

# Отчёт по лабораторной работе №3

## Дисциплина: Операционные системы

Аветисян Давид Артурович

### Содержание

Цель работы .....	1
Задание.....	1
Выполнение лабораторной работы .....	1
Выводы .....	6

### Цель работы

Целью данной работы является изучение идеологии и применение средств контроля версий.

### Задание

Сделайте отчёт по предыдущей лабораторной работе в формате Markdown. В качестве отчёта просьба предоставить отчёты в 3 форматах:pdf,docx и md (в архиве,поскольку он должен содержать скриншоты, Makefile и т.д.)

### Выполнение лабораторной работы

Создаем учётную запись на <https://github.com> (рис. -fig. 1).

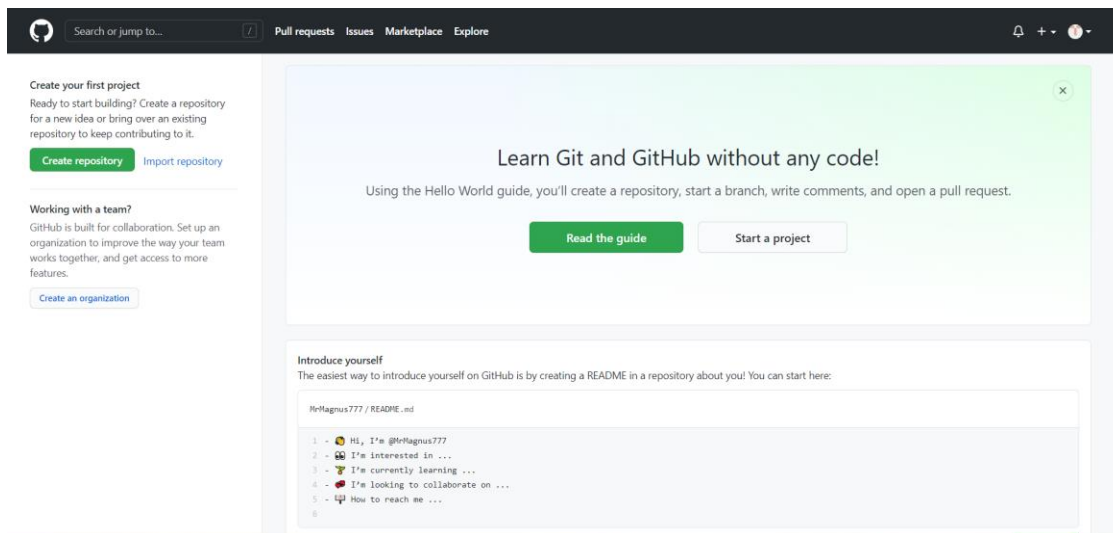


Figure 1: Создание учётной записи

- 1) Настраиваем систему контроля версий git.
  - Синхронизируем учётную запись github с компьютером (рис. -fig. 2):
  - `git config --global user.name "David Avetisyan"`
  - `git config --global user.email "David3777@yandex.ru"`
  - Затем создаём новый ключ на github `ssh-keygen -C "David AvetisyanDavid3777@yandex.ru"` (рис. -fig. 3, рис. -fig. 4) и привязываем его к компьютеру через консоль (рис. -fig. 5).

