

# Лабораторная работа №2

## Настройка git

---

Газизянов Владислав Альбертович

Российский университет дружбы народов, Москва, Россия

# Цель

---

Изучение работы и назначения системы контроля версий git приобретение навыков по работе с ней.

# Выполнение лабораторной работы

---

# Установка необходимых программы

```
[sudo] пароль для vagazizyanov:  
[root@vagazizyanov ~]# dnf install git  
Fedora 39 - x86_64 - Updates      29 kB/s | 19 kB      00:00  
Fedora 39 - x86_64 - Updates      2.0 MB/s | 2.5 MB     00:01
```

Рис. 1: Установка git

A screenshot of a terminal window with a dark background. The prompt shows the user is 'Vladislav Gazizyan' in the directory '~'. The command entered is 'git config --global user.name "Vladislav Gazizyan"'.

```
Владислав Газизьянов  
[vlagazizyanov ~]# git config --global user.name "Vladislav Gazizyan"
```

**Рис. 2:** Ввод почты и имени пользователя

## Базовая настройка. Установка параметров

```
@vagazizyanov ~]# git config --global core.quotePath false
@vagazizyanov ~]# git config --global init.defaultBranch master
@vagazizyanov ~]# git config --global core.autocrlf input
@vagazizyanov ~]# git config --global core.safecrlf warn
@vagazizyanov ~]#
```

Рис. 3: Настройка параметров

# Создание ключей

```
. . o + . . |
. + + . o . |
* + .S .o.E |
% o . . . o |
* 0 .oo. . |
=.o 0 .+. |
.=o =. . . |
----[SHA256]-----+
root@vagazizyanov ~]# ssh-keygen -t ed25519
generating public/private ed25519 key pair.
Enter file in which to save the key (/root/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_ed25519
Your public key has been saved in /root/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:nhStG1SiUXZw5nmzVwtbKV6gMgzCXamyQRUSTz7cMGg root@vagazizyanov
The key's randomart image is:
--[ED25519 256]--+
  .=*%+* . . |
  E@.&.o . . . |
  o. *.0 = o = |
  o... + + B . |
  +. S . + . |
  . = . . |
  . o |
  |
----[SHA256]-----+
root@vagazizyanov ~]# gpg --full-generate-key
gpg (GnuPG) 2.4.3; Copyright (C) 2023 g10 Code GmbH
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: создан каталог '/root/.gnupg'
Выберите тип ключа:
  ) RSA and RSA
```



## GPG keys

[New GPG key](#)

This is a list of GPG keys associated with your account. Remove any keys that you do not recognize.

  
GPG

**Email address:** ony.resset@gmail.com Unverified  
**Key ID:** 19BFE4139E2765BB  
**Subkeys:** DDCB10B1CF11E404  
Added on Feb 26, 2024

Delete

**Рис. 5:** Добавление ключей по их отпечатку на github

```
zizyanov@vagazizyanov ~]$ git config --global user.signingkey ony.r@gmail.com
zizyanov@vagazizyanov ~]$ git config --global commit.gpgsign true
zizyanov@vagazizyanov ~]$ git config --global gpg.program ${which g
```

**Рис. 6:** Добавление автоматической подписи коммитов

# Создание и клонирование шаблона

```
[vagazizyanov@vagazizyanov ~]$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Generate a new SSH key to add to your GitHub account? Yes
? Enter a passphrase for your new SSH key (Optional)
? Select a provider for your SSH key: GitHub CLI
? Would you like to authenticate GitHub CLI? Login with a web browser
```

Рис. 7: Работа с шаблоном

```
[vagazizyanov@vagazizyanov Операционные системы]$ cd ~/work/study/2022-2023/"О
перационные системы"/os-intro
bash: cd: /home/vagazizyanov/work/study/2022-2023/Операционные системы/os-intro:
Нет такого файла или каталога
[vagazizyanov@vagazizyanov Операционные системы]$ cd ~/work/study/2023-2024/"О
перационные системы"/os-intro
[vagazizyanov@vagazizyanov os-intro]$ rm package.json
[vagazizyanov@vagazizyanov os-intro]$ echo os-intro > COURSE
[vagazizyanov@vagazizyanov os-intro]$ make
Usage:
  make <target>

Targets:
  List of courses
  Generate directory structure
  module
  Update submodules
```

