

Лабораторная работа №2

Управление версиями

Сунгурова Мариян Мухсиновна

Содержание

| | | |
|----------|---------------------------------------|-----------|
| 1 | Цель работы | 4 |
| 2 | Задание | 5 |
| 3 | Теоретическое введение | 6 |
| 4 | Выполнение лабораторной работы | 8 |
| 5 | Выводы | 14 |
| | Список литературы | 16 |

Список иллюстраций

| | | |
|------|--|----|
| 4.1 | Создание учетной записи | 8 |
| 4.2 | Процесс установки | 8 |
| 4.3 | Процесс установки | 9 |
| 4.4 | Настройка git | 9 |
| 4.5 | Создание ключа | 9 |
| 4.6 | Создание ключа | 10 |
| 4.7 | Создание ключей | 10 |
| 4.8 | Добавление ключа | 11 |
| 4.9 | Добавление ключа | 11 |
| 4.10 | Ключ в поле ввода | 11 |
| 4.11 | Настройка подписей коммитов | 11 |
| 4.12 | Настройка | 12 |
| 4.13 | Создание репозитория на основе шаблона | 12 |
| 4.14 | Создание репозитория | 12 |
| 4.15 | Настройка | 13 |
| 4.16 | Настройка | 13 |

1 Цель работы

Изучить идеологию и применение средств контроля версий.–Освоить умения по работе с git.

2 Задание

-Создать базовую конфигурацию для работы с git. –Создать ключSSH.–Создать ключPGP. –Настроить подписи git. –Зарегистрироваться наGithub. –Создать локальный каталог для выполнения заданий по предмету.

3 Теоретическое введение

Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. Обычно основное дерево проекта хранится в локальном или удалённом репозитории, к которому настроен доступ для участников проекта. При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат к любой более ранней версии проекта, если это требуется. В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. Участник проекта (пользователь) перед началом работы посредством определённых команд получает нужную ему версию файлов. После внесения изменений, пользователь размещает новую версию в хранилище. При этом предыдущие версии не удаляются из центрального хранилища и к ним можно вернуться в любой момент. Сервер может сохранять неполную версию изменённых файлов, а производить так называемую дельта-компрессию — сохранять только изменения между последовательными версиями, что позволяет уменьшить объём хранимых данных. Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию, отменить изменения вообще или заблокировать файлы для изменения. В зависимости от настроек блокировка

не позволяет другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю, работающему с файлом. Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности. Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Кроме того, обычно доступна информация о том, кто из участников, когда и какие изменения вносил. Обычно такого рода информация хранится в журнале изменений, доступ к которому можно ограничить. В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным. Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд.

4 Выполнение лабораторной работы

1. Создала учётную запись на <https://github.com>. Заполнила основные данные на <https://github.com>. (рис. 4.1)