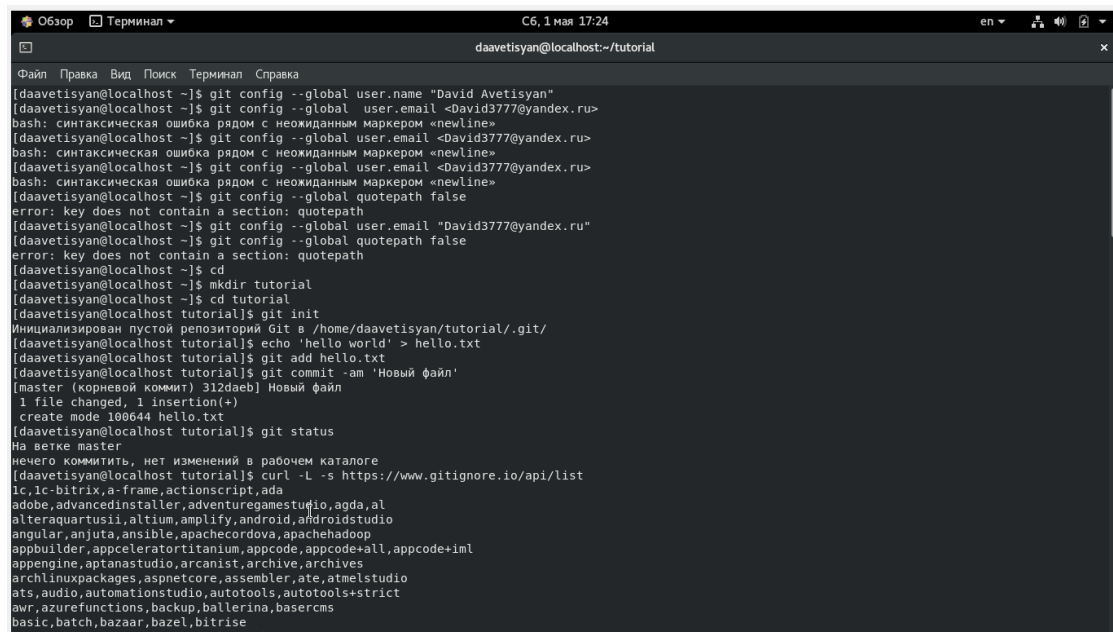


Figure 2: Синхронизация учётной записи

```
Обзор Терминал C6, 1 мая 17:42 en
daavetisyan@localhost:~/tutorial

Файл Правка Вид Поиск Терминал Справка

[daavetisyan@localhost tutorial]$ curl -L -s https://www.gitignore.io/api/c >> .gitignore
[daavetisyan@localhost tutorial]$ curl -L -s https://www.gitignore.io/api/c++ >> .gitignore
[daavetisyan@localhost tutorial]$ ssh-keygen -C "David Avetisyan <David3777@yandex.ru>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/daavetisyan/.ssh/id_rsa):
Created directory '/home/daavetisyan/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/daavetisyan/.ssh/id_rsa.
Your public key has been saved in /home/daavetisyan/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:jjXzdsbm9Mcxd3utBircNqfMHu9u1hDcc/eUBKyU54Pg David Avetisyan <David3777@yandex.ru>
The key's randomart image is:
+--[RSA 3072]----+
  o . .
  . . . o .
  .. oo.=.+
  ES o*.o+oo|
  . =.o=*|
  . .+OB|
  .o*.o|
  o==B|
+-----+
[daavetisyan@localhost tutorial]$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCcNcCLZbGfLrZy9aiihKXft5yYfjbrCV3D0YwCL7bw0sRiEzAHGrogB/VoNG58lM/8wxWfCkwoNynWe0F+dX/Zq8STlq2KfvNcPSjJ633urtVv
CjI07xai8Rhd6Ye7jssx4w6g15cIoquvzk4RmVevZl1R6Kq0KI/Nx2qgo4JfSvGrBBR0HeSZXn43Pxj/u29svrP4oo7KqFDhGnRx06bQ7nmG7m608262aTklCMGK/rKrdAnti2p6MYMBUa74uyg1
6xxxdK670usfXyMmw55SL3QXS3S+Cf35JvW4kHtfrtI+cgLIGRHNCf7mclBXcd0GyDaxLy+xy155UsJPBg/ee06i4eMRizR8xAmUND0m/LBhvl2YPhfLVDuTIZs7Moc/7kQ68IJwPX5W88r/yhiCo
mh96yd44UXS0q0UR9iOvtanB9hIQ+CbiIPmEfUMktgkLIYiziEKHYERYqAHxJW+je9K0t6eV3Q64PQAHpxby1zQTUfbqKPqd9sgeHc= David Avetisyan <David3777@yandex.ru>
[daavetisyan@localhost tutorial]$ git remote add origin
использование: git remote add [<опции>] <имя> <url>

-f, --fetch          извлечь внешние ветки
--tags              импортировать все метки и ассоциированные объекты при извлечении
                    или не извлекать метки вообще (--no-tags)
-t, --track <ветка> отслеживать ветки
-m, --master <ветка> мастер ветка
```

