

# **отчёта по лабораторной работе №2**

**Управление версиями**

Джумаев Бегенч

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
3.1	Настройка git . . . . .	7
<b>4</b>	<b>Выводы</b>	<b>10</b>

## List of Tables

# List of Figures

# 1 Цель работы

изучать идеологию и применение контроля версий.

## 2 Задание

1. настройка git
2. подключение репозитория к github
3. Первичная конфигурация
4. конфигурация git-flow

## 3 Выполнение лабораторной работы

### 3.1 Настройка git

создал аккаунтан github

#### 2. Подключение репозитория r github

```
bdzhumaev@dk6n55 ~/.ssh/laboratory $ cd
bdzhumaev@dk6n55 ~ $ rm -r laboratory
bdzhumaev@dk6n55 ~ $ cd
bdzhumaev@dk6n55 ~ $ ssh-keygen -C "Дхумаев Бера <1032204350>"
Generating public/private rsa key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/b/d/bdzhumaev/.ssh/id_rsa):
/afs/.dk.sci.pfu.edu.ru/home/b/d/bdzhumaev/.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 /afs/.dk.sci.pfu.edu.ru/home/b/d/bdzhumaev/.ssh/id_rsa
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/b/d/bdzhumaev/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:1tQ9bn31SWqLem6kt0ZX40E7KA1XkYJrYLn1wHXjdmо Дхумаев Бера <1032204350>
The key's randomart image is:
+----[RSA 3072]-----+
|      o o . . o |
|      = B o o o |
|      o = X + + + |
|      . o = * + B = |
|      S . . + o E . * |
|      . + o o . o |
|      + . . . |
|      . . = |
|      . * o . |
+----[SHA256]-----+
```

Создал рабочий laboratory b репозиторий OSLAB. Добавил файл README.md и к нему добавил commit “first commit” после чего отправил на гитхаб.

#### 3. Первичная конфигурации

```
bdzhumae@dk6n55 ~/laboratory $ curl -L -s https://www.gitignore.io/api/c >> .gitignore
bdzhumae@dk6n55 ~/laboratory $ git add
bdzhumae@dk6n55 ~/laboratory $ git status
На ветке master
Ваша ветка обновлена в соответствии с «origin/master».

Изменения, которые будут включены в коммит:
  (use "git restore --staged <файл>..." to unstage)
        new file:   .gitignore
        new file:   LICENSE

bdzhumae@dk6n55 ~/laboratory $ git commit -am ""
Омена коммита из-за пустого сообщения коммита.
bdzhumae@dk6n55 ~/laboratory $ git commit -am ""
>
Омена коммита из-за пустого сообщения коммита.
bdzhumae@dk6n55 ~/laboratory $ git push
Warning: Permanently added the RSA host key for IP address '149.82.121.4' to the list of known hosts.
Everything up-to-date
bdzhumae@dk6n55 ~/laboratory $ git commit -am "second"
[master fd32e8c] second
 2 files changed, 455 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 LICENSE
bdzhumae@dk6n55 ~/laboratory $ git push
Перечисленные объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
Поскаты изменений выполняются по 6 потоков
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (4/4), 6.43 KiB | 6.43 MiB/s, готово.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Begench747jmaev/labos.git
 fd3272c..fd32e8c master -> master
bdzhumae@dk6n55 ~/laboratory $
bdzhumae@dk6n55 ~/laboratory $
bdzhumae@dk6n55 ~/laboratory $
bdzhumae@dk6n55 ~/laboratory $ git flow init

Which branch should be used for bringing forth production releases?
- master
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? [] v
Hooks and filters directory? [/afs/.dk.sci.pfu.edu.ru/home/b/d/bdzhumae/laboratory/.git/hooks]

bdzhumae@dk6n55 ~/laboratory $ git branch
bdzhumae@dk6n55 ~/laboratory $ git flow release start 1.0.0
Переключено на новую ветку «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'

bdzhumae@dk6n55 ~/laboratory $ echo "1.0.0" >> VERSION
bdzhumae@dk6n55 ~/laboratory $ git add .
bdzhumae@dk6n55 ~/laboratory $ git commit -am'chore(main): add version
> '
[release/1.0.0 cd76549] chore(main): add version
 1 file changed, 1 insertion(+)
 create mode 100644 VERSION
bdzhumae@dk6n55 ~/laboratory $ git flow release finish 1.0.0
Переключено на ветку «master»
Ваша ветка обновлена в соответствии с «origin/master».
Merge made by the 'recursive' strategy.
  VERSION | 1 +
  1 file changed, 1 insertion(+)
 create mode 100644 VERSION
Уже на «master»
```