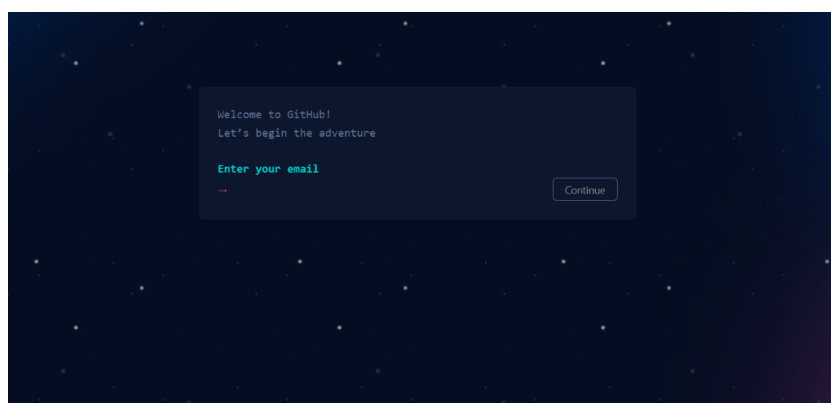


Рис. 4.1: Создание учетной записи

2. Установила git-flow в Fedora Linux (рис. 4.2)

```
mmsungurova@fedora: ~/work/study/2021-2022/Операционные системы/os-intro
[mmsungurova@fedora ~]$ cd /tmp
[mmsungurova@fedora tmp]$ wget --no-check-certificate -q https://raw.githubusercontent.com/petervanderdoes/gitflow/develop/contrib/gitflow-installer.sh
[mmsungurova@fedora tmp]$ chmod +x gitflow-installer.sh
[mmsungurova@fedora tmp]$ sudo ./gitflow-installer.sh install stable

Мы полагаем, что ваш системный администратор изложил вам основы
безопасности. Как правило, всё сводится к трем следующим правилам:

#1) Уважайте частную жизнь других.
#2) Думайте, прежде что-то вводить.
#3) С большой властью приходит большая ответственность.

[sudo] пароль для mmsungurova:
sudo: ./gitflow-installer.sh: command not found
[mmsungurova@fedora tmp]$ wget --no-check-certificate -q https://raw.githubusercontent.com/petervanderdoes/gitflow/develop/contrib/gitflow-installer.sh
[mmsungurova@fedora tmp]$ chmod +x gitflow-installer.sh
[mmsungurova@fedora tmp]$ sudo ./gitflow-installer.sh install stable
## git-flow no-make installer ##
Installing git-flow to /usr/local/bin
Cloning repo from GitHub to gitflow
Клонирование в «gitflow»...
remote: Enumerating objects: 4270, done.
remote: Total 4270 (delta 0), reused 0 (delta 0), pack-reused 4270
Получение объектов: 100% (4270/4270), 1.74 МБ | 1.21 МБ/с, готово.
Определение изменений: 100% (2533/2533), готово.
```

Рис. 4.2: Процесс установки

3. Установила gh в Fedora Linux (рис. 4.3)

```
[mmsungurova@fedora tmp]$ sudo dnf install gh
Fedora 35 - x86_64 - Updates                23 kB/s | 17 kB    00:00
Fedora 35 - x86_64 - Updates                722 kB/s | 3.2 MB  00:04
Fedora Modular 35 - x86_64 - Updates        38 kB/s | 18 kB    00:00
Зависимости разрешены.
=====
Пакет      Архитектура  Версия      Репозиторий  Размер
=====
Установка:
gh          x86_64       2.7.0-1.fc35 updates      6.8 М
=====
Результат транзакции
=====
Установка 1 Пакет

Объем загрузки: 6.8 М
Объем изменений: 32 М
Продолжить? [д/н]: д
Загрузка пакетов:
gh-2.7.0-1.fc35.x86_64.rpm                1.2 MB/s | 6.8 MB  00:05
=====
Общий размер                1.1 MB/s | 6.8 MB  00:06
Проверка транзакции
Проверка транзакции успешно завершена.
Идет проверка транзакции
Тест транзакции проведен успешно.
Выполнение транзакции
Подготовка      :
Установка       : gh-2.7.0-1.fc35.x86_64      1/1
Запуск скрипта : gh-2.7.0-1.fc35.x86_64      1/1
Проверка        : gh-2.7.0-1.fc35.x86_64      1/1
Установлен:
gh-2.7.0-1.fc35.x86_64
Выполнено!
```

Рис. 4.3: Процесс установки

4. Произвела базовую настройку git (рис. 4.4)

```
mmsungurova@fedora: ~/work/study/2021-2022/Операционные системы/os-intro
выполнено:
[mmsungurova@fedora tmp]$ git config --global user.name "Marian Sungurova"
[mmsungurova@fedora tmp]$ git config --global user.email "zessie94@mail.ru"
bash: git.config: command not found...
[mmsungurova@fedora tmp]$ git config --global user.email "zessie94@mail.ru"
[mmsungurova@fedora tmp]$ git config --global core.quotepatch false
[mmsungurova@fedora tmp]$ git config --global init.defaultBranch master
[mmsungurova@fedora tmp]$ git config --global core.autocrlf input
[mmsungurova@fedora tmp]$ git config --global core.safecrlf warn
git: «config» не является командой git. Смотрите «git --help».

Самые похожие команды:
  config
[mmsungurova@fedora tmp]$ git config --global core.safecrlf warn
```

Рис. 4.4: Настройка git

5. Создала ключ ssh по алгоритму rsa размером 4096 мб (рис. 4.5)

```
[mmsungurova@fedora tmp]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/mmsungurova/.ssh/id_rsa):
Created directory '/home/mmsungurova/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/mmsungurova/.ssh/id_rsa
Your public key has been saved in /home/mmsungurova/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:ASEF1EdZRRcDYm8ihSUS8Wxi1UdQUCW4WZ01fSM/M mmsungurova@fedora
The key's randomart image is:
+---[RSA 4096]-----+
|  ++0%X/++          |
|                    |
|                    |
|                    |
|                    |
|                    |
|                    |
|                    |
|                    |
|                    |
+---+-----+

