Отчёт по лабораторной работе №2

Дисциплина: Архитектура компьютера

Толстых Александра Андреевна

Содержание

# 1 Цель работы

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

# 2 Задание

1. Настройка github
2. Базовая настройка git
3. Создание SSH ключа
4. Создание рабочего пространства и репозитория курса на основе шаблона
5. Создание репозитория курса на основе шаблона
6. Настройка каталога курса
7. Задание для самостоятельной работы

# 3 Теоретическое введение

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.

Приведем таблицу снаиболее часто используемыми командами git в таблице №1.

| Команда | Описание |
| --- | --- |
| git init | Создание основного дерева репозитория |
| git pull | Получение обновлений (изменений) текущего дерева из центрального репозитория |
| git push | Отправка всех произведенных изменений локального дерева в центральный репозиторий |
| git status | Просмотр списка измененных файлов в текущей директории |
| git diff | Просмотр текущих изменений |
| gitt add . | Добавить все изменённые или созданные файлы или каталоги |
| git add имена\_файлов | Добавить конкретные изменённые или созданные файлы или каталоги |
| git rm имена\_файлов | Удалить файл и/или каталог из индекса репозитория (при этом файл или каталог остаётся в локальной директории) |
| git commit -am 'Описание коммита' | Сохранить все добавленные изменения и все изменённые файлы |
| git checkout -b имя\_ветки | Создание новой ветки, базирующейся на текущей |
| git checkout имя\_ветки | Переключение на некоторую ветку (при переключении на ветку, которой ещё нет в локальном репозитории, она будет создана и связана с удалённой) |
| git push origin имя\_ветки | Отправка изменений конкретной ветки в центральный репозиторий |
| git merge --no-ff имя\_ветки | Слияние ветки с текущим деревом |
| git branch -d имя\_ветки | Удаление локальной уже слитой с основным деревом ветки |
| git branch -D имя\_ветки | Принудительное удаление локальной ветки |
| git push origin :имя\_ветки | Удаление ветки с центрального репозитория |

# 4 Выполнение лабораторной работы

## 4.1 Настройка github

Сначала мы создаем учетную запись на сайте github и заполняем основные данные. Поскольку сайт не позволял регистрировать учетную запись с использованием корпоративной почты (рис. 1), я зашла на сайт используя свою уже существующую учетную запись на github, созданную ранее на личную почту (рис. 2).

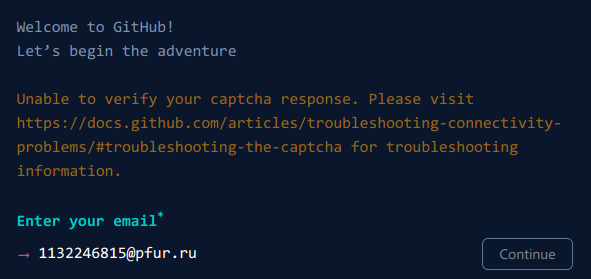


Рис. 1: Проблемы с регистрацией на сайте

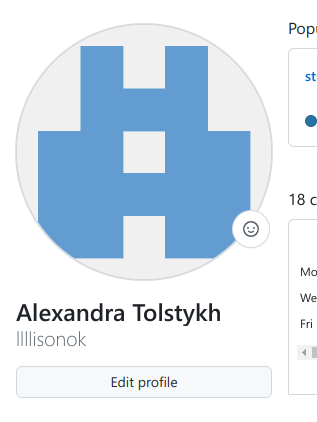


Рис. 2: Созданная ранее учётная запись

## 4.2 Базовая настройка git

Открываю терминал и делаю предварительную конфигурацию git (рис. 3). Ввожу команды «git config –global user.name» и «git config –global user.email», указывая данные от своего аккаунта. При помощи команды «git config –global core.quotepath false» настраиваю utf-8 в выводе сообщений git для их корректного отображения. Далее задаю имя начальной ветки «master», а также параметры «autocrlf» (параметр необходим для настройки конвертации crlf в lf) и «safecrlf» (параметр необходим для проверки преобразования на обратимость).