## Добавил файл лицензии 4. Конфигуриция git-flow

```
bdzhumae@dk6n55 ~/laboratory $ git flow init

Which branch should be used for bringing forth production releases?
- master
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? [] v
Hooks and filters directory? [/afs/.dk.sci.pfu.edu.ru/home/b/d/bdzhumae/laboratory/.git/hooks]

bdzhumae@dk6n55 ~/laboratory $ git branch
bdzhumae@dk6n55 ~/laboratory $ git flow release start 1.0.0
Переключено на новую ветку «release/1.0.0»

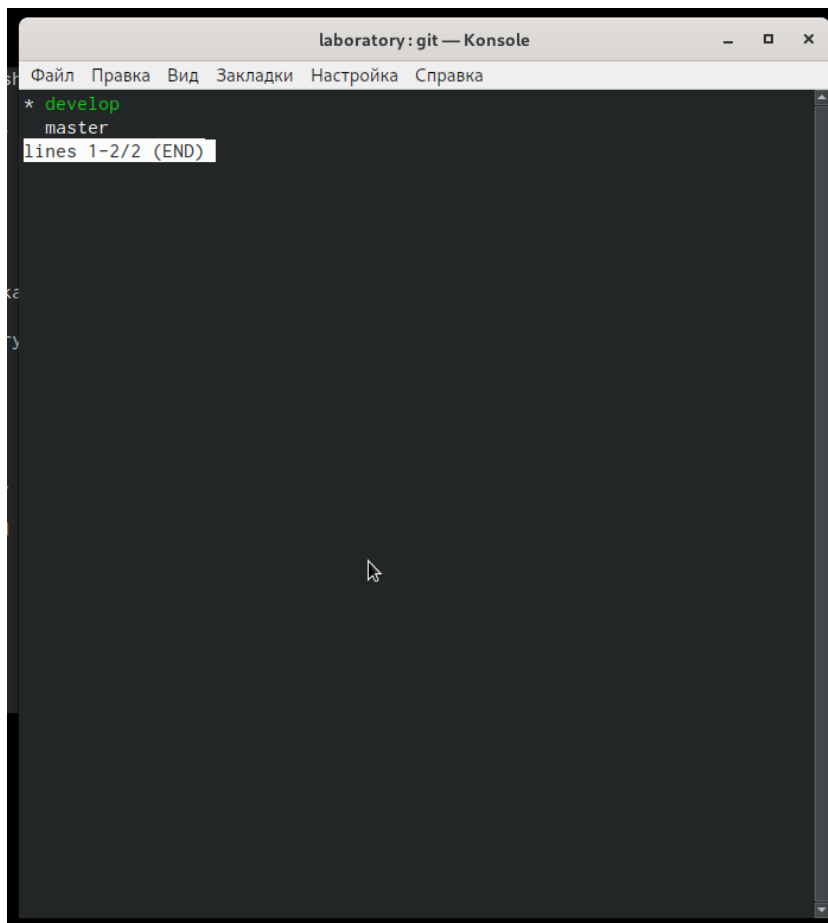
Summary of actions:
- A new branch 'release/1.0.0' was created, based on 'develop'
- You are now on branch 'release/1.0.0'

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'

bdzhumae@dk6n55 ~/laboratory $ echo "1.0.0" >> VERSION
bdzhumae@dk6n55 ~/laboratory $ git add .
bdzhumae@dk6n55 ~/laboratory $ git commit -am'chore(main): add version
> '
[release/1.0.0 cd76549] chore(main): add version
 1 file changed, 1 insertion(+)
 create mode 100644 VERSION
bdzhumae@dk6n55 ~/laboratory $ git flow release finish 1.0.0
Переключено на ветку «master»
Ваша ветка обновлена в соответствии с «origin/master».
Merge made by the 'recursive' strategy.
  VERSION | 1 +
  1 file changed, 1 insertion(+)
 create mode 100644 VERSION
Уже на «master»
```





Инициализировал git flow проверил что нахожусь на ветке develop создал релиз версии 1.0.0 и записал туда релиз Добавил релизную ветку в основную и отправил данные на гитхаб.

## 4 Выводы

научился применять средства котроля версий.