Figure 3: Создание нового ssh ключа

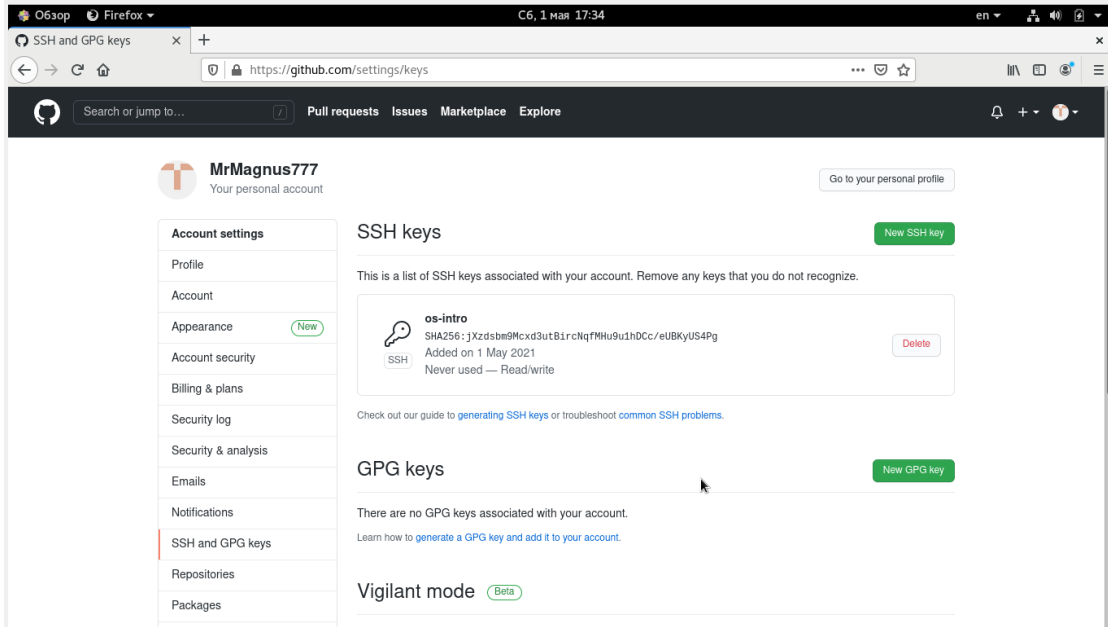


Figure 4: Добавление ssh ключа на github

```
Обзор Терминал C6, 1 мая 17:51 en 40
daavetisyan@localhost:~/tutorial

Файл Правка Вид Поиск Терминал Справка
fatal: Не удалось прочитать из внешнего репозитория.

Удостоверьтесь, что у вас есть необходимые права доступа
и репозиторий существует.
[daavetisyan@localhost tutorial]$ sudo git push -u origin master
[sudo] пароль для daavetisyan:
ssh: Could not resolve hostname github.com:MrMagnus777: Name or service not known
fatal: Не удалось прочитать из внешнего репозитория.

Удостоверьтесь, что у вас есть необходимые права доступа
и репозиторий существует.
[daavetisyan@localhost tutorial]$ git remote add origin ssh://git@github.com/MrMagnus777/work.git
fatal: внешний репозиторий origin уже существует
[daavetisyan@localhost tutorial]$ git push -u origin main
error: src refspec main does not match any
error: не удалось отправить некоторые сценки в «ssh://github.com/MrMagnus777/work.git»
[daavetisyan@localhost tutorial]$ git remote add origin https://github.com/MrMagnus777/work.git
fatal: внешний репозиторий origin уже существует
[daavetisyan@localhost tutorial]$ git push -u origin main
error: src refspec main does not match any
error: не удалось отправить некоторые сценки в «https://github.com/MrMagnus777/work.git»
[daavetisyan@localhost tutorial]$ git remote set-url origin https://github.com/MrMagnus777/work.git
[daavetisyan@localhost tutorial]$ git push -u origin main
error: src refspec main does not match any
error: не удалось отправить некоторые сценки в «https://github.com/MrMagnus777/work.git»
[daavetisyan@localhost tutorial]$ git branch -M main
[daavetisyan@localhost tutorial]$ git push -u origin main
Gtk-Message: 17:50:12.068: Failed to load module "canberra-gtk-module"
Gtk-Message: 17:50:31.357: Failed to load module "canberra-gtk-module"
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 247 bytes | 247.00 KiB/s, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/MrMagnus777/work.git
 * [new branch]      main -> main
Ветка «main» отслеживает внешнюю ветку «main» из «origin».
[daavetisyan@localhost tutorial]$
```

