

# **Отчет по лабораторной работе № 4**

**Продвинутое использование git**

Алиса Алексеевна Скобеева

# 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>
<b>4</b>	<b>Выводы</b>	<b>18</b>

# Список иллюстраций

3.1	Выполняем команду . . . . .	7
3.2	Выполняем следующую команду . . . . .	7
3.3	Ввод команд . . . . .	8
3.4	Выполняем <code>pnpm setup</code> . . . . .	8
3.5	Установка коммита <code>commitizen</code> . . . . .	8
3.6	Установка коммита <code>standart-changelog</code> . . . . .	8
3.7	Создаем репозиторий на GitHub . . . . .	9
3.8	Последовательно выполняем все действия . . . . .	9
3.9	Добавление коммита 1 . . . . .	10
3.10	Добавление коммита 2 . . . . .	10
3.11	Выполняем команду . . . . .	10
3.12	Редактируем файл . . . . .	11
3.13	Выполняем команды . . . . .	11
3.14	Выполняем <code>git push</code> для отправки на GitHub . . . . .	12
3.15	Выбор нужных параметров . . . . .	12
3.16	Выполняем команду . . . . .	12
3.17	Выполняем команду . . . . .	13
3.18	Выполняем команды . . . . .	13
3.19	Выполняем команду . . . . .	13
3.20	Выполняем команду . . . . .	14
3.21	Выполняем команду . . . . .	14
3.22	Выполняем команду . . . . .	14
3.23	Выполняем команду . . . . .	14
3.24	Выполняем команду . . . . .	15
3.25	Выполняем команду . . . . .	15
3.26	Последовательно выполняем все команды 1 . . . . .	15
3.27	Последовательно выполняем все команды 2 . . . . .	16
3.28	Последовательно выполняем все команды 3 . . . . .	16
3.29	Последовательно выполняем все команды 4 . . . . .	16
3.30	Последовательно выполняем все команды 5 . . . . .	17
3.31	Последовательно выполняем все команды 6 . . . . .	17
3.32	Последовательно выполняем все команды 7 . . . . .	17

## **Список таблиц**

# 1 Цель работы

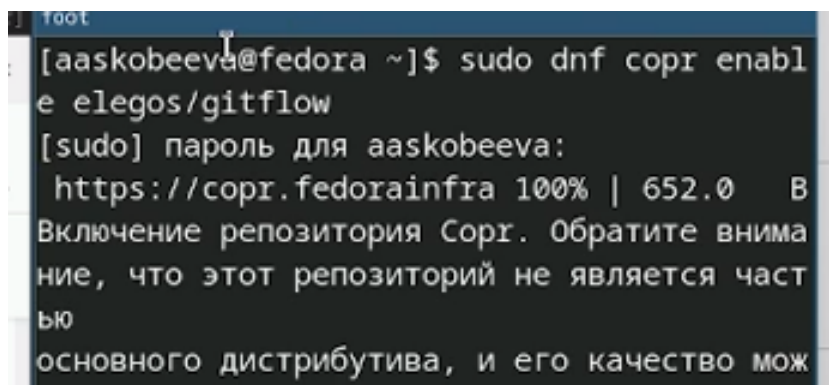
Получение навыков правильной работы с репозиториями git.

## 2 Задание

Создать репозиторий и выполнить с ним команды, которые указаны на платформе.

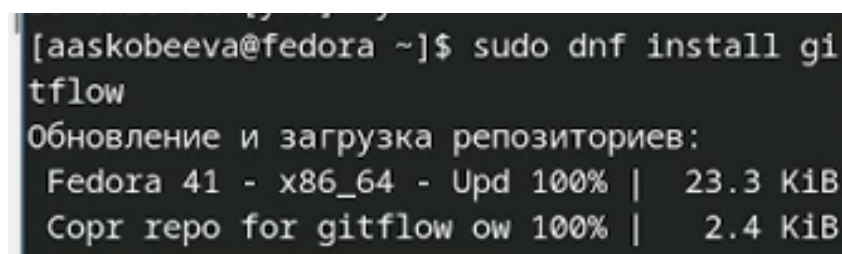
### 3 Выполнение лабораторной работы

Установка git-flow Устанавливаем git-flow из коллекции репозитория Copr:



```
[aaskobeeva@fedora ~]$ sudo dnf copr enable elegos/gitflow
[sudo] пароль для aaskobeeva:
https://copr.fedorainfra 100% | 652.0 В
Включение репозитория Copr. Обратите внима
ние, что этот репозиторий не является част
ью
основного дистрибутива, и его качество мож
```