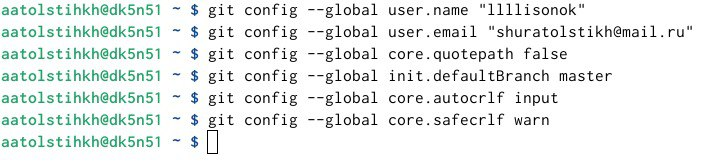


Рис. 3: Выполнение команд для базовой настройки git

## 4.3 Создание SSH-ключа

Для последующей идентификации пользователя на сервере репозиториев необходимо сгенерировать приватный и открытый ключи. Ввожу команду «sshkeygen -C» указывая как аргументы свои данные: имя, фамилию и почту (рис. 4). Ключи сохраняются в каталоге ~/.ssh/.

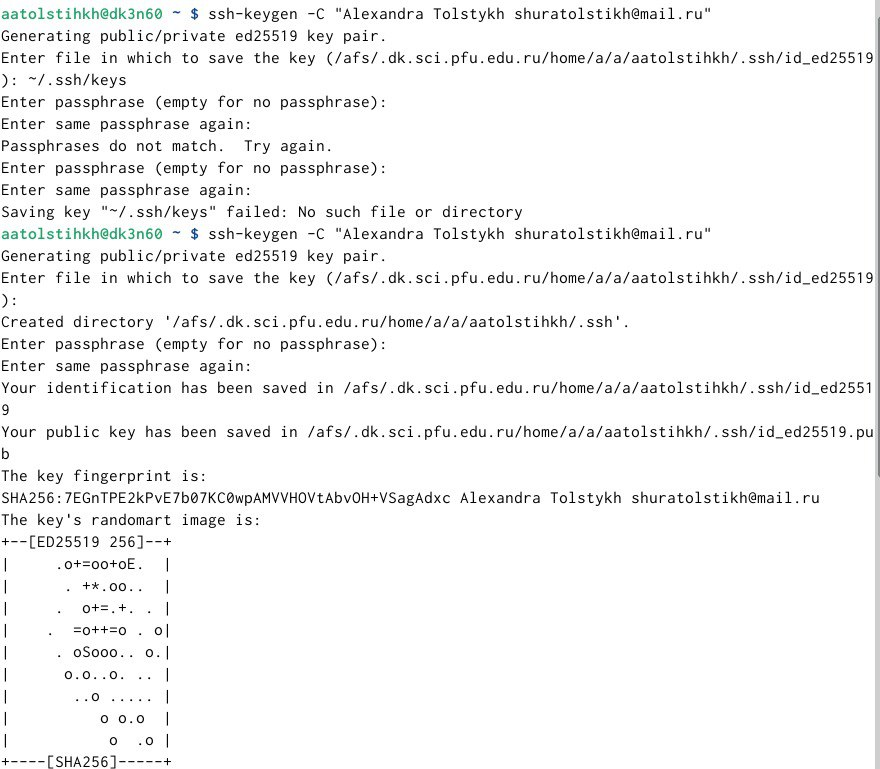


Рис. 4: Создание SSH-ключей

Копируем ключ в буфер обмена при помощи команды «cat ~/.ssh/id\_rsa.pub | xclip -sel clip» (рис. 5).

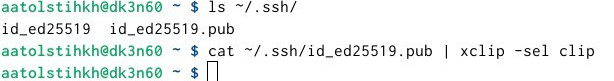


Рис. 5: Копирование открытого ключа из файла

Вставляю скопированный ключ в поле на сайте, чтобы добавить его (рис. 6).

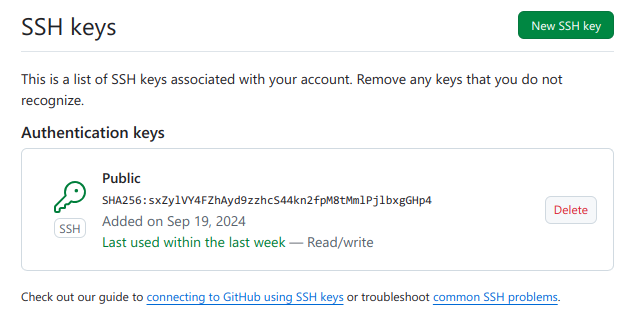


Рис. 6: Добавление ключа через сайт

## 4.4 Создание рабочего пространства и репозитория курса на основе шаблона

При выполнении лабораторных работ следует придерживаться структуры рабочего пространства, поэтому при помощи команды mkdir с аргументом -p создаю нужную вложенную последовательность каталогов (рис. 7).

