Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ **ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ**

КАФЕДРА **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.04.01 - ИУ6-32Б

О Т Ч Е Т

по лабораторной работе № 1

Название: **Основы Git & GitHub**

Дисциплина: Языки интернет-программирования

Студент ИУ6-32Б _____ Кондратов С.Ю.
(Группа) (Подпись, дата) (И.О. Фамилия)

Преподаватель _____ Шульман В.Д.
(Подпись, дата) (И.О. Фамилия)

Москва, 2024

Цель работы: научиться работать с Git.

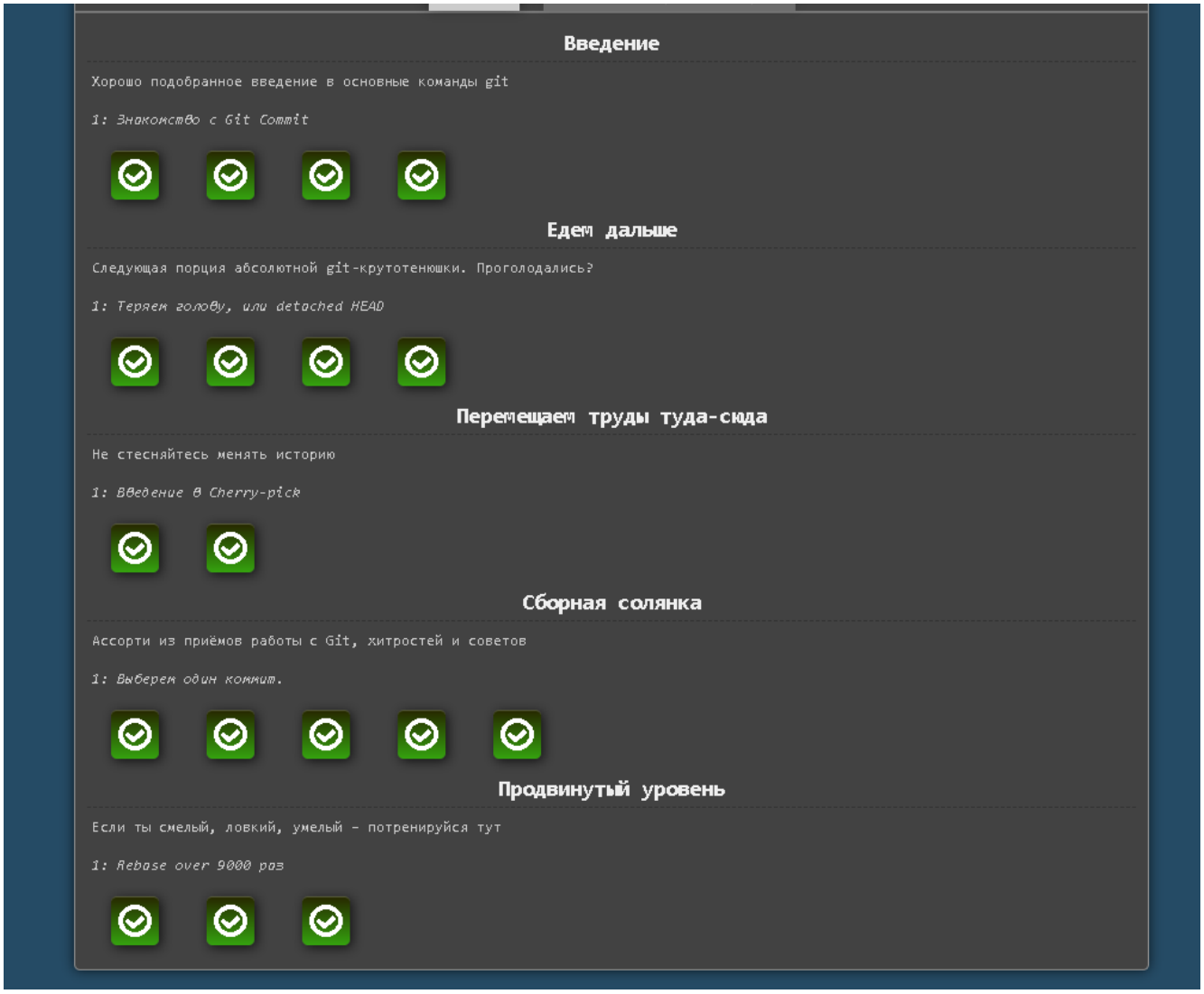

Рисунок 1

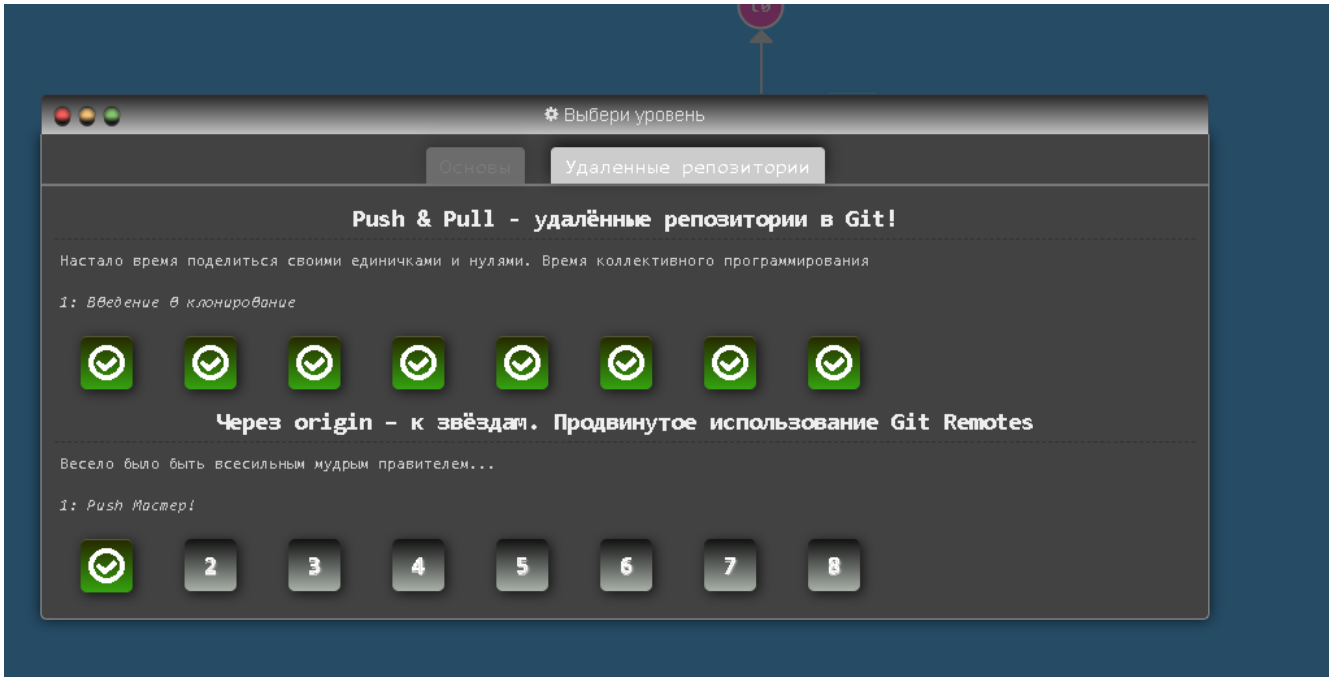

Рисунок 2

Рисунки 1,2 - результаты практики с Git на https://learngitbranching.js.org/?locale=ru_RU