Рис. 3.1: Выполняем команду



```
[aaskobeeva@fedora ~]$ sudo dnf install gitflow
Обновление и загрузка репозитория:
Fedora 41 - x86_64 - Upd 100% | 23.3 KiB
Copr repo for gitflow ow 100% | 2.4 KiB
```

Рис. 3.2: Выполняем следующую команду

Установка Node.js Последовательно выполняем команды:

```
[aaskobeeva@fedora git-extended]$ pnpm init
Wrote tc /home/aaskobeeva/git-extended/package.json

{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
```

Рис. 3.3: Ввод команд

Запускаем:

```
[aaskobeeva@fedora ~]$ pnpm setup
Appended new lines to /home/aaskobeeva/.bashrc

Next configuration changes were made:
export PNPM_HOME="/home/aaskobeeva/.local/share/pnpm"
```

Рис. 3.4: Выполняем pnpm setup

Общепринятые коммиты Добавляем коммиты commitizen и standart-changelog

```
[aaskobeeva@fedora ~]$ pnpm add -g commitizen
Progress: resolved 1, reused 0, downloaded 0, added 0
```

Рис. 3.5: Установка коммита commitizen

```
[aaskobeeva@fedora ~]$ pnpm add -g standart-changelog
Progress: resolved 0, reused 1, downloaded 0
Progress: resolved 3, reused 2, downloaded 0
Progress: resolved 5, reused 4, downloaded 0
Progress: resolved 7, reused 6, downloaded 0
```

Рис. 3.6: Установка коммита standart-changelog



Создание репозитория git

Создаем репозиторий git. Называем его git-extended.

