|  |  |
| --- | --- |
| МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  РОССИЙСКОЙ ФЕДЕРАЦИИ | |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  “Пермский государственный национальный исследовательский университет” | |
| *Лабораторная работа №1*  *по дисциплине «Технологии разработки распределенных приложений»* | |
| Работу выполнила студентка группы КМБ-16 5 курса механико-математического факультета  Кузнецова Александра Дмитриевна  «2» октября 2020 г. |  |

СОДЕРЖАНИЕ

[Постановка задачи](#_heading=h.1fob9te) 3

[Технологии и структура приложения](#_heading=h.phjtsjge2fiv) 3

[Описание приложения](#_heading=h.lbaixyaut4w4) 4

[Соответствие поставленной задаче](#_heading=h.7t5ihf7x9puv) 12

[Список источников](#_heading=h.jjf33hanci5w) 13

# 

# 

# **Постановка задачи**

**Цель:** Изучение возможностей использования API современных популярных информационных ресурсов внешними приложениями.

**Формируемая компетенция:** способность применять на практике теоретические основы и общие принципы разработки распределенных систем.

**Организация выполнения работы:** Каждый студент выполняет индивидуальное задание.

**Требования к выполнению работы:**

1. Разработанное приложение должно предоставлять возможности создания, чтения, обновления и удаления некоторых объектов выбранного информационного ресурса (выполнения CRUD-операций).
2. CRUD-операции должны выполняться при помощи API выбранного ресурса, причём по крайней мере один тип операции должен требовать аутентификации на нём.
3. Должна быть возможность выполнения по крайней мере двух CRUD-операций для одного и того же объекта выбранного информационного ресурса.

# **Технологии и структура приложения**

В качестве информационного ресурса был выбран GitHub [1]. Приложение совместимо с семейством ОС Linux.

* Язык - python 3.7.4 [2];
* Среда разработки - PyCharm Professional 2019.3.3 [3], предоставленная по индивидуальной студенческой лицензии JetBrains [4].
* Для взаимодействия с API GitHub была использована библиотека PyGitHub 1.53 [5];
* Другие библиотеки, используемые в приложении:
  + PyYAML 5.3.1 [6];
  + cryptography 3.1.1 [7];
  + click 7.1.2 [8];

Приложение состоит из нескольких файлов:

* cmd.py - содержит определения параметров командной строки и логику выполнения команд с параметрами;
* githubapi.py - содержит функции для работы с GitHub;
* githubapi\_config.yaml - конфигурационный файл;

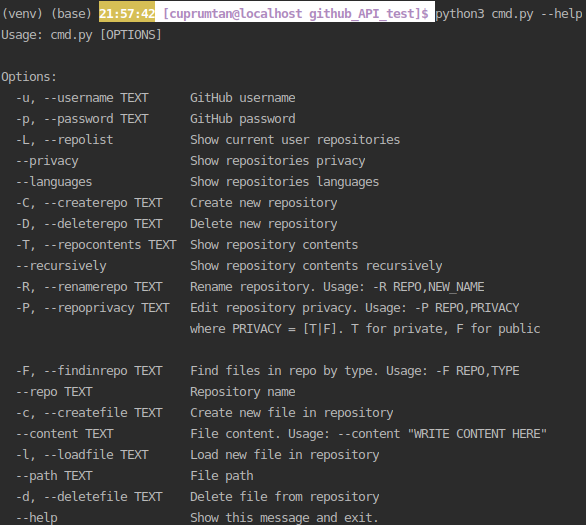
В конфигурационном файле githubapi\_config.yaml задается местоположение двух файлов, порождаемых приложением:

* log\_file - файл для журналирования событий приложения;
* cred\_file - файл для сохранения параметров последней успешно запущенной сессии;

# **Описание приложения**

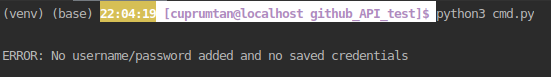
В информационном источнике были выделены две основные сущности - репозиторий и файл. Для каждой из сущностей были реализованы CRUD-операции.

Список всех флагов и их описание доступны по опции *--help*:



Приложение работает только с авторизованными пользователями. Авторизация происходит с помощью логина и пароля. При каждой успешной попытке авторизации логин и пароль пользователя сохраняются в зашифрованном виде в файл *cred\_file*. Можно задать логин и пароль явно с помощью флагов *-u* и *-p* соответственно. Если файл *cred\_file* существует и флаги *-u* и *-p* не заданы информация берется из этого файла, т.е. используется последняя успешная сессия.

