

Лабораторная работа №3

Дисциплина: Операционные системы

Саттаров Константин Григорьевич

Содержание

Цель работы

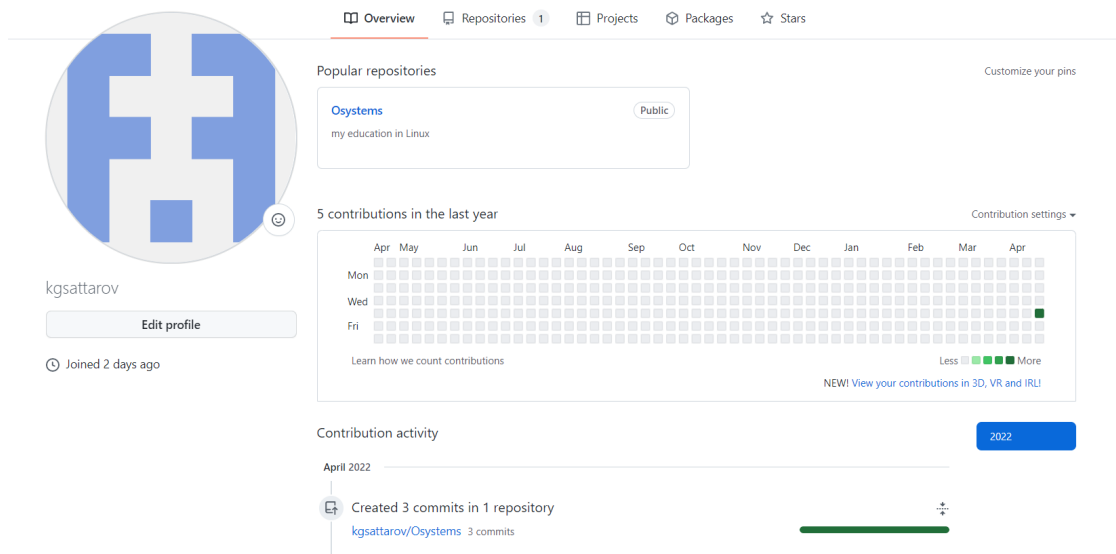
Цель лабораторной работы №3 — научиться работать с языком разметки markdown

