Лабораторная работа №3

Дисциплина: Операционные системы

Иван Подоляк

Содержание

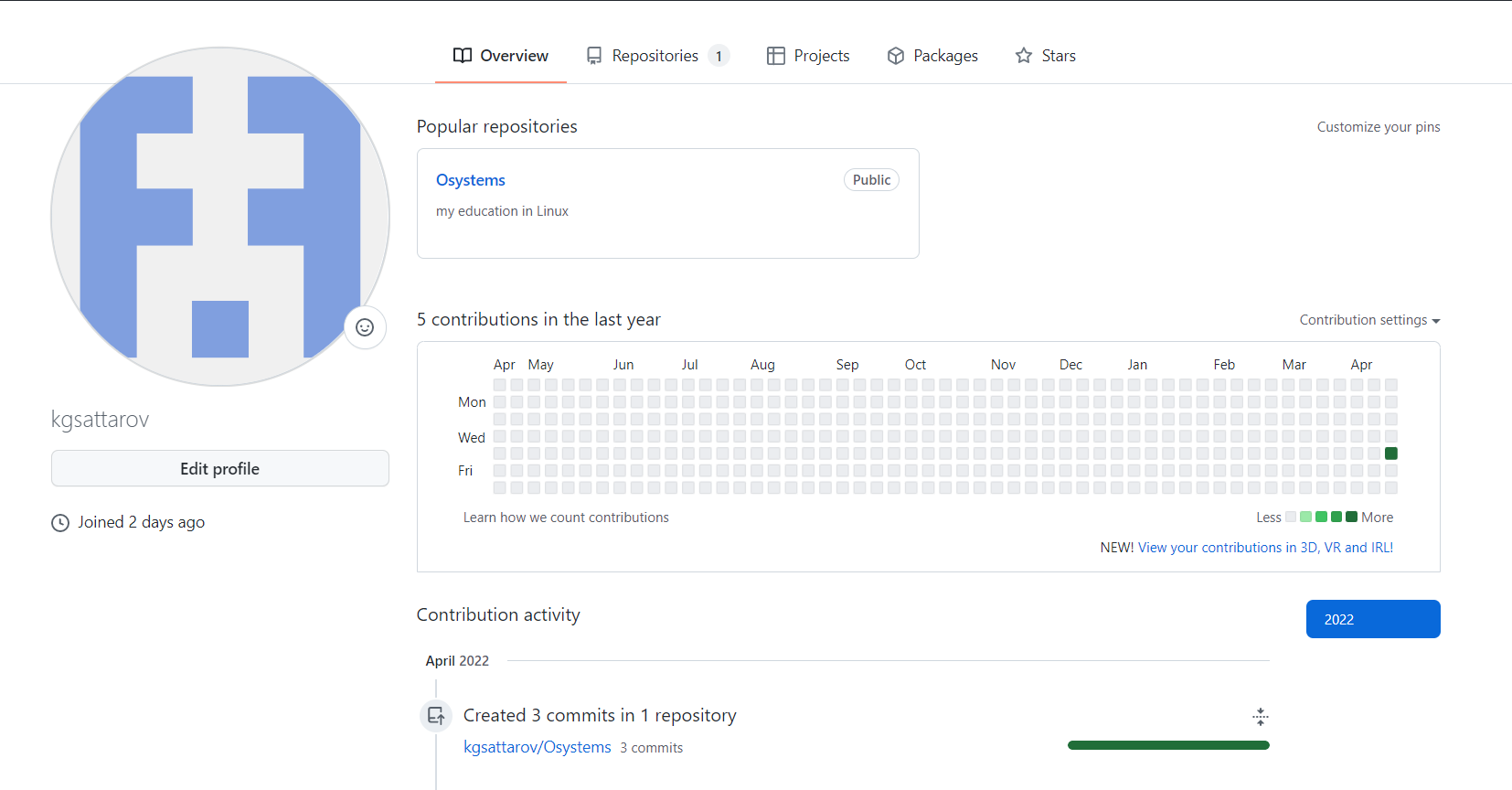
# Цель работы

Цель лабораторной работы №3 — научиться работать с языком разметки markdown

# Задание to-do

1. Сделать отчет предыдущей лабораторной работы в markdown
2. Сделать отчет в нескольких форматах: pdf, docx и md.

# Ход выполнения лабораторной работы



\1. Создаем профиль на гитхабе

Рис 1

\2. Настраиваем систему контроля версий:

git config –global user.name”Имя Фамилия”(.space) git config –global user.email”work@mail”

После этого необходимо настроить и создать новый ключ на github с помощью команды ssh-keygen -C “IDPODOLYAK[103221092@pfur.ru](mailto:103221092@pfur.ru)” и привязываем к компьютеру через консоль.

