

# Отчёт о выполнении

## Лабораторная работа № 4

---

Просина К. М.

08 марта 2024

Российский университет дружбы народов, Москва, Россия

# Информация

---

- Просина Ксения Максимовна
- студентка Российского университета дружбы народов

# **Вводная часть**

---

- Целью лабораторной работы является получение практических навыков правильной работы с репозиториями git.

# Выполнение лабораторной работы

---

# Выполнение лабораторной работы

Для начала было необходимо скачать git-flow используя команды: Enable the copr repository

`dnf copr enable elegos/gitflow`

Install gitflow

`dnf install gitflow`

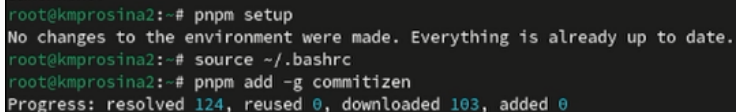
```
kmprosina@kmprosina2:~$ # Enable the copr repository
kmprosina@kmprosina2:~$ dnf copr enable elegos/gitflow
Ошибка: Эта команда должна быть выполнена от имени пользователя root.
kmprosina@kmprosina2:~$ sudo dnf copr enable elegos/gitflow
[sudo] пароль для kmprosina:
Включение репозитория Copr. Обратите внимание, что этот репозиторий
не является частью основного дистрибутива, и качество может отличаться.

Проект Fedora не имеет какого-либо влияния на содержимое этого
репозитория за рамками правил, описанных в Вопросах и Ответах Copr в
<https://docs.pagure.org/copr.copr/user\_documentation.html#what-i-can-build-in-copr
>,
а качество и безопасность пакетов не поддерживаются на каком-либо уровне.

Не отправляйте сообщения об ошибках этих пакетов в Fedora
Bugzilla. В случае возникновения проблем обращайтесь к владельцу этого репозитория.

Do you really want to enable copr.fedorainfracloud.org/elegos/gitflow? [y/N]: y
Репозиторий успешно подключен.
```

Далее идет установка и настройка Node.js

A terminal window with a dark background and light green text. The prompt is 'root@kmprosina2:~#'. The first command is 'pnpm setup', followed by the output 'No changes to the environment were made. Everything is already up to date.' The second command is 'source ~/.bashrc'. The third command is 'pnpm add -g commitizen', followed by the output 'Progress: resolved 124, reused 0, downloaded 103, added 0'.

```
root@kmprosina2:~# pnpm setup
No changes to the environment were made. Everything is already up to date.
root@kmprosina2:~# source ~/.bashrc
root@kmprosina2:~# pnpm add -g commitizen
Progress: resolved 124, reused 0, downloaded 103, added 0
```

Рис. 2: Настройка нод



Используем `pnpm add -g commitizen` для помощи в форматировании коммитов. При этом устанавливается скрипт `git-cz`, который мы и будем использовать для коммитов.

```
root@kmprosina2:~# pnpm setup
No changes to the environment were made. Everything is already up to date.
root@kmprosina2:~# source ~/.bashrc
root@kmprosina2:~# pnpm add -g commitizen
Progress: resolved 78, reused 0, downloaded 13, added 0
```