Рис. 7: Создание вложенных каталогов согласно структуре пространства

Рис. 7: Создание вложенных каталогов согласно структуре пространства

## 4.5 Создание репозитория курса на основе шаблона

Репозиторий на основе шаблона можно создать через web-интерфейс github. Для этого перехожу на станицу репозитория с шаблоном курса по ссылке, указанной в лабораторной работе. Далее использую кнопку «Use this template» (рис. 8).

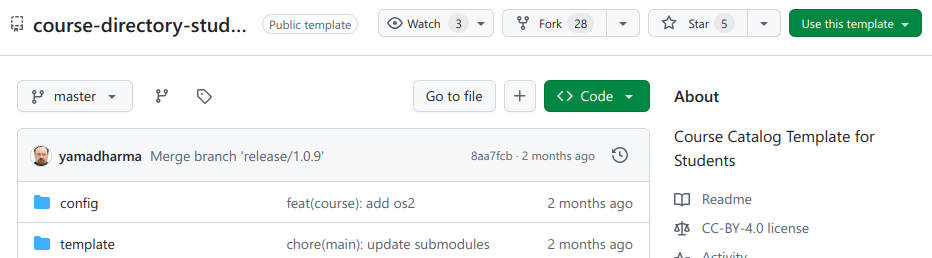


Рис. 8: Использование шаблона

В открывшемся окне задаю имя репозитория «study\_2024-2025\_arhpc» и создаю его при помощи кнопки «Create repository from template» (рис. 9).

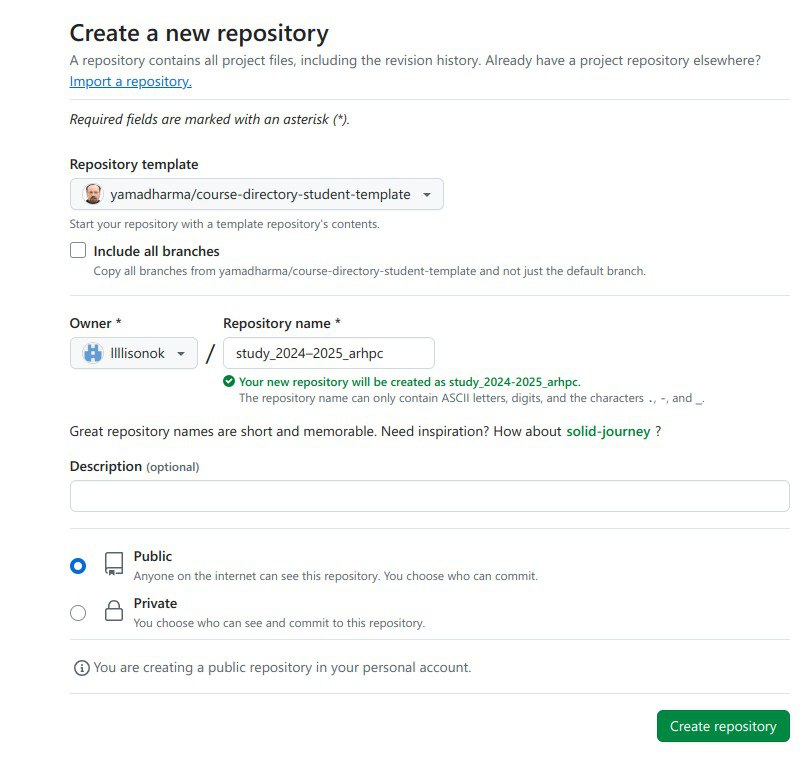


Рис. 9: Создание репозитория на основе шаблона

Через терминал перехожу в созданный каталог курса при помощи команды cd. Клонирую данный репозиторий при помощи команды «git clone –recursive git@github.com:/llllisonok/study\_2024-2025\_arhpc.git» (рис. 10).