Figure 5: Привязка через консоль

## 2) Созданием и подключаем репозиторий к github.

- На сайте заходим в «repository» и создаём новый репозиторий под названием work. Переносим его на наш компьютер (рис. -fig. 6).

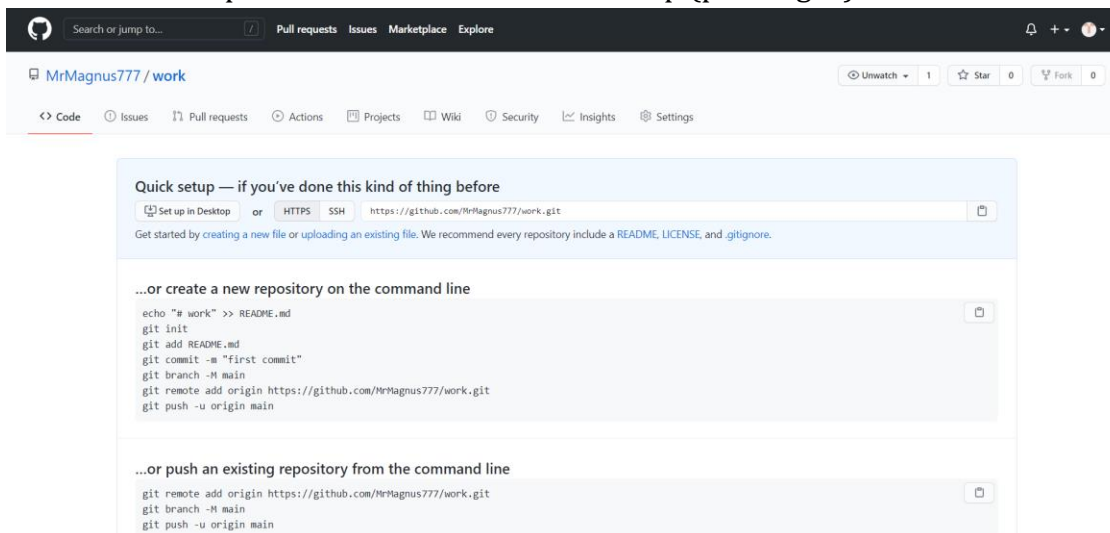


Figure 6: Создание нового репозитория

Создаем рабочий каталог (рис. -fig. 7).