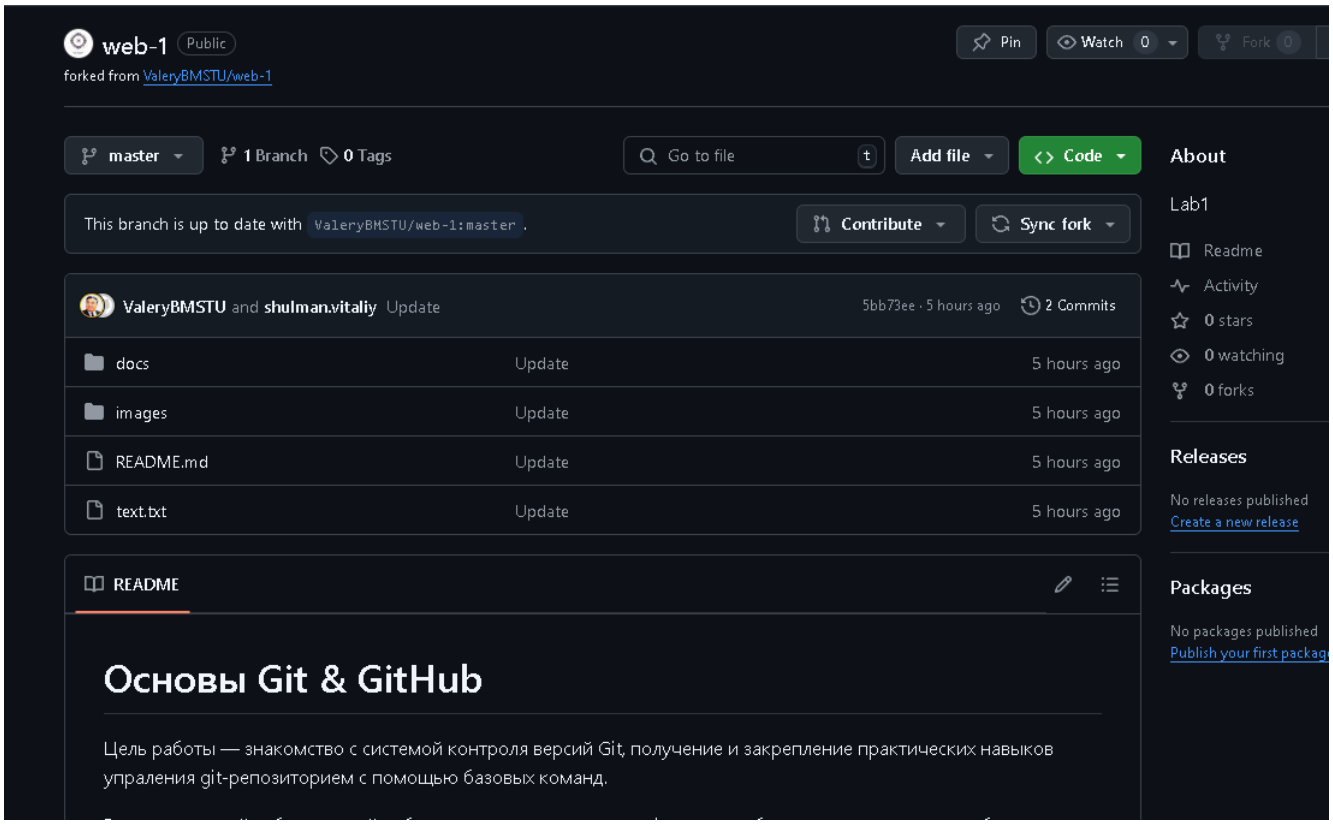

Рисунок 3

Рисунок 3 - результат выполнения команды fork.

Поскольку github аккаунт у меня уже есть, остается только сгенерировать ssh-ключ. Для этого следуя инструкции необходимо ввести команду:

```
ssh-keygen -t rsa -b 4096 -C "your email address"
```