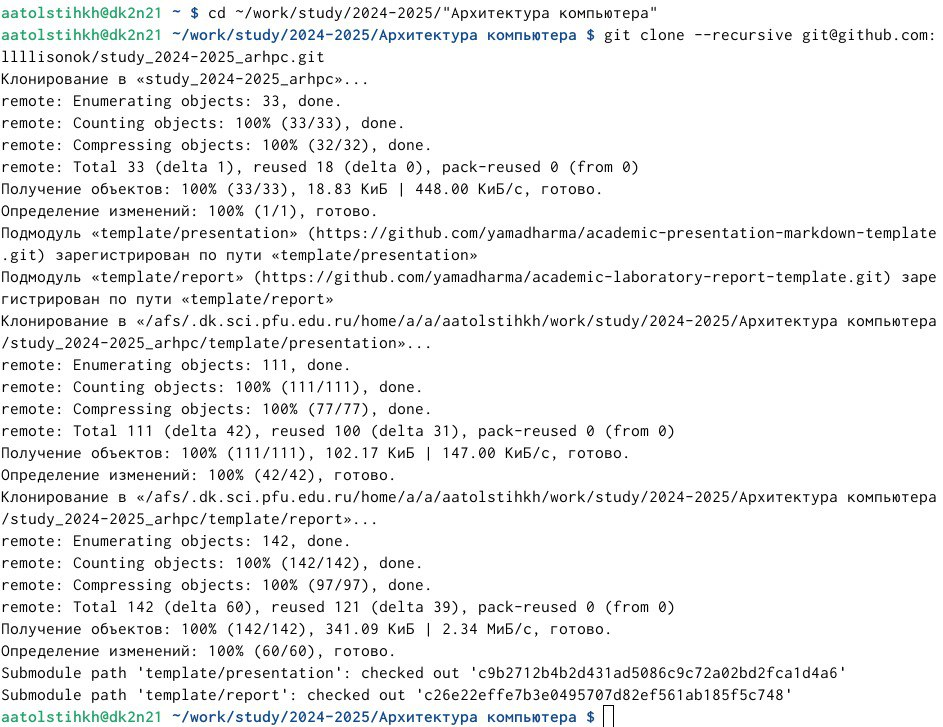


Рис. 10: Клонирование репозитория

## 4.6 Настройка каталога курса

Перехожу в каталог курса при помощи команды cd, а затем при помощи команды rm удаляю ненужные файлы (рис. 11).



Рис. 11: Удаление ненужных файлов

Далее создаю необходимые каталоги так, как это указано в задании (рис. 12).

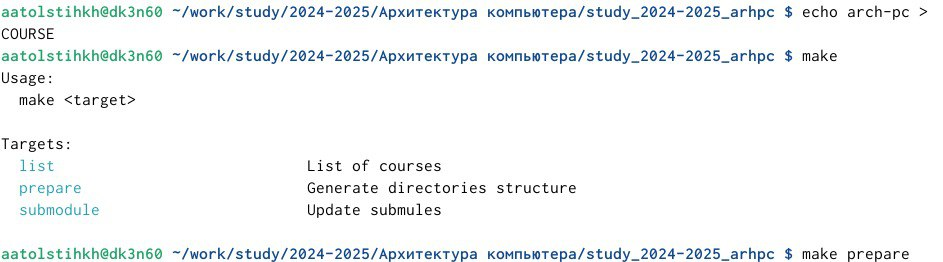


Рис. 12: Создание необходимых каталогов

Используя команды «git add .», «git commit -am», «git push» комментирую и сохраняю внесенные изменения (рис. 13).

