

Отчёт по лабораторной работе №2

Дисциплина: архитектура компьютера

Лысенко Маргарита Олеговна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Выводы	13

Список иллюстраций

4.1	Регистрация	9
4.2	Владелец репозитория	9
4.3	Настройка utf-8	9
4.4	Имя начальной ветки	9
4.5	Настройка параметров	10
4.6	Генерация ключей	10
4.7	Загрузка ключа	10
4.8	Каталог	10
4.9	Клонирование	11
4.10	Удаление	11
4.11	Проверка	11
4.12	Проверка на github	12

Список таблиц

1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git.

2 Задание

Зарегистрироваться на github, создать отчёты и загрузить их туда.

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 не полную 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 вовсе или заблокировать файлы для изменения. В зави-

симости 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.

Система контроля версий Git представляет собой набор программ командной строки. Доступ к ним можно получить из терминала посредством ввода команды `git` с различными опциями. Благодаря тому, что Git является распределённой системой контроля версий, резервную копию локального хранилища можно сделать простым копированием или архивацией.

4 Выполнение лабораторной работы

Создала учётную запись на сайте github (рис. 4.1).

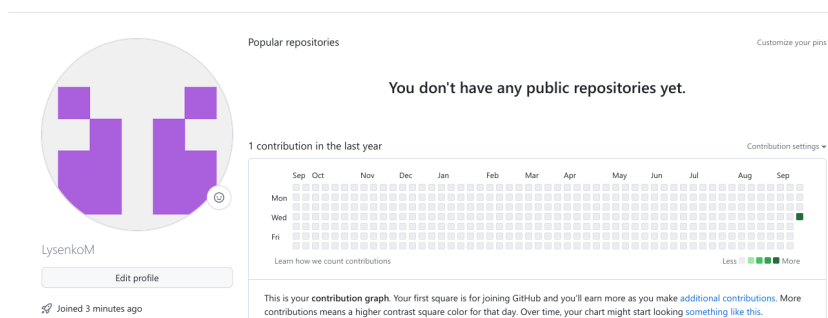


Рис. 4.1: Регистрация

Указала имя и email владельца репозитория (рис. 4.2).

```
molihsenko@dk4n63 ~ $ git config --global user.name "LysenkoM"
molihsenko@dk4n63 ~ $ git config --global user.email "rita.lysenko.05@bk.ru"
```

Рис. 4.2: Владелец репозитория

Настроила utf-8 в выводе сообщений git (рис. 4.3).

```
molihsenko@dk4n63 ~ $ git config --global core.quotePath false
```

Рис. 4.3: Настройка utf-8

Задала имя начальной ветки (master) (рис. 4.4).

```
molihsenko@dk4n63 ~ $ git config --global core.autocrlf input
```

Рис. 4.4: Имя начальной ветки

Настроила параметры autocrlf и safecrlf (рис. 4.5).

```
molihsenko@dk4n63 ~ $ git config --global core.autocrlf input
molihsenko@dk4n63 ~ $ git config --global core.safecrlf warn
```

Рис. 4.5: Настройка параметров

Сгенерировала пару ключей (приватный и открытый) (рис. 4.6).

```
molihsenko@dk4n63 ~ $ git config --global user.email "rita.lysenko.05@bk.ru"
molihsenko@dk4n63 ~ $ ssh-keygen -C "Лысенко Маргарита rita.lysenko.05@bk.ru"
Generating public/private rsa key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/m/o/molihsenko/.ssh/id_rsa):
Created directory '/afs/.dk.sci.pfu.edu.ru/home/m/o/molihsenko/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /afs/.dk.sci.pfu.edu.ru/home/m/o/molihsenko/.ssh/id_rsa
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/m/o/molihsenko/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:ix2f73Qj320FDns/6tuFyYcfT4nWSwXl+OCtOyefMt0 Лысенко Маргарита rita.lysenko.05@bk.ru
The key's randomart image is:
+---[RSA 3072]---+
|                 .|
|                + |
|               +  |
|              . = |
|             .   |
|            S    +|
```

Рис. 4.6: Генерация ключей

Загрузила сгенерённый открытый ключ, скопировав из локальной консоли ключ в буфер обмена. Вставила ключ в поле на сайте, указав имя ключа (рис. 4.7).