```

Рис. 4.5: Создание ключа

6. Создала ключ по алгоритму ed25519 (рис. 4.6)

```
mmsungurova@fedora:~/work/study/2021-2022/Операционные системы/os-intro
[mmsungurova@fedora tmp]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/mmsungurova/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/mmsungurova/.ssh/id_ed25519
Your public key has been saved in /home/mmsungurova/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:bucC+oS0i0mSsLx0GijJ91jEkFulF5WpUwaZJyevW0g mmsungurova@fedora
The key's randomart image is:
+--[ED25519 256]--+
|      .o.. .+.o   |
|      o o  =.B    |
|      o   .X      |
|      o   .E      |
|      . o S . +   |
|+=+   .... o .   |
|B* o.o..o . o    |
|o+o+o= .o .     |
|oood.o.  ..     |
+---[SHA256]-----+
```

Рис. 4.6: Создание ключа

7. Создала ключи pgr. (рис. 4.7)

```
mmsungurova@fedora tmp]$ gpg --full-generate-key
gpg (GnuPG) 2.3.2; Copyright (C) 2021 Free Software Foundation, Inc.
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 '/home/mmsungurova/.gnupg'
gpg: создан шит с ключами '/home/mmsungurova/.gnupg/pubring.kbx'
Выберите тип ключа:
(1) RSA and RSA
(2) DSA and Elgamal
(3) DSA (sign only)
(4) RSA (sign only)
(9) ECC (sign and encrypt) «default»
(10) ECC (только для подписи)
(14) Existing key from card
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
0 = не ограничен
<n> = срок действия ключа - n дней
<nw> = срок действия ключа - n недель
<m> = срок действия ключа - n месяцев
<y> = срок действия ключа - n лет
Срок действия ключа? (0)
Срок действия ключа не ограничен
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.
Ваше полное имя: Marian
Адрес электронной почты: zessie94@mail.ru
Примечание:
Вы выбрали следующий идентификатор пользователя:
  "Marian <zessie94@mail.ru>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? E
Адрес электронной почты: zessie94@mail.ru
Вы выбрали следующий идентификатор пользователя:
  "Marian <zessie94@mail.ru>"
```

Рис. 4.7: Создание ключей

Из предложенных опций выбрала: – тип RSA and RSA; – размер 4096; – выбрала срок действия по умолчанию— 0 (срок действия не истекает никогда). – Имя. – Адрес электронной почты.

8. Добавила PGP ключ в GitHub (рис. 4.8)

```

[mmsungurova@fedora tmp]$ gpg --list-secret-keys --keyid-format LONG
gpg: проверка таблицы доверия
gpg: marginals needed: 2 completes needed: 1 trust model: pgp
gpg: глубина: 0 достоверных: 1 подписанных: 0 доверие: 0-, 0q, 0n, 0m, 0f, 1u
/home/mmsungurova/.gnupg/pubring.kbx
-----
sec   rsa4096/D7DF00942F315C28 2022-04-23 [SC]
      D4288A0BF900A4CDD00E96D70F00942F315C28
uid          [ абсолютно ] Marian <zessie94@mail.ru>
ssb   rsa4096/00486577924C23B7 2022-04-23 [E]

[mmsungurova@fedora tmp]$ "C
[mmsungurova@fedora tmp]$ gpg --armor --export <PGP Fingerprint> | xclip -sel clip
bash: синтаксическая ошибка рядом с неожиданным маркером «|»
[mmsungurova@fedora tmp]$ gpg --armor --export <D7DF00942F315C28> | xclip -sel clip
bash: синтаксическая ошибка рядом с неожиданным маркером «|»
[mmsungurova@fedora tmp]$ gpg --armor --export D7DF00942F315C28 | xclip -sel clip
bash: xclip: command not found...
Install package 'xclip' to provide command 'xclip'? [N/y] y

* Waiting in queue...
The following packages have to be installed:
xclip-0.13-15.git11c0a01.fc35.x86_64  Command line clipboard grabber
Proceed with changes? [N/y] y

* Waiting in queue...
* Waiting for authentication...
* Waiting in queue...
* Downloading packages...
* Requesting data...
* Testing changes...
* Installing packages...