**Рис. 3:** Форматирование коммитов

## Выполнение лабораторной работы

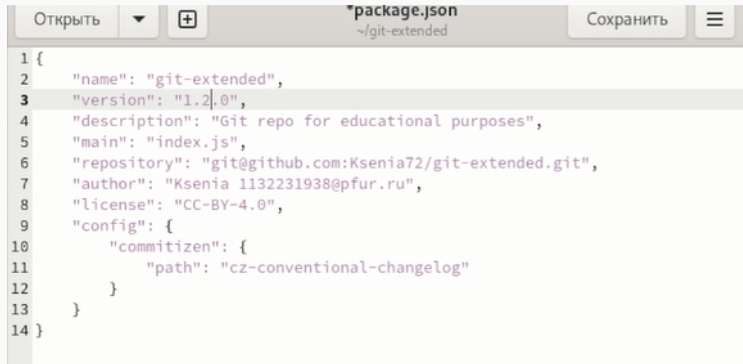
Подключение репозитория к github. Создали репозиторий на GitHubc названием git-extended. Делаем первый коммит и выкладываем на github.

```
kmprosina@kmprosina2:~$ git config --global user.name "Ksenia"
kmprosina@kmprosina2:~$ git commit -m "first commit"
[master (корневой коммит) fc56a29] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md
kmprosina@kmprosina2:~$ git branch -M main
kmprosina@kmprosina2:~$ git remote add origin git@github.com:Ksenia72/git-extended.git
kmprosina@kmprosina2:~$ git push -u origin main
git@github.com: Permission denied (publickey).
fatal: Не удалось прочесть из внешнего репозитория.

Удостоверьтесь, что у вас есть необходимые права доступа
и репозиторий существует.
kmprosina@kmprosina2:~$ sudo git push -u origin main
[sudo] пароль для kmprosina:
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 226 байтов | 17.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:Ksenia72/git-extended.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

# Выполнение лабораторной работы

Заполняем файл package.json согласно лабораторной работе.



The image shows a code editor window with the title bar "package.json" and the file path "~/git-extended". The editor contains the following JSON content:

```
1 {  
2   "name": "git-extended",  
3   "version": "1.2|.0",  
4   "description": "Git repo for educational purposes",  
5   "main": "index.js",  
6   "repository": "git@github.com:Ksenia72/git-extended.git",  
7   "author": "Ksenia 1132231938@pfur.ru",  
8   "license": "CC-BY-4.0",  
9   "config": {  
10     "commitizen": {  
11       "path": "cz-conventional-changelog"  
12     }  
13 }  
14 }
```

**Рис. 5:** package.json

# Выполнение лабораторной работы

Добавляем новые файлы .

```
kmprosina@kmprosina2:~/git-extended$ git add .
kmprosina@kmprosina2:~/git-extended$ git cz
cz-cli@4.3.0, cz-conventional-changelog@3.3.0

? Select the type of change that you're committing: feat:      A new feature
? What is the scope of this change (e.g. component or file name): (press enter to skip) readme.md
? Write a short, imperative tense description of the change (max 83 chars):
  (10) added file
? Provide a longer description of the change: (press enter to skip)

? Are there any breaking changes? No
? Does this change affect any open issues? No
[main 5a1676f] feat(readme.md): added file
 1 file changed, 14 insertions(+)
 create mode 100644 package.json
kmprosina@kmprosina2:~/git-extended$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 1.15 КиБ | 1.15 МБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:Ksenia72/git-extended.git
 49a1ca7..5a1676f  main -> main
kmprosina@kmprosina2:~/git-extended$ git flow init
```

## Выполнение лабораторной работы

Инициализируем git-flow и устанавливаем префикс для ярлыков v. Проверяем, что мы на ветке develop и загружаем весь репозиторий в хранилище.

```
kmprosina@kmprosina2:~/git-extended$ git flow init -f

Which branch should be used for bringing forth production releases?
  - develop
  - main
Branch name for production releases: [main]

Which branch should be used for integration of the "next release"?
  - develop
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? [/home/kmprosina/git-extended/.git/hooks]
kmprosina@kmprosina2:~/git-extended$ git branch
* develop
  main
```

## Выполнение лабораторной работы

Устанавливаем внешнюю ветку как вышестоящую для этой ветки. Создаем релиз с версией 1.0.0 и журнал изменений. Добавляем журнал изменений в индекс.

```
kmprosina@kmprosina2:~/git-extended$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
kmprosina@kmprosina2:~/git-extended$ git flow release start 1.0.0
Переключились 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'

kmprosina@kmprosina2:~/git-extended$ standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
kmprosina@kmprosina2:~/git-extended$ git add CHANGELOG.md
kmprosina@kmprosina2:~/git-extended$ git commit -am 'chore(site): add changelog'
```

# Выполнение лабораторной работы

Заливаем релизную ветку в основную ветку и отправляем данные на github.

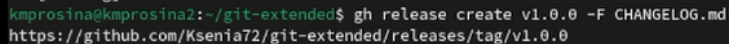
```
kmprosina@kmprosina2:~/git-extended$ git flow release finish 1.0.0
Branches 'main' and 'origin/main' have diverged.
And local branch 'main' is ahead of 'origin/main'.
Уже на «main»
Ваша ветка опережает «origin/main» на 2 коммита.
(используйте «git push», чтобы опубликовать ваши локальные коммиты)
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Merge made by the 'ort' strategy.
 CHANGELOG.md | 10 ++++++++
 1 file changed, 10 insertions(+)
 create mode 100644 CHANGELOG.md
Ветка release/1.0.0 удалена (была 3b3fe22).