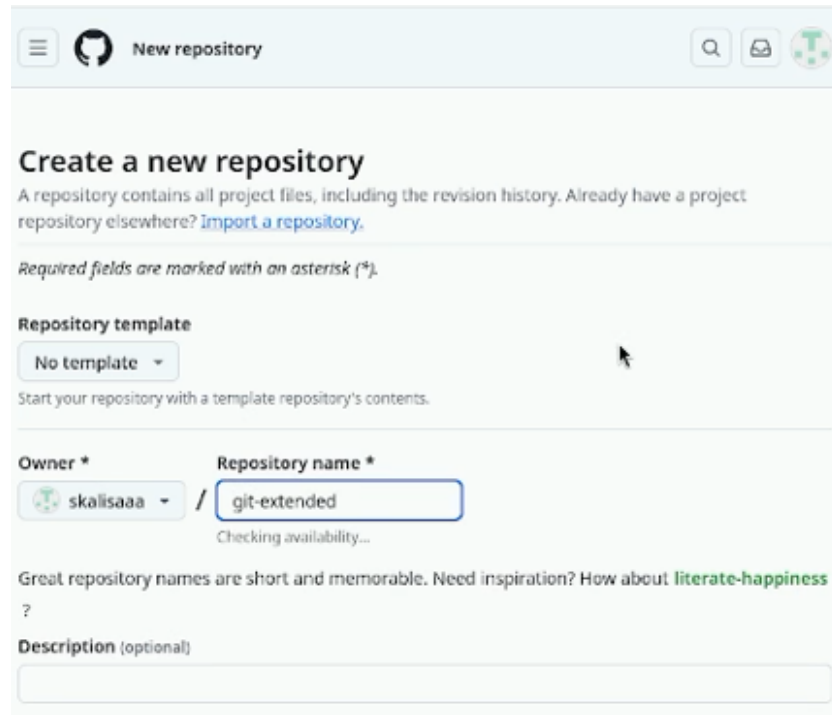


Рис. 3.7: Создаем репозиторий на GitHub

Добавляем какой-нибудь файл, чтобы репозиторий корректно работал. Назовем его TOUCHME. Откроем его, и напишем произвольный текст. После, можем создавать первый коммит.

```
[aaskobeeva@fedora git-extended]$ touch README.md
[aaskobeeva@fedora git-extended]$ nano README.md
[aaskobeeva@fedora git-extended]$ git add .
[aaskobeeva@fedora git-extended]$ git commit -m "first commit"
```

Рис. 3.8: Последовательно выполняем все действия

Добавляем коммит:

```
[aaskobeeva@fedora git-extended]$ git remote add origin
git@github.com:git@github.com:skalisaaa/git-extended.git
/git-extended.git
error: внешний репозиторий origin уже существует
```

Рис. 3.9: Добавление коммита 1

```
[aaskobeeva@fedora git-extended]$ git push -u origin mai
n
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 870 байтов | 870.00 КиБ/с,
готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (fr
```

Рис. 3.10: Добавление коммита 2

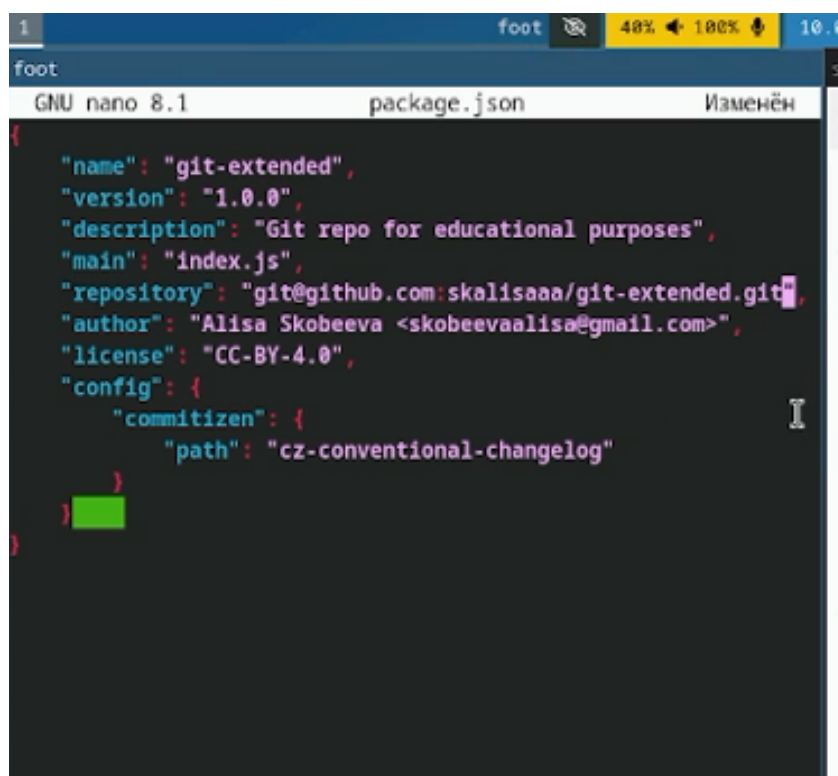
Конфигурация общепринятых коммитов Выполняем команду `pnpm init`:

```
[aaskobeeva@fedora git-extended]$ pnpm init
Wrote tc /home/aaskobeeva/git-extended/package.json

{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
```