B GitBash

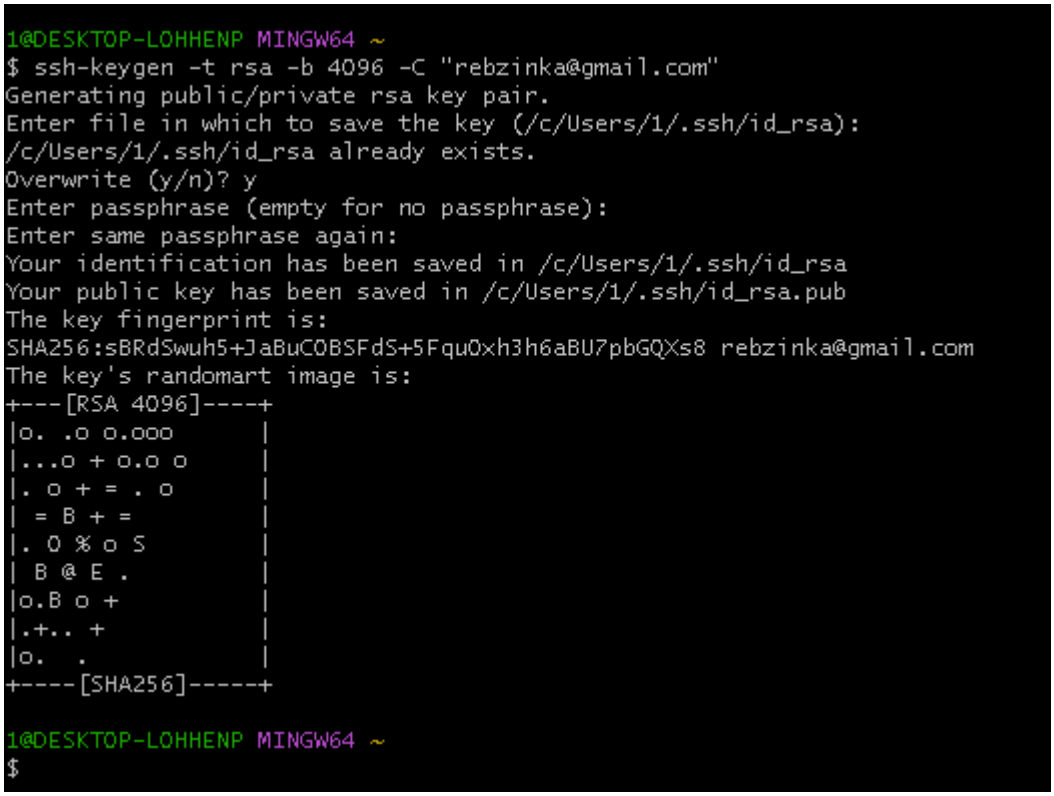

Рисунок 4

Рисунок 5

На рис 4,5 представлены результаты выполнения команд из инструкции.