```

Рис. 4.8: Добавление ключа

```

[mmsungurova@fedora tmp]$ gpg --armor --export D7DF00942F315C28 | xclip -sel clip

```

Рис. 4.9: Добавление ключа

9. Перешла в настройки GitHub (<https://github.com/settings/keys>), нажала на кнопку New GPG key и вставила полученный ключ в поле ввода.(рис. 4.10)

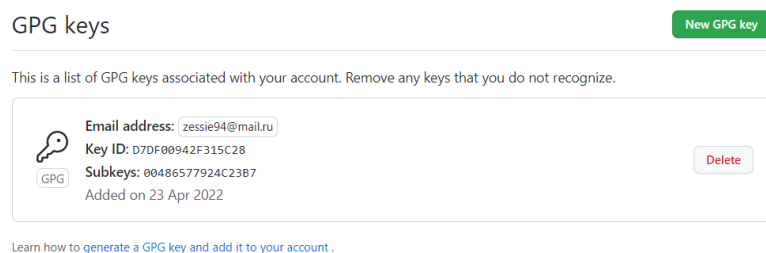


Рис. 4.10: Ключ в поле ввода

10. Произвела настройку автоматических подписей коммитов git (рис. 4.11)

```

[mmsungurova@fedora tmp]$ git config --global user.signingkey D7DF00942F315C28
[mmsungurova@fedora tmp]$ git config --global commit.gpgsign true
git: «config» не является командой git. Смотрите «git --help».

Самые похожие команды:
  config
[mmsungurova@fedora tmp]$ git config --global commit.gpgsign true
[mmsungurova@fedora tmp]$ git config --global gpg.program $(which gpg2)

```

Рис. 4.11: Настройка подписей коммитов

11. Настроила gh (рис. 4.12)

```
mmsungurova@fedora tmp]$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

First copy your one-time code: BDA6-4138
Press Enter to open github.com in your browser...
Authentication complete.
- gh config set -h github.com git_protocol https
- Configured git protocol
- Logged in as mmsungurova
```

Рис. 4.12: Настройка

12. Создала репозиторий курса на основе шаблона (рис. 4.13)

```
mmsungurova@fedora Операционные системы]$ gh repo create study_2021-2022_os-intro --template=yamadharma/course-directory-student-template --public git clone --recursive
unknown flag: --recursive

Usage: gh repo create [<name>] [flags]

Flags:
  -c, --clone                Clone the new repository to the current directory
  -d, --description string   Description of the repository
      --disable-issues       Disable issues in the new repository
      --disable-wiki         Disable wiki in the new repository
  -g, --gitignore string     Specify a gitignore template for the repository
  -h, --homepage URL        Repository home page URL
      --internal             Make the new repository internal
  -l, --license string       Specify an Open Source license for the repository
      --private             Make the new repository private
      --public              Make the new repository public
      --push                Push local commits to the new repository
  -r, --remote string        Specify remote name for the new repository
  -s, --source string        Specify path to local repository to use as source
  -t, --team name           The name of the organization team to be granted access
  -p, --template repository Make the new repository based on a template repository

mmsungurova@fedora Операционные системы]$ gh repo create study_2021-2022_os-intro --template=yamadharma/course-directory-student-template --public
Created repository mmsungurova/study_2021-2022_os-intro on GitHub
```

Рис. 4.13: Создание репозитория на основе шаблона

```
mmsungurova@fedora Операционные системы]$ git clone --recursive git@github.com:mmsungurova/study_2021-2022_os-intro.git os-intro
Cloning into 'os-intro'...
The authenticity of host 'github.com (140.82.121.4)' can't be established.
ED25519 key fingerprint is SHA256:0D1Y3wvV6Tuj3hbpZisF/zLDA8zPMSvHdkr4UvCOqU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
remote: Enumerating objects: 20, done.
remote: Counting objects: 100% (20/20), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 20 (delta 2), reused 15 (delta 2), pack-reused 0
Получение объектов: 100% (20/20), 12.49 KiB | 6.24 MiB/c, готово.
Присвоение именам: 100% (2/2), готово.
Подмодуль «template/presentation» (https://github.com/yamadharma/academic-presentation-markdown-template.git) заархивирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) заархивирован по пути «template/report»
Клонирование в «/home/mmsungurova/work/study/2021-2022/Операционные системы/os-intro/template/presentation»...
remote: Enumerating objects: 42, done.
remote: Counting objects: 100% (42/42), done.
remote: Compressing objects: 100% (34/34), done.
remote: Total 42 (delta 9), reused 40 (delta 7), pack-reused 0
Получение объектов: 100% (42/42), 11.19 KiB | 425.00 KiB/c, готово.
Присвоение именам: 100% (9/9), готово.
Клонирование в «/home/mmsungurova/work/study/2021-2022/Операционные системы/os-intro/template/report»...
remote: Enumerating objects: 78, done.
remote: Counting objects: 100% (78/78), done.
remote: Compressing objects: 100% (52/52), done.
remote: Total 78 (delta 31), reused 69 (delta 22), pack-reused 0
Получение объектов: 100% (78/78), 292.27 KiB | 478.00 KiB/c, готово.
Присвоение именам: 100% (31/31), готово.
Подмодуль по пути «template/presentation»: забрано состояние «3eae6b7586f8a9aded2b506cd1018e625b228b93»
Подмодуль по пути «template/report»: забрано состояние «df7b2ef88f8def3b9a496f8695277469a1a7842a»
```