Рис. 3.11: Выполняем команду

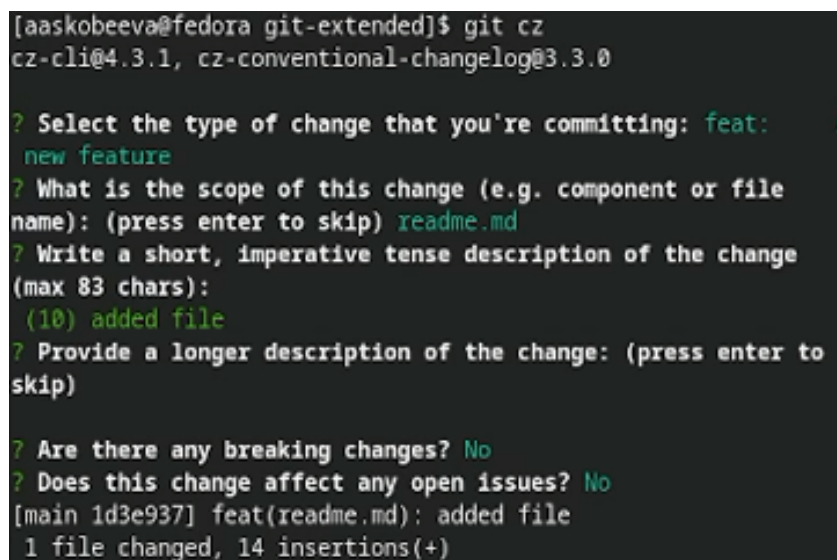
Последовательно заполняем файл так, как от нас требуется:



```
1 foot 48% 100% 10.0
foot
GNU nano 8.1 package.json Изменён
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:skalisaaa/git-extended.git",
  "author": "Alisa Skobeeva <skobeevaalisa@gmail.com>",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

Рис. 3.12: Редактируем файл

Добавляем новые файлы с помощью `git add`. Далее выполняем коммит `git cz` и отправляем все на GitHub:



```
[aaskobeeva@fedora git-extended]$ git cz
cz-cli@4.3.1, cz-conventional-changelog@3.3.0

? Select the type of change that you're committing: feat:
  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 1d3e937] feat(readme.md): added file
1 file changed, 14 insertions(+)
```

Рис. 3.13: Выполняем команды

```
[aaskobeeva@fedora git-extended]$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (3/3), готово.
```

Рис. 3.14: Выполняем git push для отправки на GitHub

Конфигурация git-flow Инициализируем git-flow с помощью git flow init и выбираем нужные нам параметры:

```
[aaskobeeva@fedora git-extended]$ git flow init
Already initialized for gitflow.
To force reinitialization, use: git flow init -f
[aaskobeeva@fedora 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/aaskobeeva/git-extended/.git/hooks]
```

Рис. 3.15: Выбор нужных параметров

Проверяем, находимся ли мы на ветке develop и далее загружаем весь репозиторий в хранилище:

```
[aaskobeeva@fedora git-extended]$ git push --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:
```

Рис. 3.16: Выполняем команду

Устанавливаем внешнюю ветку как вышестоящую для этой ветки:

```
[aaskobeeva@fedora git-extended]$ git branch --set-upstream-to=origin/develop develop  
branch 'develop' set up to track 'origin/develop'.
```

Рис. 3.17: Выполняем команду

Далее, создаем релиз с версией 1.0.0, создаем журнал изменений и добавляем его в индекс:

```
[aaskobeeva@fedora git-extended]$ git add CHANGELOG.md  
[aaskobeeva@fedora git-extended]$ git commit -am 'chore(site): add changelog'  
[release/1.0.0 508f69f] chore(site): add changelog  
1 file changed, 9 insertions(+)  
create mode 100644 CHANGELOG.md
```

Рис. 3.18: Выполняем команды

Заливаем релизную ветку в основную ветку и отправляем данные на github. Далее создаем релиз на github. Для этого используем утилиты работы с github:

```
[aaskobeeva@fedora git-extended]$ git flow release finish 1.0.0  
Переключились на ветку «main»  
Эта ветка соответствует «origin/main».  
подсказка: Ожидание, пока вы закроете редактор с файлом.  
Merge made by the 'ort' strategy.  
CHANGELOG.md | 9 +++++++  
1 file changed, 9 insertions(+)  
create mode 100644 CHANGELOG.md  
Уже на «main»  
Ваша ветка опережает «origin/main» на 2 коммита.  
(используйте «git push», чтобы опубликовать ваши локальные коммиты)  
подсказка: Ожидание, пока вы закроете редактор с файлом.
```

Рис. 3.19: Выполняем команду

