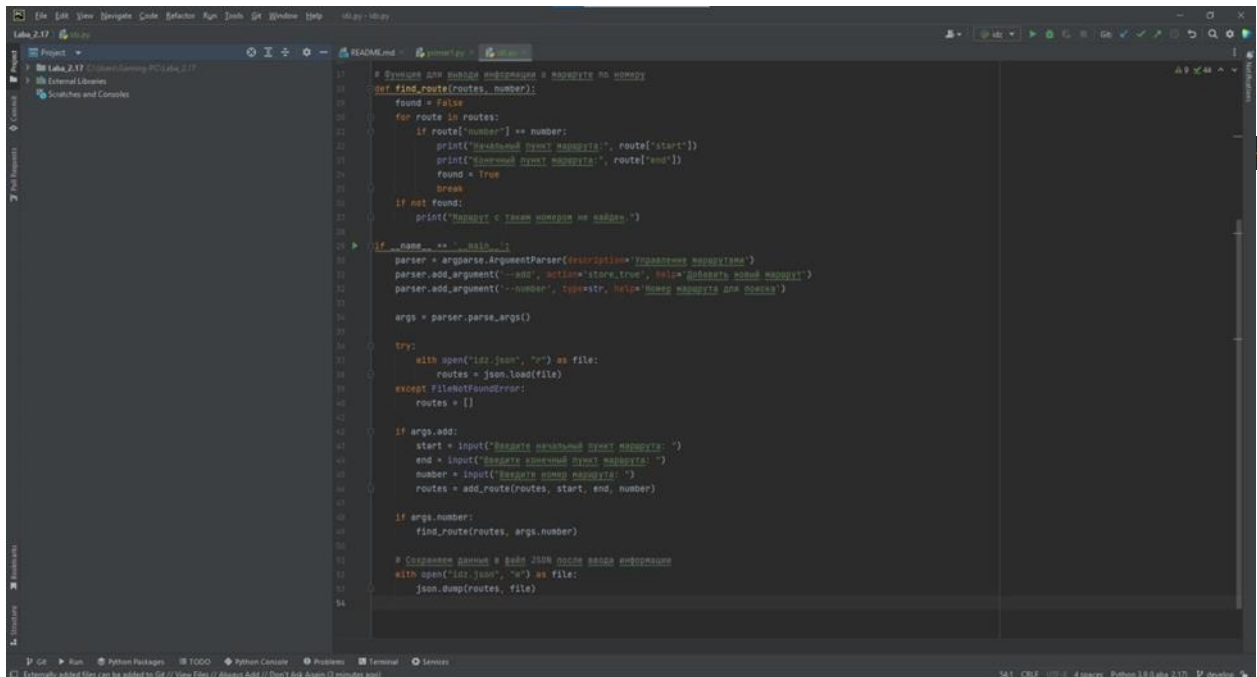


Рисунок 8. Код программы

Рисунок 9. Содержимое json файла

```

PS C:\Users\Gaming-PC\Laba_2.17> python idz.py --add
Введите начальный пункт маршрута: Харьков
Введите конечный пункт маршрута: Ставрополь
Введите номер маршрута: 21
PS C:\Users\Gaming-PC\Laba_2.17> python idz.py --add
Введите начальный пункт маршрута: Донецк
Введите конечный пункт маршрута: Ставрополь
Введите номер маршрута: 56
PS C:\Users\Gaming-PC\Laba_2.17> python idz.py --number 56
Начальный пункт маршрута: Донецк
Конечный пункт маршрута: Ставрополь
PS C:\Users\Gaming-PC\Laba_2.17>

```

Рисунок 10. Реализация программы

Задание 5.

После выполнения работы на ветке develop, слила ее с веткой main и отправила изменения на удаленный сервер. Создала файл envirement.yml и деактивировала виртуальное окружение.

```

(base) PS C:\Users\Gaming-PC\Laba_2.17> conda env export > envirement.yml
(base) PS C:\Users\Gaming-PC\Laba_2.17> conda deactivate

```

Рисунок 11. Деактивация виртуального окружения

Ссылка: <https://github.com/stefa-b/Python-lab-4.git>

Ответы на контрольные вопросы:

1. В чем отличие терминала и консоли?

Терминал - это программное обеспечение или устройство, позволяющее пользователю взаимодействовать с операционной системой. Он предоставляет текстовый интерфейс для ввода команд и получения вывода. Обычно, терминал используется для запуска команд и управления системой.

Консоль - это окно, где пользователь может вводить команды, запускать приложения и видеть вывод этих программ. Это графическое представление терминала.

2. Что такое консольное приложение?

Консольное приложение - это программа, предназначенная для выполнения в командной строке (консоли). Такие приложения обрабатывают ввод пользователя и выводят результат на консоль без графического интерфейса.

3. Какие существуют средства языка программирования Python для построения приложений командной строки?

В Python существует несколько встроенных модулей для создания CLI-приложений: `sys`, `argparse`, `getopt` и другие.

4. Какие особенности построение CLI с использованием модуля `sys`?

Модуль `sys` предоставляет доступ к некоторым переменным и функциям, связанным с интерпретатором Python. Он обеспечивает доступ к аргументам командной строки через `sys.argv`, что позволяет обрабатывать аргументы при запуске скрипта.

5. Какие особенности построение CLI с использованием модуля `getopt`?

Модуль `getopt` предоставляет функции для парсинга аргументов командной строки. Он позволяет более гибко управлять аргументами командной строки и их опциями.

6. Какие особенности построение CLI с использованием модуля `argparse`?

Модуль `argparse` является более мощным и гибким инструментом для создания интерфейса командной строки в Python. Он позволяет определять аргументы, их типы, флаги и даже создавать справочную информацию для пользователей. `argparse` автоматически генерирует справку о том, как использовать ваше CLI-приложение.

Вывод: приобрела навыки построения приложений с интерфейсом командной строки с помощью языка программирования Python версии 3.x.