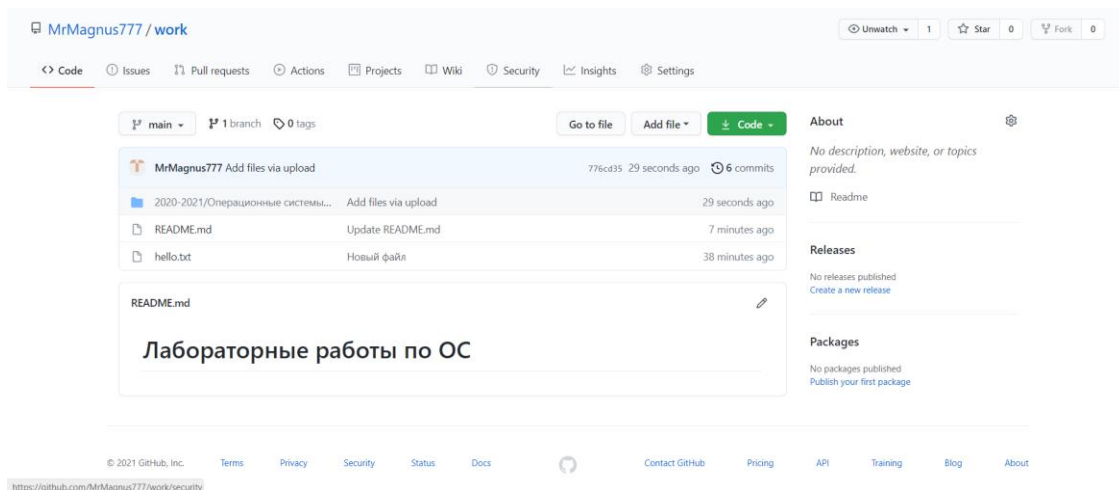


Figure 7: Создание рабочего каталога

Добавляем первый commit и выкладываем на github. Для того, чтобы правильно разместить первый коммит, необходимо добавить команду git add ., после этого с помощью команды git commit -m "first commit" выкладываем коммит. Сохраняем первый коммит, используя команду git push (рис. -fig. 8).

```

Обзор Терминал C6, 1 мая 18:26 en
daavetisyan@localhost:~/tutorial/work/2020-2021/Операционные системы/laboratory

Файл Правка Вид Поиск Терминал Справка
[daavetisyan@localhost 2020-2021]$ ls
'Операционные системы'
[daavetisyan@localhost 2020-2021]$ cd
[daavetisyan@localhost ~]$ cd ~/tutorial/work/2020-2021
[daavetisyan@localhost 2020-2021]$ ls
'Операционные системы'
[daavetisyan@localhost 2020-2021]$ cd 'Операционные системы'
[daavetisyan@localhost Операционные системы]$ ls
laboratory
[daavetisyan@localhost Операционные системы]$ cd laboratory
[daavetisyan@localhost laboratory]$ git init
Инициализирован пустой репозиторий Git в /home/daavetisyan/tutorial/work/2020-2021/Операционные системы/laboratory/.git/
[daavetisyan@localhost laboratory]$ echo '# Лабораторные работы' >> README.md
[daavetisyan@localhost laboratory]$ git add README.md
[daavetisyan@localhost laboratory]$ git commit -m 'first commit'
[master (корневой коммит) 3474e9f] first commit
1 file changed, 1 insertion(+)
 create mode 100644 README.md
[daavetisyan@localhost laboratory]$ git remote add origin git@github.com:MrMagnus777/work.git
[daavetisyan@localhost laboratory]$ git push -u origin master
The authenticity of host 'github.com (140.82.121.3)' can't be established.
RSA key fingerprint is SHA256:nThbg6kXUpJWGL7EIIG0CsPROMTxdCARLVIKw6ESSY8.
Are you sure you want to continue connecting (yes/no/fingerprint)? yes
Warning: Permanently added 'github.com,140.82.121.3' (RSA) to the list of known hosts.
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 249 bytes | 124.00 KiB/s, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/MrMagnus777/work/pull/new/master
remote:
To github.com:MrMagnus777/work.git
 * [new branch]      master -> master
Ветка «master» отслеживает внешнюю ветку «master» из «origin».
[daavetisyan@localhost laboratory]$

```

Figure 8: Добавляем первый commit

### 3) Первичная конфигурация.

- Добавляем файл лицензии. Добавляем шаблон игнорируемых файлов. Просматриваем список имеющихся шаблонов (рис. -fig. 9). Скачиваем шаблон (например, для C) и выполняем коммит. Отправляем на github (команда git push) (рис. -fig. 10).