```
Merge branch 'release/1.0.0'
# Пожалуйста, введите сообщение коммита, для объяснения>
# это слияние, особенно, если это слияние обновленной в>
# ветки в тематическую ветку.
#
# Lines starting with '#' will be ignored, and an empty>
# the commit.
```

Рис. 3.20: Выполняем команду

```
[aaskobeeva@fedora git-extended]$ git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (5/5), готово.
```

Рис. 3.21: Выполняем команду

```
[aaskobeeva@fedora git-extended]$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 154 байта | 22.00 КиБ/с, го
ТОВО.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (fr
om 0)
```

Рис. 3.22: Выполняем команду

```
[aaskobeeva@fedora git-extended]$ gh release create v1.0
-F CHANGELOG.md
https://github.com/skalisaaa/git-extended/releases/tag/v
1.0.0
```

Рис. 3.23: Выполняем команду

Работа с репозиторием git. Разработка новой функциональности Создаем ветку для новой функциональности, далее, продолжаем работу с git как обычно, и по окончании объединяем ветку feature\_branch с develop:



```

1.0.0
[aaskobeeva@fedora 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, bas
ed on 'develop'

```

Рис. 3.24: Выполняем команду

```

[aaskobeeva@fedora git-extended]$ git flow feature finis
h feature_branch
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Уже актуально.
Ветка feature/feature_branch удалена (была 1e12bf5).

```

Рис. 3.25: Выполняем команду

Создание релиза git-flow Создаем релиз с версией 1.2.3. Обновляем номер версии в файле package.json. Далее, создаем журнал изменений и добавляем его в индекс. Далее, заливаем релизную ветку в основную ветку и отправляем данные на GitHub. Создаем релиз на github с комментарием из журнала изменений:

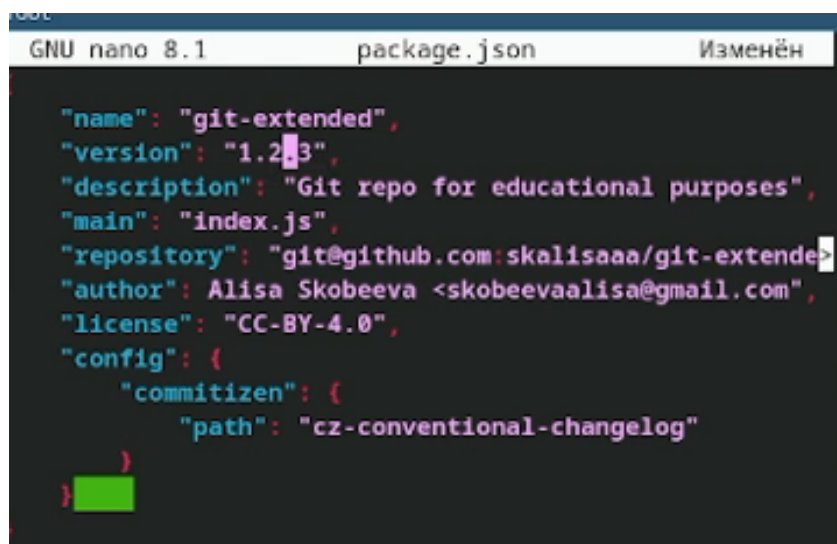
```

[aaskobeeva@fedora 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 'de
velop'
- You are now on branch 'release/1.2.3'

```

Рис. 3.26: Последовательно выполняем все команды 1

A screenshot of a terminal window showing the nano text editor editing a file named package.json. The editor's title bar shows 'GNU nano 8.1' and 'package.json' with a status 'Изменён'. The JSON content is as follows:

```
"name": "git-extended",
"version": "1.2.3",
"description": "Git repo for educational purposes",
"main": "index.js",
"repository": "git@github.com:skalisaaa/git-extended",
"author": "Alisa Skobeeva <skobeevaalisa@gmail.com>",
"license": "CC-BY-4.0",
"config": {
  "commitizen": {
    "path": "cz-conventional-changelog"
  }
}
```

Рис. 3.27: Последовательно выполняем все команды 2

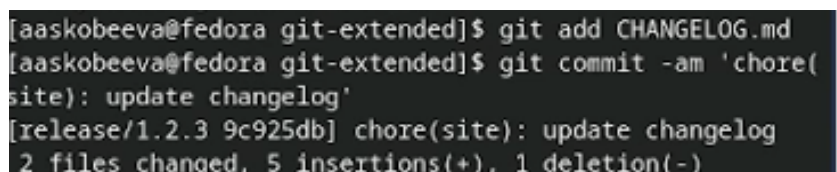
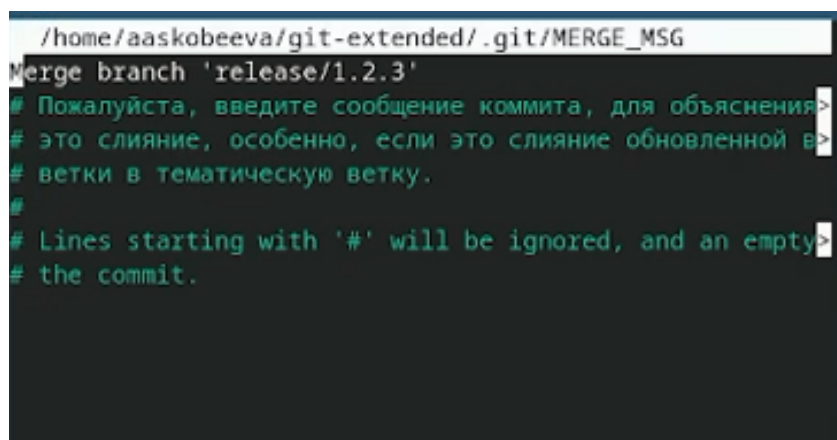
A screenshot of a terminal window showing a series of git commands and their output. The commands are: 'git add CHANGELOG.md', 'git commit -am \'chore(site): update changelog\'', and the output of 'git release' showing a new release '1.2.3' with commit hash '9c925db' and a message 'chore(site): update changelog'. It also shows '2 files changed, 5 insertions(+), 1 deletion(-)'.

Рис. 3.28: Последовательно выполняем все команды 3

A screenshot of a terminal window showing the content of a file named .git/MERGE\_MSG. The file contains a merge message template for merging the 'release/1.2.3' branch. The text is as follows:

```
Merge branch 'release/1.2.3'
# Пожалуйста, введите сообщение коммита, для объяснения
# это слияние, особенно, если это слияние обновленной ветки
# ветки в тематическую ветку.
#
# Lines starting with '#' will be ignored, and an empty
# the commit.
```

Рис. 3.29: Последовательно выполняем все команды 4



```
[aaskobeeva@fedora git-extended]$ git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 2.76 КиБ | 942.00 КиБ/с, го
тово.
Total 6 (delta 2), reused 0 (delta 0), pack-reused 0 (fr
```

Рис. 3.30: Последовательно выполняем все команды 5

```
[aaskobeeva@fedora git-extended]$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 153 байта | 153.00 КиБ/с, г
отово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (fr
om 0)
To github.com:skalisaaa/git-extended.git
```

Рис. 3.31: Последовательно выполняем все команды 6

```
[aaskobeeva@fedora git-extended]$ gh release create v1.2
.3 -F CHANGELOG.md
https://github.com/skalisaaa/git-extended/releases/tag/v
1.2.3
[aaskobeeva@fedora git-extended]$
```

Рис. 3.32: Последовательно выполняем все команды 7

## 4 Выводы

В ходе выполнения лабораторной работы мы приобрели навыки продвинутого использования git.