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



01 width 70%}

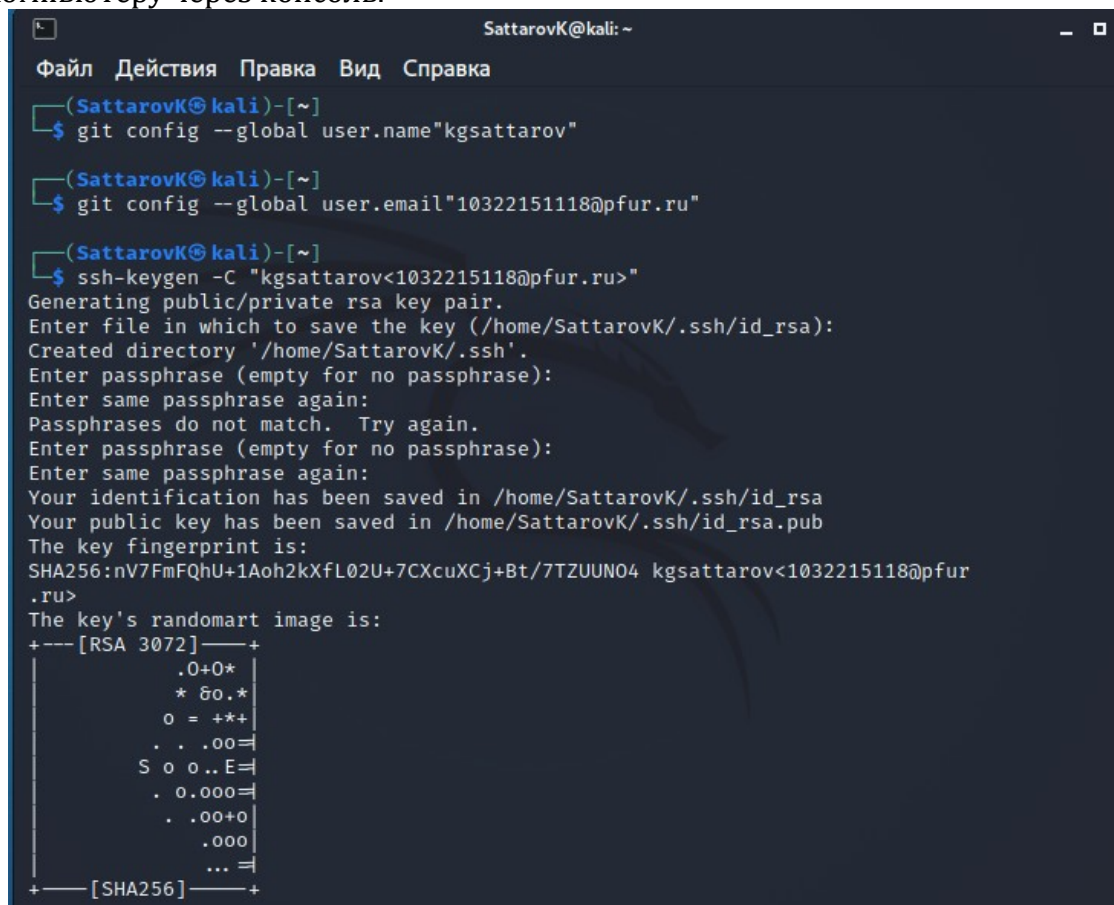
Рис 1

\2. Настраиваем систему контроля версий:

{fig:0

```
git config --global user.name "Имя Фамилия"(.space) git config --global user.email "work@mail"
```

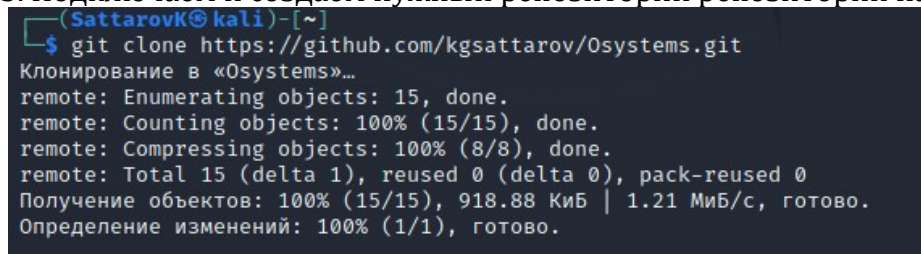
После этого необходимо настроить и создать новый ключ на github с помощью команды `ssh-keygen -C "kgsattarov1032215118@pfur.ru"` и привязываем к компьютеру через консоль.



```
SattarovK@kali: ~  
Файл Действия Правка Вид Справка  
(SattarovK@kali)-[~]  
$ git config --global user.name "kgsattarov"  
(SattarovK@kali)-[~]  
$ git config --global user.email "1032215118@pfur.ru"  
(SattarovK@kali)-[~]  
$ ssh-keygen -C "kgsattarov<1032215118@pfur.ru>"  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/SattarovK/.ssh/id_rsa):  
Created directory '/home/SattarovK/.ssh'.  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Passphrases do not match. Try again.  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/SattarovK/.ssh/id_rsa  
Your public key has been saved in /home/SattarovK/.ssh/id_rsa.pub  
The key fingerprint is:  
SHA256:nV7FmFqHU+1Aoh2kXfL02U+7CXcuXCj+Bt/7TZUUN04 kgsattarov<1032215118@pfur.ru>  
The key's randomart image is:  
+---[RSA 3072]---+  
|.0+0*|  
|* 80.*|  
|O = +++|  
|. . .00=|  
|S O o..E=|  
|. o.000=|  
|. .00+0|  
|. .000|  
|...=|  
+---[SHA256]---+
```

Рис 2

\3. Подключаем и создаем нужный репозиторий на github:



```
(SattarovK@kali)-[~]  
$ git clone https://github.com/kgsattarov/Osystems.git  
Клонирование в «Osystems»...  
remote: Enumerating objects: 15, done.  
remote: Counting objects: 100% (15/15), done.  
remote: Compressing objects: 100% (8/8), done.  
remote: Total 15 (delta 1), reused 0 (delta 0), pack-reused 0  
Получение объектов: 100% (15/15), 918.88 Киб | 1.21 Миб/с, готово.  
Определение изменений: 100% (1/1), готово.
```

