ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

дисциплина: Операционные системы

Выполнила:
Егорова Александра
Группа: НПМбд-02-20

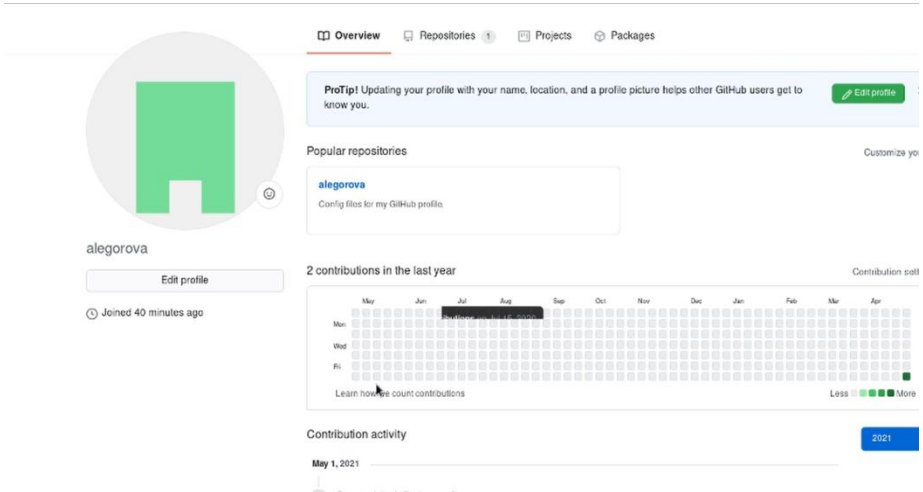
Цель работы

Изучить идеологию и применение средств контроля версий.

Ход работы

- **Настройка git**

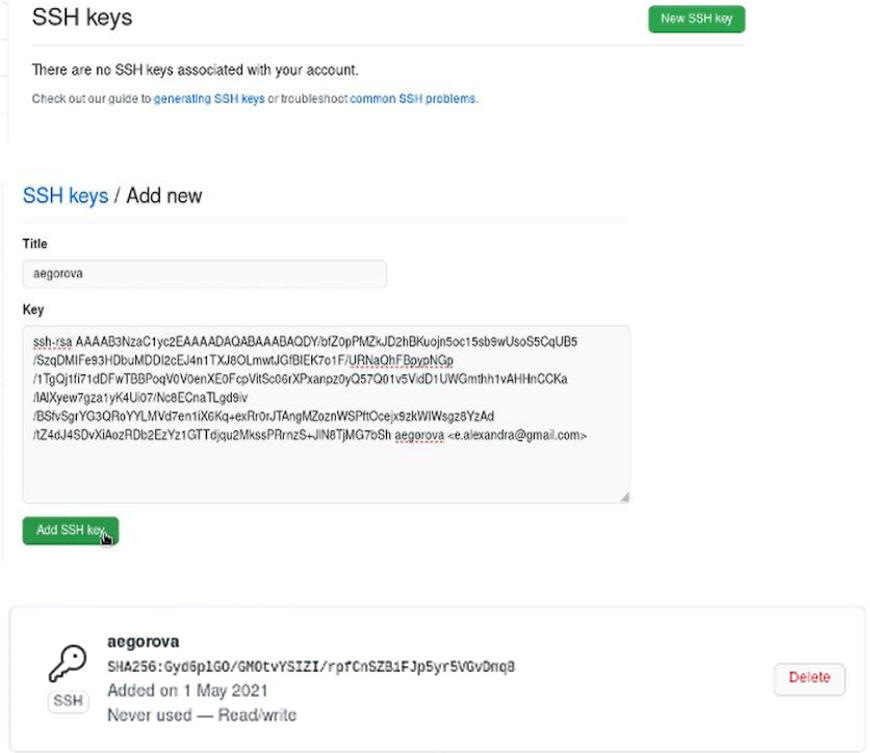
1) Создаем учётную запись на https://github.com. Настраиваем систему контроля версий git. Синхранизируем учётную запись github с компьютером




```
Файл  Правка  Вид  Поиск  Терминал  Справка
[aegorova@aegorova ~]$ git config --global user.name "aegorova"
[aegorova@aegorova ~]$ git config --global user.email "e.alexandra2002@gmail.com"
```