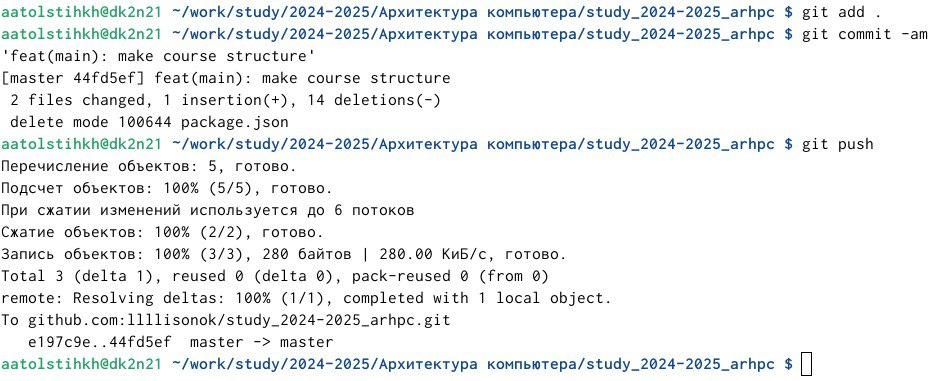


Рис. 13: Комментирование и сохранение изменений

Затем открываю сайт и проверяю, что все выполнилось корректно (рис. 14). Для этого сравниваю папки своего репозитория с шаблоном, отличие должно быть в двух файлах: «package.json» и «labs». Это действительно так. Значит каталог курса настроен корректно.

