

# Презентация по лабораторной работе № 4

Продвинутое использование git

---

Неустроева И.Н.

29 февраля 2024

Российский университет дружбы народов, Москва, Россия

# Информация

---

- Неустроева Ирина Николаевна
- студентка группы НБИ 02-23
- Российский университет дружбы народов
- <https://inneustroeva.github.io/ru/>

# **Вводная часть**

---

Получение навыков правильной работы с репозиториями git.

- Процессор pandoc для входного формата Markdown
- Результирующие форматы
  - pdf
  - html
- Автоматизация процесса создания: Makefile

## Основная часть

---





# Установка программного обеспечения

## Установливаем git-flow

```
inneustroeva@inneustroeva:~$ sudo -i  
root@inneustroeva:~# dnf copr enable elegos/gitflow
```

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](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.
```

```
root@inneustroeva:~# dnf install gitflow
```

Copr repo for gitflow owned by elegos

Fedora 39 - x86\_64 - Updates

[ ===

## Добавление общепринятых коммитов.

При помощи утилиты `pnpm` добавляем две программы: `standard changelog` и `commitizen`.

```
root@inneustroeveva:~# pnpm add -g commitizen
```

```
Update available! 8.12.0 → 8.15.4.  
Changelog: https://github.com/pnpm/pnpm/releases/tag/v8.15.4  
Run "pnpm add -g pnpm" to update.
```

```
Follow @pnpmjs for updates: https://twitter.com/pnpmjs
```

```
Packages: +152
```



```
Downloading registry.npmjs.org/typescript/5.4.2: 5,82 MB/5,82 MB, done
```

```
Progress: resolved 152, reused 0, downloaded 152, added 152, done
```

```
/root/.local/share/pnpm/global/5:
```

```
+ commitizen 4.3.0
```

## Конфигурация общепринятых коммитов.

Введем команду `pnpm init`. Вывод команды демонстрирует содержание файла и его место в системе.

```
branch main set up to track new-origin/main .
inneustroeva@inneustroeva:~/git-extended$ pnpm init
Wrote to /home/inneustroeva/git-extended/package.json

{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": ""
}
```

# Конфигурация git-flow.

Инициализируем git-flow и проверим, что мы находимся на нужной ветке (develop)

```
inneustroeva@inneustroeva:~/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? [hooks/]
Hooks and filters directory? [hooks/]
Hooks and filters directory? [hooks/]
```

## Разработка новой функциональности.

Создадим ветку для новой функциональности:

```
inneustroeva@inneustroeva:~/git-extended$ git flow feature start feature_branch
A      CHANGELOG.md
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
```

Далее работа с гит продолжается как обычно. Для тестового репозитория никакую новую функциональность разрабатывать не будем, переходим сразу к следующему шагу: объединение веток feature\_branch и develop

## Создание релиза.

Создадим релиз с версией 1.2.3 и оказываемся на ветке release/1.2.3:

```
inneustroeva@inneustroeva:~/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

Получены навыки правильной работы с репозиториями git, выполнена работа для тестового репозитория и дальнейшие преобразования для основного репозитория курса. :::