Рис 3

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

```
(SattarovK@kali)-[~/0systems/lab02]
└─$ wget https://creativecommons.org/licenses/by/4.0/legalcode.txt -O LICENSE
--2022-04-23 19:29:01-- https://creativecommons.org/licenses/by/4.0/legalcode.txt
Распознаётся creativecommons.org (creativecommons.org)... 104.20.151.16, 172.67.34.140, 104.
20.150.16, ...
Подключение к creativecommons.org (creativecommons.org)|104.20.151.16|:443 ... соединение у
становлено.
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: нет данных [text/plain]
Сохранение в: «LICENSE»

LICENSE [ ⇐ ] 18,22K -- *KB/s за 0s

2022-04-23 19:29:02 (99,4 MB/s) - «LICENSE» сохранён [18657]
```

.gitignore

Рис 6

Далее шаблон игнорируемых файлов:

```
└─$ curl -L -s https://www.gitignore.io/api/list
1c,1c-bitrix,a-frame,actionscript,ada
adobe,advancedinstaller,adventuregamestudio,agda,al
alteraquartusii,altium,amplify,android,androidstudio
angular,anjuta,ansible,ansibletower,apachecordova
apachehadoop,appbuilder,appcelerator,titanium,appcode,appcode+al
appcode+iml,appengine,aptanastudio,arcanist,archive
archives,archlinuxpackages,aspnetcore,assembler,ate
atmelstudio,ats,audio,automationstudio,autotools
autotools+strict,awr,azurefunctions,azurite,backup
ballerina,basercms,basic,batch,bazaar
bazel,bitrise,bitrix,bittorrent,blackbox
bloop,bluej,bookdown,bower,bricxcc
buck,c,c++,cake,cakephp
cakephp2,cakephp3,calabash,carthage,certificates
ceylon,cfwheels,cheffcookbook,chocolatey,circuitpython
clean,clion,clion+all,clion+iml,clojure
cloud9,cmake,cocoapods,cocos2dx,cocoscreator
codeblocks,codecomposerstudio,codeigniter,codeio,codekit
codesniffer,coffeescript,commonlisp,compodoc,composer
compressed,compressedarchive,compression,conan,concrete5
coq,cordova,craftcms,crashlytics,crbasic
crossbar,crystal,cs-cart,csharp,cuda
cvs,cypressio,d,dart,darteditor
data,database,datarecovery,dbeaver,defold
delphi,dframe,diff,direnv,diskimage
django,dm,docfx,docpress,docz
dotenv,dotfilessh,dotnetcore,dotsettings,dreamweaver
dropbox,drupal,drupal7,drupal8,e2studio
eagle,easybook,eclipse,eiffelstudio,elasticbeanstalk
elisp,elixir,elm,emacs,ember
ensime,epi-server,erlang,espresso,executable
exercism,expressionengine,extjs,fancy,fastlane
finale,firebase,flashbuilder,flask,flatpak
flex,flexbuilder,floobits,flutter,font
fontforge,forcedotcom,forgegradle,fortran,freecad
```

Игнорм файлы

Рис 7

Согласно методичке, необходимо скачать шаблон на С и “запустить его”

```
(SattarovK@kali)-[~/Osystems/lab02]
$ curl -L -s https://www.gitignore.io/api/c >> .gitignore
```

Шаблон для C

Рис 8

\5. Работа с git flow

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

```
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/] v
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? [] v
```

Git flow

Рис 9

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

```
Переключено на ветку «main»
Ваша ветка обновлена в соответствии с «origin/main».
Merge made by the 'recursive' strategy.
2020-2021/OS/VERSION | 1 +
1 file changed, 1 insertion(+)
create mode 100644 2020-2021/OS/VERSION
Уже на «main»
Ваша ветка опережает «origin/main» на 2 коммита.
(используйте «git push», чтобы опубликовать ваши локальные коммиты)
fatal: нет описания метки?
Fatal: Tagging failed. Please run finish again to retry.
```

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.