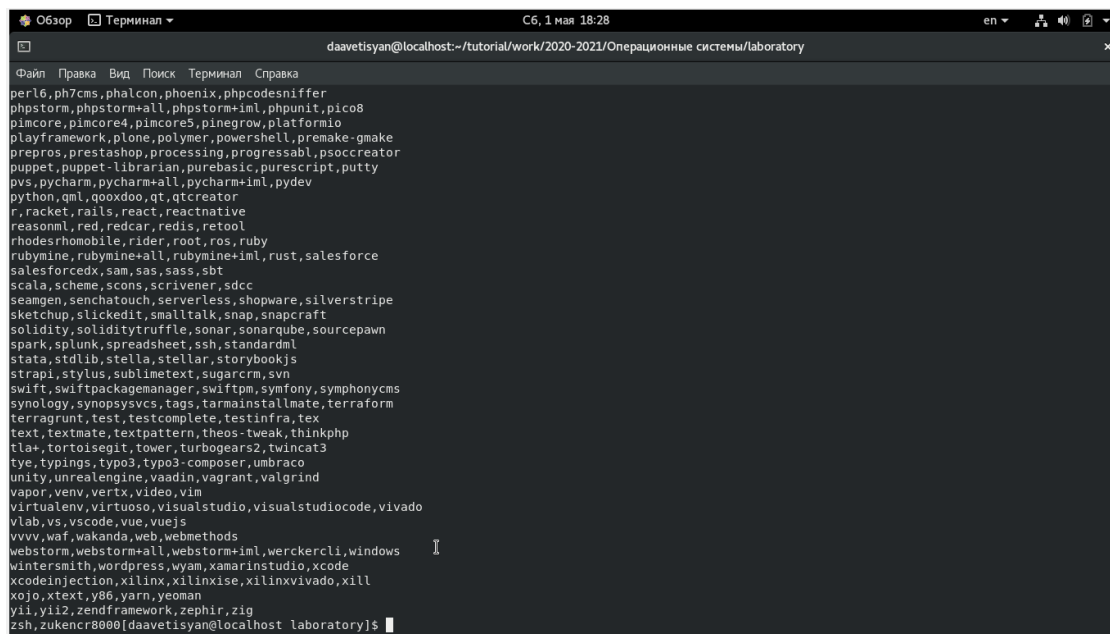


Figure 9: Просматриваем список шаблонов

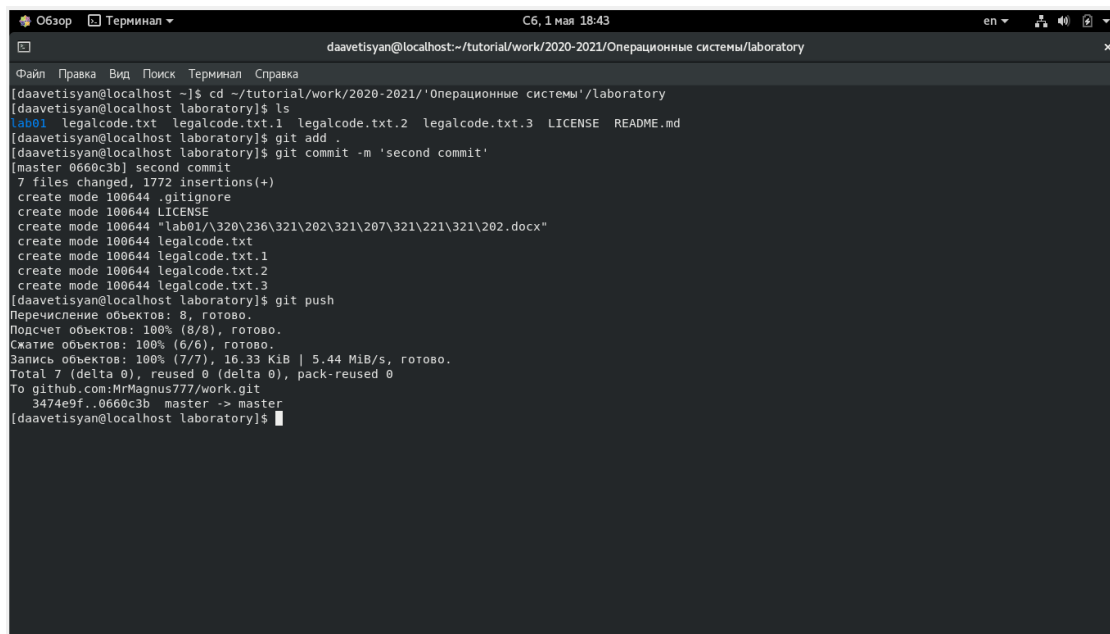


Figure 10: Скачиваем шаблон и отправляем на github

- 4) Работаем с конфигурацией git-flow.
  - У нас не получилось установить git-flow, так как root этого не допустил. В связи с этим дальнейшие действия выполнить невозможно.

## Выводы

Я изучил идеологию и применение контроля версий.