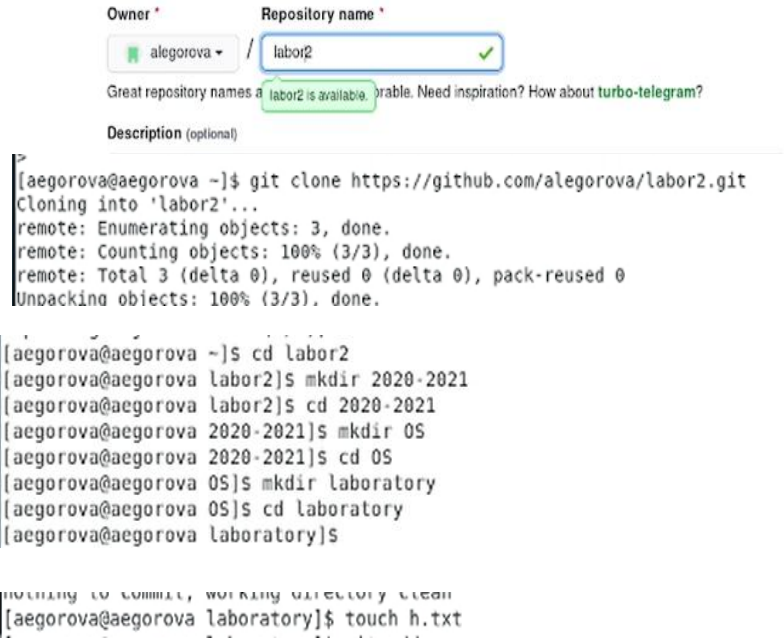
Создаем новый ключ на github и привязываю его к компьютеру через консоль.

```
[aegorova@aegorova ~]$ ssh-keygen -C "aegorova <e.alexandra@gmail.com>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/aegorova/.ssh/id_rsa):
/home/aegorova/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/aegorova/.ssh/id_rsa.
Your public key has been saved in /home/aegorova/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:Gyd6plGO/GMOtvYSIZI/rpfCnSZBiFJp5yr5VGvDmq8 aegorova <e.alexandra@gmail.com>
The key's randomart image is:
+---[RSA 2048]----+
|    .            |
|   + .           |
|.+ +             |
|+ + + .          |
|.o * o .S .      |
|o + B..= =       |
| = B =B.=        |
|  B B.oOo        |
|  EO..o=+.       |
+----[SHA256]-----+
[aegorova@aegorova ~]$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDY/bfZ0pPMZkJD2hBKuojn5oc15sb9wUsoS5CqUB5/SzqDMIFe93HDbuMDDI2cEJ4
n1TXJ8OLmwtJGfBIEK7o1F/URNaQhFBpypNGp/1TgQj1fi71dDFwTBBPoqV0V0enXE0FcpVitSc06rXPxanpz0yQ57Q01v5VidD1UWG
mthh1vAHHnCCKa/lAlXyew7gza1yK4Ui07/Nc8ECnaTLgd9iv/B5fvSgrYG3QRoYYLMVd7en1iX6Kq+exRr0rJTAngMZoznWSPftOce
jx9zkWIWsgz8YzAd/tZ4dJ45DvXiAozRDb2EzYz1GTTdjqu2MkssPRrnzS+JlN8TjMG7bSh aegorova <e.alexandra@gmail.com
```



- **Подключение репозитория к github.**

Заходим в «repositories» и создаем новый репозиторий. Копируем в консоль ссылку на репозиторий. Работаем с каталогом и папками через консоль. Создаем файлы



Добавляем первый коммит и выкладываем на githup. Чтобы правильно разместить первый коммит, необходимо добавить команду git add . , далее с помощью команды git commit -am "first commit" выкладываем коммит. Сохраняем первый коммит (git push).

```
[aegorova@aegorova laboratory]$ git add .
[aegorova@aegorova laboratory]$ git commit -am "first commit"
[main 98f3c66] first commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 2020-2021/OS/laboratory/h.txt
```

```
[aegorova@aegorova laboratory]$ git push
warning: push.default is unset; its implicit value is changing in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the current behavior after the default changes, use:

  git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

  git config --global push.default simple

See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Username for 'https://github.com': alegorova
Password for 'https://alegorova@github.com':
Counting objects: 7, done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (6/6), 406 bytes | 0 bytes/s, done.
Total 6 (delta 0), reused 0 (delta 0)
To https://github.com/alegorova/labor2.git
   5332bfc..98f3c66  main -> main
```