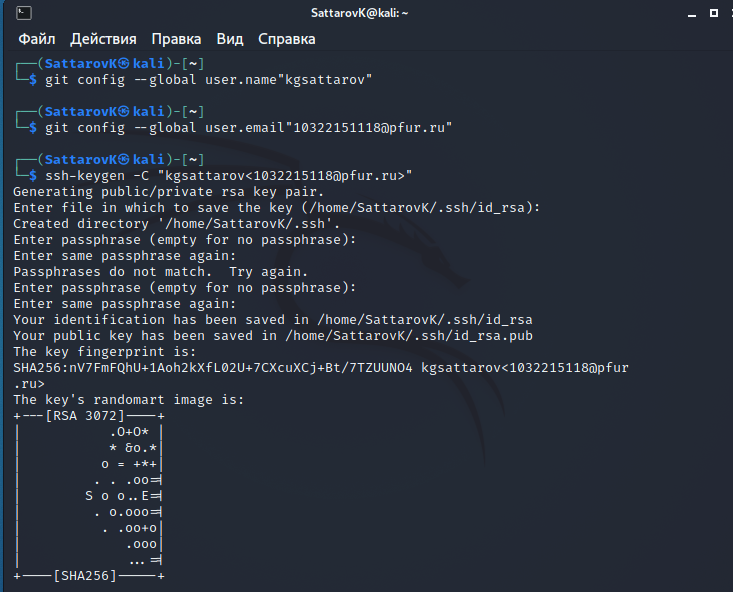


Рис 2

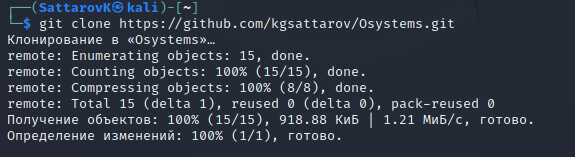
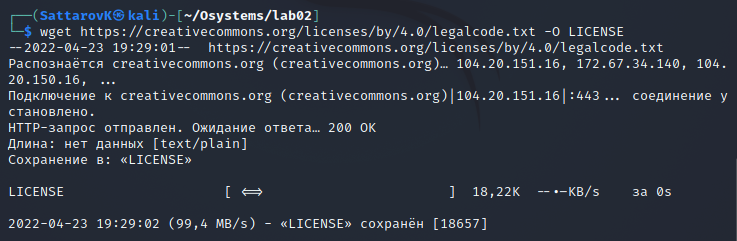
\3. Подключаем и создаем нужный репозиторий репозиторий на github: 

Рис 3

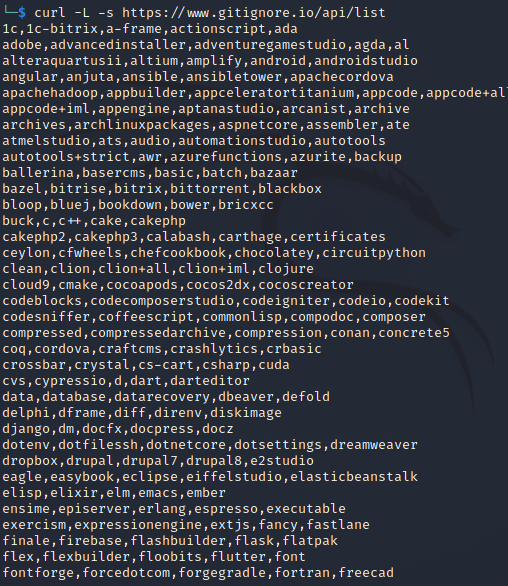
\4. Необходимо добавить несколько файлов таких как gitignore и LICENSE.



.gitignore

Рис 6

Далее шаблон игнорируемых файлов:



Игнорм файлы

Рис 7

Согласно методичке, необходимо скачать шаблон на C и “запушить его”

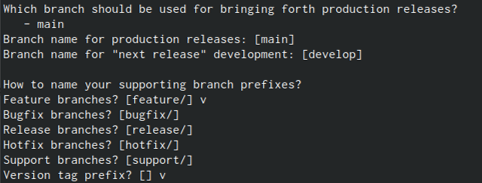
Шаблон для С

Шаблон для С

Рис 8

\5. Работа с git flow

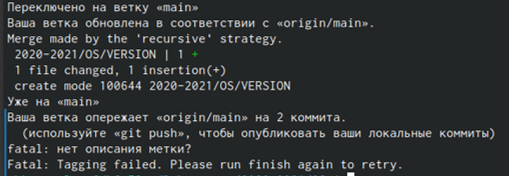
Инициализируем git-flow, используя команду git flow init -f (префикс для ярлыков установлен в v):



Git flow

Рис 9

Далее создаем релиз, куда запишем простой код со схожим python синтаксисом



Release

Рис 10

Создаем релиз на github. Для этого заходим в «Releases», нажимаем «Создать новый релиз». Заходим в теги и заполняем все поля (создаём теги для версии 1.0.0). После создания тега, автоматически сформируется релиз.

#Вывод:

Я научился основам git, которые ранее для меня были очень запутанными и сложными, а эта лабораторная дала мне практическое задание, где я получил новые знания и пролила свет на некоторые “темные” для меня области git.