Summary of actions:
- Release branch 'release/1.0.0' has been merged into 'main'
- The release was tagged 'v1.0.0'
- Release tag 'v1.0.0' has been back-merged into 'develop'
- Release branch 'release/1.0.0' has been locally deleted
- You are now on branch 'develop'

kmprosina@kmprosina2:~/git-extended$ git push --all
```

```
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (5/5), 2.84 КиБ | 364.00 КиБ/с, готово.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
```

Создаем релиз на github.

A terminal window showing a command to create a GitHub release. The prompt is 'kmprosina@kmprosina2:~/git-extended\$'. The command is 'gh release create v1.0.0 -F CHANGELOG.md'. The output is 'https://github.com/Ksenia72/git-extended/releases/tag/v1.0.0'.

```
kmprosina@kmprosina2:~/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/Ksenia72/git-extended/releases/tag/v1.0.0
```

Рис. 10: Релиз



## Выполнение лабораторной работы

Разработка новой функциональности Создаем ветку для новой функциональности и продолжаем работу с git как обычно. Объединяем ветку feature\_branch с develop.

```
kmprosina@kmprosina2:~/git-extended$ git flow feature start feature_branch
Переключились на новую ветку «feature/feature_branch»

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

Now, start committing on your feature. When done, use:

    git flow feature finish feature_branch

kmprosina@kmprosina2:~/git-extended$ git flow feature finish feature_branch
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Уже актуально.
Ветка feature/feature_branch удалена (была 277d30a).

Summary of actions:
- The feature branch 'feature/feature_branch' was merged into 'develop'
- Feature branch 'feature/feature_branch' has been locally deleted
- You are now on branch 'develop'
```

# Выполнение лабораторной работы

Создаем релиз с версией 1.2.3 и обновляем номер версии в файле package.json и журнал изменений. После этого добавляем журнал изменений в индекс

```
kmprosina@kmprosina2:~/git-extended$ git flow release start 1.2.3
Переключились на новую ветку «release/1.2.3»

Summary of actions:
- A new branch 'release/1.2.3' was created, based on 'develop'
- You are now on branch 'release/1.2.3'

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

    git flow release finish '1.2.3'

kmprosina@kmprosina2:~/git-extended$ standard-changelog
✓ output changes to CHANGELOG.md
kmprosina@kmprosina2:~/git-extended$ git add CHANGELOG.md
kmprosina@kmprosina2:~/git-extended$ git commit -am 'chore(site): update changelog'
Текущая ветка: release/1.2.3
нечего коммитить, нет изменений в рабочем каталоге
```

## Выполнение лабораторной работы

Заливаем релизную ветку в основную ветку и отправляем данные на github.  
Далее создаем релиз на github с комментарием из журнала изменений

```
kmprosina@kmprosina2:~/git-extended$ git flow release finish 1.2.3
Переключились на ветку «main»
Эта ветка соответствует «origin/main».
Merge made by the 'ort' strategy.
Уже на «main»
Ваша ветка опережает «origin/main» на 2 коммита.
  (используйте «git push», чтобы опубликовать ваши локальные коммиты)
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Merge made by the 'ort' strategy.
Ветка release/1.2.3 удалена (была 277d30a).

Summary of actions:
- Release branch 'release/1.2.3' has been merged into 'main'
- The release was tagged 'v1.2.3'
- Release tag 'v1.2.3' has been back-merged into 'develop'
- Release branch 'release/1.2.3' has been locally deleted
- You are now on branch 'develop'

kmprosina@kmprosina2:~/git-extended$ gedit package.json
kmprosina@kmprosina2:~/git-extended$ git push --all
Перечисление объектов: 2, готово.
Подсчет объектов: 100% (2/2), готово.
При сжатии изменений используется до 2 потоков
```

- Во время выполнения лабораторной работы мне удалось получить навыки правильной работы с репозиториями git.