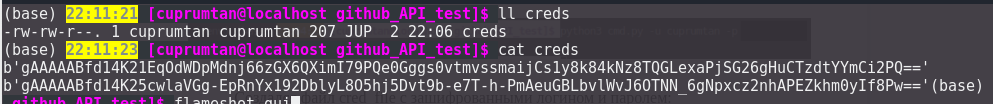
Попытка запуска приложения без флагов *-u* и *-p* и файла *cred\_file*:

**

Запуск приложения с явно заданными флагами:

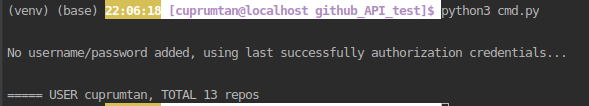


Создался файл *cred\_file* с зашифрованными логином и паролем:



Шифрование реализовано с помощью конструкции Фернета [9].

С *cred\_file* возможен вход без указания параметров:

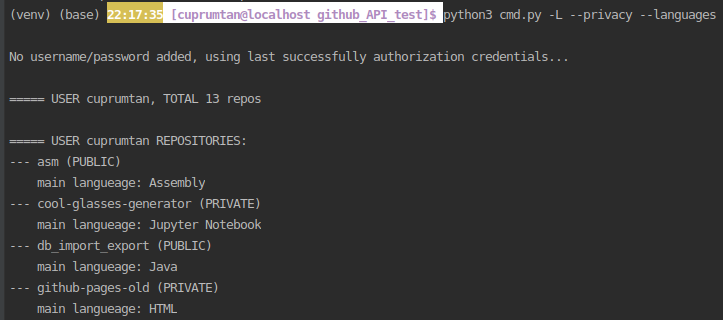


Сразу после успешной аутентификации выполняется одна из CRUD-операций - чтение списка всех репозиториев пользователя и их подсчет. На изображении выше показано, что у текущего пользователя 13 репозиториев.

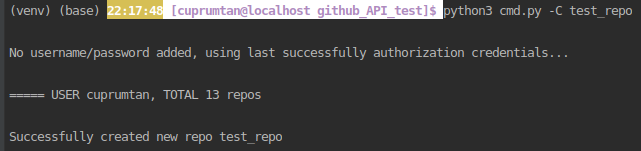
В приложении доступны операции на уровне репозиториев и на уровне файлов. Флаги для работы с репозиториями обозначены заглавными буквами, для работы с файлами - строчными.

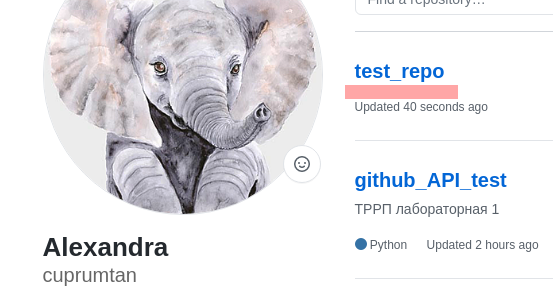
Примеры выполнения операций с репозиториями

Получение списка всех репозиториев пользователя с их статусом приватности и основным языком:

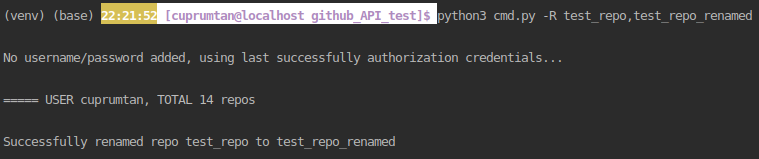


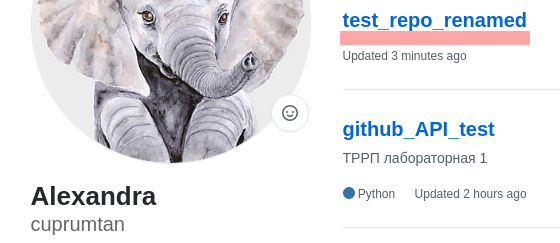
Создание репозитория:



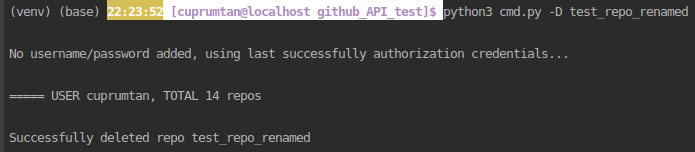


Переименование репозитория:

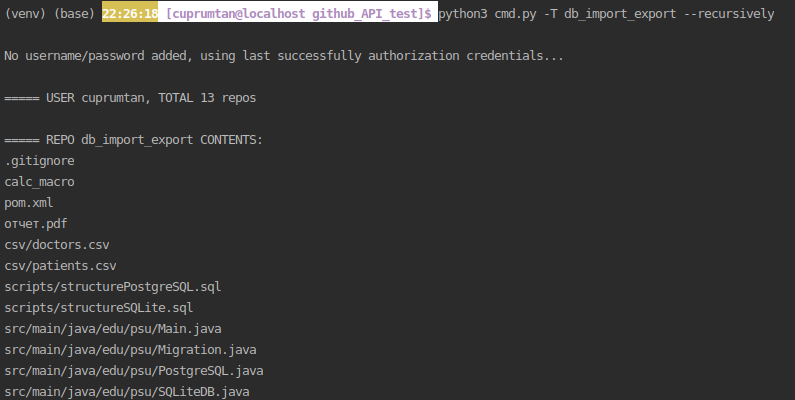




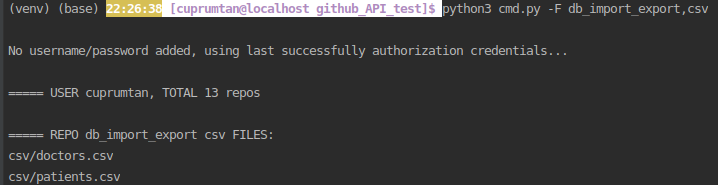
Удаление репозитория:



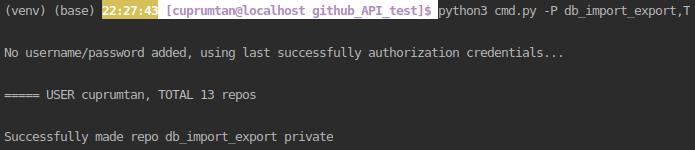
Рекурсивный вывод содержимого репозитория:



Поиск файлов определенного типа в репозитории:



Изменение видимости репозитория:

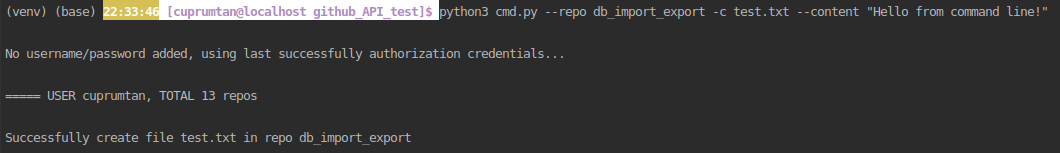


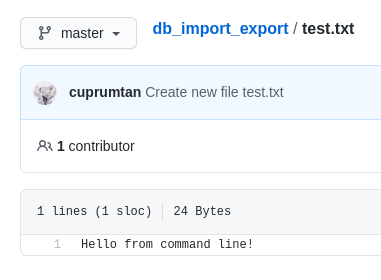


Примеры выполнения операций с файлами

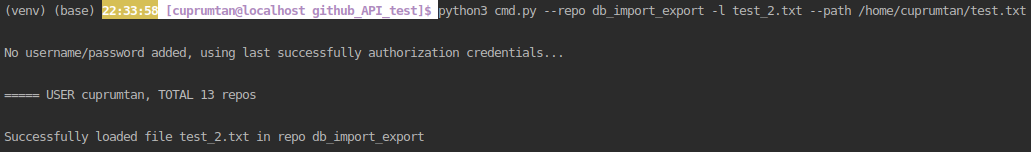
Для работы с файлами необходимо указывать название репозитория.

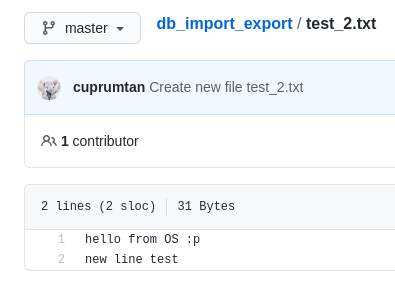
Добавление файла и его заполнение из командной строки:



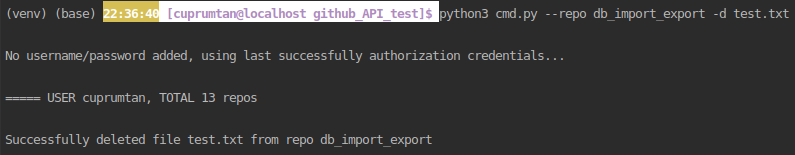


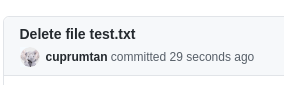
Загрузка существующего файла:





Удаление файла:





Пример содержания лог-файла:

|  |
| --- |
| 2020-10-01 22:17:35.288989 | Successfully authorized as cuprumtan  2020-10-01 22:17:47.248149 | Successfully authorized as cuprumtan  2020-10-01 22:17:48.427228 | Loaded all repositories for user cuprumtan recursively with languages  2020-10-01 22:21:49.859276 | Successfully authorized as cuprumtan  2020-10-01 22:21:52.135698 | Created new repository test\_repo for user cuprumtan  2020-10-01 22:23:47.691540 | Successfully authorized as cuprumtan  2020-10-01 22:23:52.092449 | Renamed repository test\_repo to test\_repo\_renamed for user cuprumtan  2020-10-01 22:25:11.008812 | Successfully authorized as cuprumtan  2020-10-01 22:25:12.210416 | Deleted repository test\_repo\_renamed for user cuprumtan  2020-10-01 22:26:17.447683 | Successfully authorized as cuprumtan  2020-10-01 22:26:18.468429 | Loaded repository github\_API\_test contents for user cuprumtan  2020-10-01 22:26:33.927993 | Successfully authorized as cuprumtan  2020-10-01 22:26:38.576796 | Loaded repository db\_import\_export contents recursively for user cuprumtan  2020-10-01 22:27:40.624871 | Successfully authorized as cuprumtan  2020-10-01 22:27:43.518751 | Found csv files in repository db\_import\_export for user cuprumtan  2020-10-01 22:28:29.710547 | Successfully authorized as cuprumtan  2020-10-01 22:28:31.138639 | Setted repository db\_import\_export to private for user cuprumtan  2020-10-01 22:29:34.876117 | Successfully authorized as cuprumtan  2020-10-01 22:29:37.118772 | Setted repository db\_import\_export to public for user cuprumtan  2020-10-01 22:33:56.079647 | Successfully authorized as cuprumtan  2020-10-01 22:33:57.997368 | Created file test.txt in repository db\_import\_export for user cuprumtan  2020-10-01 22:35:28.167977 | Successfully authorized as cuprumtan  2020-10-01 22:35:29.862436 | Loaded file test\_2.txt into repository db\_import\_export for user cuprumtan  2020-10-01 22:36:38.048276 | Successfully authorized as cuprumtan  2020-10-01 22:36:40.600296 | Deleted file test\_2.txt from repository db\_import\_export for user cuprumtan  2020-10-01 22:36:50.051128 | Successfully authorized as cuprumtan  2020-10-01 22:36:51.869783 | Deleted file test.txt from repository db\_import\_export for user cuprumtan |

# **Соответствие поставленной задаче**

|  |  |
| --- | --- |
| **Требование** | **Статус** |
| Приложение позволяет выполнять CRUD-операции над объектами выбранного ресурса | Выполнено.  Create-операции:   * Создание репозитория; * Создание файла;   Read-операции:   * Получение количества всех репозиториев пользователя * Получение списка всех репозиториев пользователя; * Получение списка всех файлов в репозитории;   Update-операции:   * Изменение имени репозитория; * Изменение видимости репозитория;   Delete-операции:   * Удаление репозитория; * Удаление файла; |
| Приложение позволяет выполнять по крайней мере две CRUD-операции для одного и того же объекта выбранного ресурса | Выполнено. |
| Приложение выполняет по крайней мере одну из CRUD-операций после успешной аутентификации | Выполнено. |
| Приложение не требует повторной аутентификации при перезапуске программы | Выполнено. |
| Параметры успешной аутентификации сохраняются при перезапуске программы в зашифрованном виде | Выполнено. |

# **Список источников**

1. <https://github.com/> (дата обращения 01.10.2020).
2. <https://www.python.org/downloads/release/python-374/>, лицензия <https://docs.python.org/3.7/license.html> (дата обращения 01.10.2020).
3. <https://confluence.jetbrains.com/display/PYH/PyCharm+2019.3.3+Release+Notes?_ga=2.174882918.554375686.1601653569-364471055.1569302137> (дата обращения 01.10.2020).
4. <https://www.jetbrains.com/ru-ru/community/education/#students> (дата обращения 01.10.2020).
5. <https://pypi.org/project/PyGithub/1.53/>, лицензия <https://github.com/PyGithub/PyGithub/blob/master/COPYING> (дата обращения 01.10.2020).
6. <https://pypi.org/project/PyYAML/5.3.1/>, лицензия <https://github.com/yaml/pyyaml/blob/master/LICENSE> (дата обращения 01.10.2020).
7. <https://pypi.org/project/cryptography/3.1.1/>, лицензия <https://github.com/pyca/cryptography/blob/master/LICENSE.PSF> (дата обращения 01.10.2020).
8. <https://pypi.org/project/click/7.1.2/>, лицензия <https://github.com/pallets/click/blob/master/LICENSE.rst> (дата обращения 01.10.2020).
9. <https://github.com/fernet/spec/> (дата обращения 01.10.2020).