- **Первичная конфигурация**

Добавим файл лицензии и шаблон игнорируемых файлов. Получим список имеющихся шаблонов.

```
[aegorova@aegorova laboratory]$ wget https://creativecommons.org/licenses/by/4.0/legalcode.txt -O LICENSE
--2021-05-01 23:21:33--  https://creativecommons.org/licenses/by/4.0/legalcode.txt
Распознаётся creativecommons.org (creativecommons.org)... 172.67.34.140, 104.20.150.16, 104.20.151.16, ...
Подключение к creativecommons.org (creativecommons.org)|172.67.34.140|:443... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: нет данных [text/plain]
Сохранение в: «LICENSE»

    [ <=>                                                                ] 18 657      --.-K/s   за 0s

2021-05-01 23:21:35 (71,3 MB/s) - «LICENSE» сохранён [18657]
```

```
[aegorova@aegorova laboratory]$ 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,apachecordova,apachehadoop
appbuilder,appceleratortitanium,appcode,appcode+all,appcode+iml
appengine,aptanastudio,arcanist,archive,archives
archlinuxpackages,aspnetcore,assembler,ate,atmelstudio
ats,audio,automationstudio,autotools,autotools+strict
awr,azurefunctions,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
chefcookbook,chocolatey,clean,clion,clion+all
clion+iml,clojure,cloud9,cmake,cocoapods
cocos2dx,cocoscreator,code,code-java,codeblocks
```

Затем скачаем шаблон, например, для C. Также добавляю новые файлы и выполняю коммит.

```
[aegorova@aegorova laboratory]$ curl -L -s https://www.gitignore.io/api/c >> .gitignore
[aegorova@aegorova laboratory]$ git add .
[aegorova@aegorova laboratory]$ git commit -am "first commit"
[main 46ed2b4] first commit
 2 files changed, 455 insertions(+)
 create mode 100644 2020-2021/OS/laboratory/.gitignore
 create mode 100644 2020-2021/OS/laboratory/LICENSE
```

Отправляем на github.

```
[aegorova@aegorova laboratory]$ git push
warning: push.default is unset; its implicit value is changing in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the current behavior after the default changes, use:

  git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

  git config --global push.default simple

See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Username for 'https://github.com': alegorova
Password for 'https://alegorova@github.com':
Counting objects: 11, done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (7/7), 6.59 KiB | 0 bytes/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To https://github.com/alegorova/labor2.git
   98f3c66..46ed2b4  main -> main
```

- **Конфигурация git-flow**

Инициализируем git-flow с помощью команды git flow init -f. Префикс для ярлыков установим в v.

```
[aegorova@aegorova laboratory]$ git flow init -f

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

Проверяем, что находимся на ветке develop (git branch).

```
[aegorova@aegorova laboratory]$ git branch
* develop
  main
```

Создадим релиз с версией 1.0.0.

```
[aegorova@aegorova laboratory]$ git flow release start 1.0.0
Switched to a new branch 'release/1.0.0'

Summary of actions:
- A new branch 'release/1.0.0' was created, based on 'develop'
- You are now on branch 'release/1.0.0'

Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

     git flow release finish '1.0.0'
```

Запишем версию и добавим в индекс.

```
[aegorova@aegorova laboratory]$ echo "1.0.0">> VERSION
[aegorova@aegorova laboratory]$ git add .
[aegorova@aegorova laboratory]$ git commit -am 'chore(main): add version'
[release/1.0.0 6fbbf60] chore(main): add version
 1 file changed, 2 insertions(+)
 create mode 100644 2020-2021/OS/laboratory/VERSION
```

Зальём релизную ветку в основную ветку.

```
[aegorova@aegorova laboratory]$ git flow release finish 1.0.0
Switched to branch 'main'
Merge made by the 'recursive' strategy.
 2020-2021/OS/laboratory/VERSION | 2 ++
 1 file changed, 2 insertions(+)
 create mode 100644 2020-2021/OS/laboratory/VERSION
fatal: no tag message?
Tagging failed. Please run finish again to retry.
```

Отправим данные на github.

```
[aegorova@aegorova laboratory]$ git push --all
Username for 'https://github.com': alegorova
Password for 'https://alegorova@github.com':
Counting objects: 11, done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (7/7), 623 bytes | 0 bytes/s, done.
Total 7 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/alegorova/labor2.git
   46ed2b4..787aa68  main -> main
 * [new branch]      develop -> develop
 * [new branch]      release/1.0.0 -> release/1.0.0
[aegorova@aegorova laboratory]$ git push --tags
Username for 'https://github.com': alegorova
Password for 'https://alegorova@github.com':
Everything up-to-date
```

Создаем релиз на github. Заходим в теги и заполняем все поля. После создания тега, автоматически сформируется релиз.

Вывод:

В ходе выполнения лабораторной работы я изучила идеологию и применение средств контроля версий