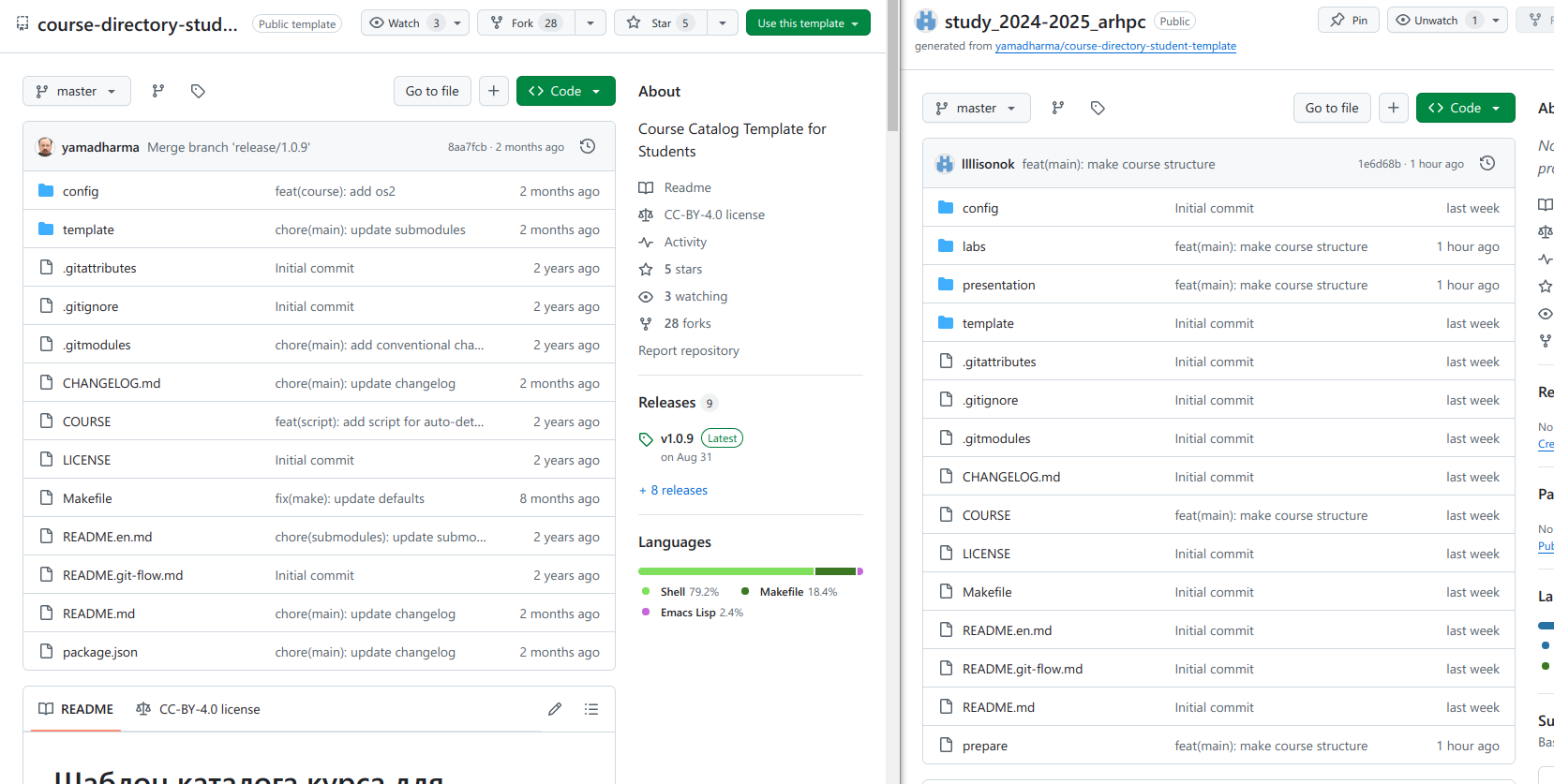


Рис. 14: Сравнение репозиториев моего курса и шаблона

## 4.7 Задание для самостоятельной работы

Сначала загружу отчет о выполнении первой лабораторной (рис. 15). Для этого сохраню его в папку «Документы», а затем скопирую в папку «lab01». Затем комментирую и сохраняю изменения.

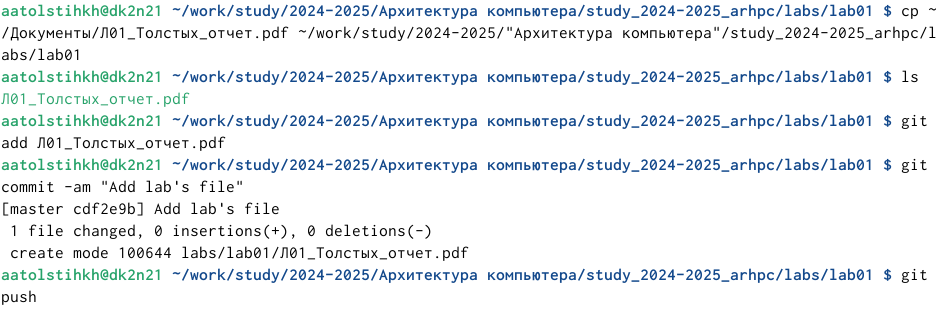


Рис. 15: Загрузка отчета первой лабораторной

Далее, чтобы убедиться в приобретенных навыках, перехожу в папку для второй лабораторной и при помощи команды touch создаю файл для отчета данной (второй) лабораторной работы (рис. 16). Комментирую и сохраняю изменения. (Данный файл позже будет заменен, поскольку отчеты о лабораторных я пишу дома, действия выполняются лишь для того чтобы убедиться, что необходимые команды изучены).

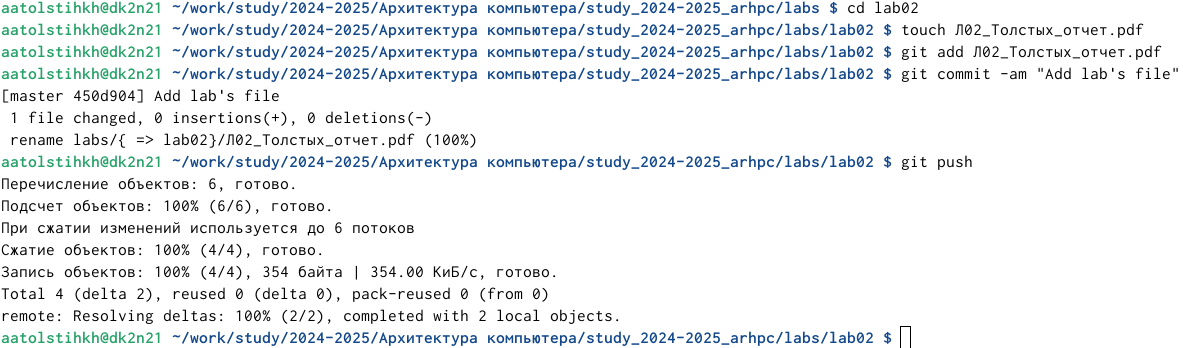


Рис. 16: Создание файла для второй лабораторной

Затем открываю сайт и проверяю наличие отчета о первой лабораторной работе, а также файла для отчета о второй лабораторной работе. Все выполнено корректно. Далее завершаю написание отчета и через сайт заменяю файл с отчетом второй лабораторной.

# 5 Выводы

В результате выполнения данной лабораторной работы я изучила идеологию и применение средств контроля версий, а также приобрела практические навыки по работе с системой git.