Рис. 8: Создание структуры курса

# Сохранение изменений

```
.jpg
create mode 100644 project-personal/stage6/report/pandoc/csl/gost-r-7-0-5-8-numeric.csl
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_eq
.py
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_fi
s.py
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_se
s.py
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_ta
nos.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxno
_init__.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxno
ore.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxno
ain.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxno
andocattributes.py
create mode 100644 project-personal/stage6/report/report.md
[vagazizyanov@vagazizyanov os-intro]$ git push
Перечисление объектов: 40, готово.
Подсчет объектов: 100% (40/40), готово.
При сжатии изменений используется до 5 потоков
е объектов: 100% (30/30), готово.
ь объектов: 100% (38/38), 342.11 КиБ | 2.50 МБ/с, готово.
38 (изменений 4), повторно использовано 0 (изменений 0), повторно исп
о пакетов 0
e: Resolving deltas: 100% (4/4), completed with 1 local object.
thub.com:ONVX8880/study_2023-2024_os-intro.git
030d9..cf31b5b master -> master
zizyanov@vagazizyanov os-intro]$
```

## **Контрольные вопросы**

---

## Ответы на контрольные вопросы

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначены?
  - Они применяются при работе нескольких человек с одним проектом. При внесении изменений позволяют фиксировать, совмещать и возвращать изменения разных людей
2. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия.
  - Хранилище - место, где находятся данные(файлы, коды и тд)
  - Commit - команда, для сохранения изменений
  - История - информация о предыдущих изменениях
  - Рабочая копия - одна из версий проекта, с которой ведется работа(= текущая/основная)

3. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.
  - Централизованные системы предполагают наличие единого репозитория для хранения данных(CVC, Subversion)
  - В децентрализованных системах центральный репозиторий не обязателен(Git, Bazaar)
4. Опишите действия с VCS при единоличной работе с хранилищем. Работа происходит на своем компьютере, сначала обновляются данные, в конце они размещаются в центральном репозитории



5. Опишите порядок работы с общим хранилищем VCS. Для идентификации на сервере необходимы ключи и затем создание репозитория, только затем можно работать на локальной машине. Также в конце изменения добавляются на сервер
6. Каковы основные задачи, решаемые инструментальным средством git?
  - хранение информации о всех изменениях
  - обеспечение удобства командной работы

## Ответы на контрольные вопросы

7. Назовите и дайте краткую характеристику командам git.

- Создание основного дерева репозитория:

git init - Получение обновлений (изменений) текущего дерева из центрального репозитория:

git pull - Отправка всех произведённых изменений локального дерева в центральный репозиторий:

git push - Просмотр списка изменённых файлов в текущей директории:

git status - Просмотр текущих изменений:

git diff

## Ответы на контрольные вопросы

7.

- Сохранение текущих изменений:

добавить все изменённые и/или созданные файлы и/или каталоги:

`git add .` добавить конкретные изменённые и/или созданные файлы и/или каталоги:

`git rm имена_файлов` Сохранение добавленных изменений:

- сохранить все добавленные изменения и все изменённые файлы:

`git commit -am 'Описание коммита'` - сохранить добавленные изменения с внесением комментария через встроенный редактор:

`git checkout -b имя_ветки`

## Ответы на контрольные вопросы

8. Приведите примеры использования при работе с локальным и удалённым репозиториями. локальный репозиторий - работа со своими файлами удаленный репозиторий - совместная работа, общий проект
9. Что такое и зачем могут быть нужны ветви (branches)? Ветви это пути к отдельным состояниям (отделы проекта),они дают возможность вносить изменения только в часть проекта и не трогать все вышестоящее
10. Как и зачем можно игнорировать некоторые файлы при commit? Игнорировать файлы можно с помощью .gitignore. Нужно это, например, при наличии ненужных(лишних или созданных автоматически) файлов.

## Вывод

---

В ходе работы была освоена работа с системой контроля версий, был установлен git, проведена авторизация, заданы базовые настройки, создались ключи, клонировался репозиторий и так далее.