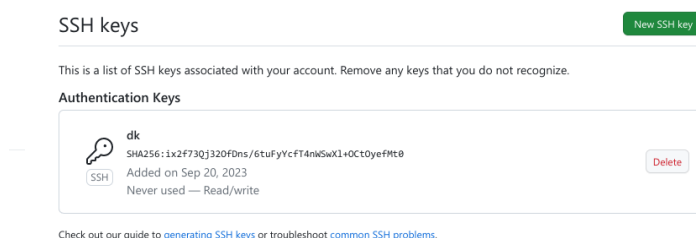


Рис. 4.7: Загрузка ключа

Создала каталог для предмета “Архитектура компьютера” (рис. 4.8).

```
molihsenko@dk4n63 ~ $ mkdir -p ~/work/study/2023-2024/"Архитектура компьютера"
```

Рис. 4.8: Каталог

Перешла в каталог курса и клонировала созданный репозиторий (рис. 4.9).

```

molihsenko@dk4n63 ~ $ cd work/study/2023-2024/Архитектура\ компьютера/
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера $ git clone --recursive git@github.com:LysenkoM/arch-ps.git arch-pc
Клонирование в «arch-pc»...

```

Рис. 4.9: Клонирование

Удалила лишние файлы и создала необходимые каталоги. Отправила файлы на сервер (рис. 4.10).

```

molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ rm package.json
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ echo arch-pc > COURSE
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ cat C
CHANGELOG.md COURSE
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ cat C
CHANGELOG.md COURSE
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ cat COURSE
arch-pc
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ make
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git add .
molihsenko@dk4n63 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git commit -am 'feat(main): make course structure'
[master 940abcc] feat(main): make course structure

```

Рис. 4.10: Удаление

Проверила правильность создания иерархии рабочего пространства в локальном репозитории (рис. 4.11).

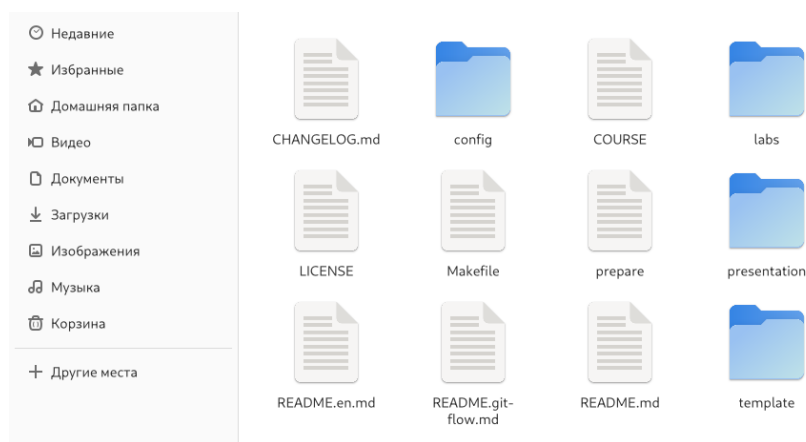


Рис. 4.11: Проверка

Проверила правильность создания иерархии рабочего пространства на странице github (рис. 4.12).


 LysenkoM Update README.md		d657ae7 last week	🕒 3 commits
📁 config	Initial commit	last week	
📁 labs	feat(main): make course structure	last week	
📁 presentation	feat(main): make course structure	last week	
📁 template	Initial commit	last week	
📄 .gitattributes	Initial commit	last week	
📄 .gitignore	Initial commit	last week	
📄 .gitmodules	Initial commit	last week	
📄 CHANGELOG.md	Initial commit	last week	
📄 COURSE	feat(main): make course structure	last week	
📄 LICENSE	Initial commit	last week	
📄 Makefile	Initial commit	last week	

Рис. 4.12: Проверка на github

5 Выводы

В ходе лабораторной работы я изучила идеологию и применение средств контроля версий. Приобрела практические навыки по работе с системой git.