Рис. 4.14: Создание репозитория

13. Настроила каталог курса (рис. 4.15)

```

[mmsungurova@fedora Операционные системы]$ cd ~/work/study/2021-2022/"Операционные системы"/os-intro
[mmsungurova@fedora os-intro]$ rm package.json
[mmsungurova@fedora os-intro]$ make COURSE=os-intro
[mmsungurova@fedora os-intro]$ git add .
[mmsungurova@fedora os-intro]$ git commit -am 'feat(main): make course structure'
[master 83564c1] feat(main): make course structure
149 files changed, 16590 insertions(+), 14 deletions(-)
create mode 100644 labs/lab01/presentation/Makefile
create mode 100644 labs/lab01/presentation/presentation.md
create mode 100644 labs/lab01/report/Makefile
create mode 100644 labs/lab01/report/bib/cite.bib
create mode 100644 labs/lab01/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab01/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 labs/lab01/report/report.md
create mode 100644 labs/lab02/presentation/Makefile
create mode 100644 labs/lab02/presentation/presentation.md
create mode 100644 labs/lab02/report/Makefile
create mode 100644 labs/lab02/report/bib/cite.bib
create mode 100644 labs/lab02/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab02/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 labs/lab02/report/report.md
create mode 100644 labs/lab03/presentation/Makefile
create mode 100644 labs/lab03/presentation/presentation.md
create mode 100644 labs/lab03/report/Makefile
create mode 100644 labs/lab03/report/bib/cite.bib
create mode 100644 labs/lab03/report/image/placeimg_800_600_tech.jpg

```

Рис. 4.15: Настройка

```

[mmsungurova@fedora os-intro]$ git push
Перечисление объектов: 20, готово.
Подсчет объектов: 100% (20/20), готово.
Сжатие объектов: 100% (16/16), готово.
Запись объектов: 100% (19/19), 266.52 КиБ | 1.80 Миб/с, готово.
Всего 19 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To github.com:mmsungurova/study_2021-2022_os-intro.git
 83f96d9..83564c1 master -> master
[mmsungurova@fedora os-intro]$

```

Рис. 4.16: Настройка

5 Выводы

В этой лабораторной работе я научилась работать с git и подключать удаленные репозитории, добавлять и удалять необходимые файлы. Я также научилась использовать Git Flow, который значительно упрощает разработку проекта и навигацию между ветвями.

#Ответы на контрольные вопросы

1. Системы контроля версий -VCS- это программное обеспечение, которое используется для управления версиями кода.
- Репозиторий в системе контроля версий - Это удаленный репозиторий, в котором хранятся все изменения кода.
- commit - фиксирует изменения перед загрузкой файлов в систему контроля версий
- история хранит все изменения в проекте, и при необходимости позволяет откатиться к предыдущей версии.
- рабочая копия - это копия проекта на компьютере разработчика. Если другой член команды внесет изменения, рабочая копия будет обновлена.
3. В отличие от классических, в распределённых системах контроля версий центральная копия не требуется.
4. Для начала необходимо создать и подключить удаленный репозиторий. Затем, после создания репозитория, можно начинать работу над проектом.
5. Участник проекта (пользователь) перед началом работы посредством определённых команд устанавливает локальную копию проекта.
6. Упрощение обмена информацией, ускорение разработки, устранение ошибок и недочётов.
- git init - инициализирует локальный репозиторий
- git add * или add. - добавляет файлы в репозиторий
- git commit - версия фиксации
- git pull - загружает текущую версию проекта
- git push - отправляет измененный проект на сервер
- git checkout - позволяет переключаться между ветками
- git status - текущий статус проекта
- git branch - просмотреть доступные ветки

- `git remote add` - добавить удаленный репозиторий

8. `git push --all` (push origin master/любой branch)

9. Ветви функций, также иногда называемые ветвями тем, используются для разработки

10. Существуют временные и системные файлы, которые загромождают проект и не нужны

`doc: bash;@tannenbaum:arch-pc:ru;@tannenbaum:modern-os:ru]`

Список литературы