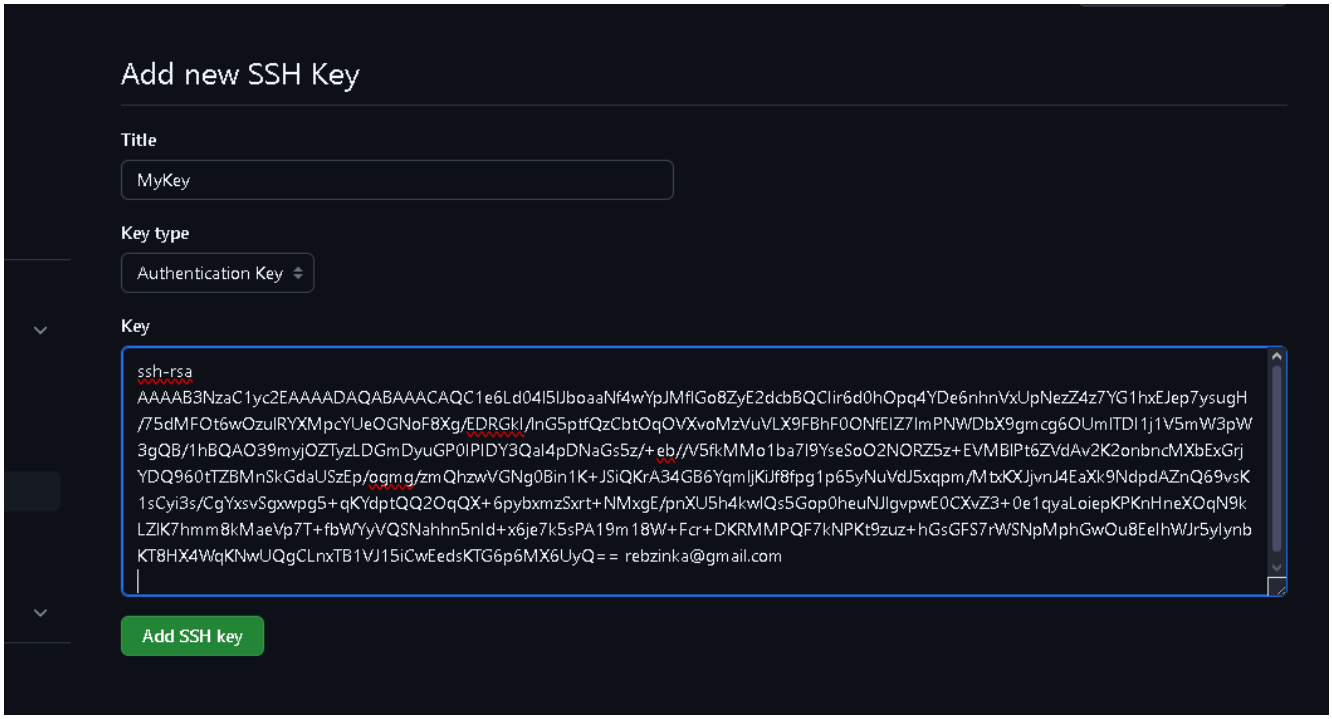

Рисунок 6

Рисунок 7

На рис 6,7 представлены последние шаги добавления ключа в настройках github аккаунта.

```
PS C:\Users\1> cd desktop
PS C:\Users\1\Desktop> mkdir lab1git

    Directory: C:\Users\1\Desktop

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d----           07.09.2024    16:15                lab1git

PS C:\Users\1\Desktop> cd lab1git
PS C:\Users\1\Desktop\lab1git> git clone git@github.com:StepOne-ai/web-1.git
Cloning into 'web-1'...
remote: Enumerating objects: 14, done.
remote: Counting objects: 100% (2/2), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 14 (delta 0), reused 0 (delta 0), pack-reused 12 (from 1)
Receiving objects: 100% (14/14), 588.78 KiB | 1.11 MiB/s, done.
PS C:\Users\1\Desktop\lab1git>
```

Рисунок 8

На рисунке 8 показан процесс clone-ирования репозитория.

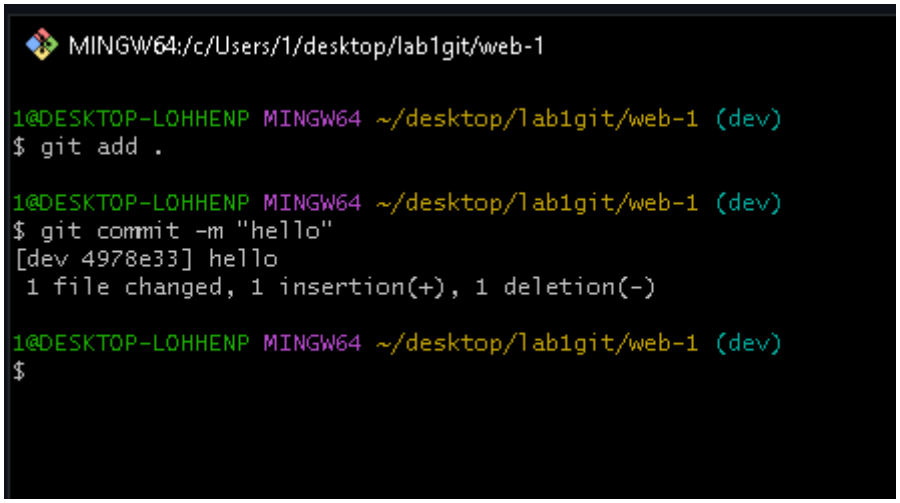

Рисунок 9 (Изменений файла text.txt)

Заключение

Я научился работать с git. Выполнять commit-ы. Создавать ssh ключ. Fork-ать и clone-ировать чужие public репозитории на GitHub.

Контрольные вопросы

1. **В чём разница между Git и Github?** - Git - это инструмент, а Github - это платформа, которая использует Git для управления репозиториями.

Примерно так же, как Word - это текстовый редактор, а Google Docs - это онлайн-сервис, который позволяет работать с документами в облаке.

2. **Как можно объединить несколько коммитов в один коммит?** - Чтобы объединить несколько коммитов в один коммит, можно использовать команду **git rebase -i**. Эта команда позволяет вам интерактивно редактировать историю коммитов.
3. **Для чего нужен git rebase, если есть git merge? - git merge** используется для объединения двух веток, создавая новый коммит, который объединяет изменения из обеих веток. При этом создается новый коммит, который содержит все изменения из обеих веток. Это означает, что история коммитов остается неизменной, и вы можете видеть, какие изменения были внесены в каждой ветке. **git rebase,** с другой стороны, используется для повторного применения коммитов из одной ветки на другую. При этом коммиты из исходной ветки применяются в том же порядке, в котором они были созданы, но уже на новой ветке. Это означает, что история коммитов изменяется, и коммиты из исходной ветки становятся частью новой ветки.

4. **Опишите назначение комманд: clone, add, pull, commit, push, merge, rebase -**

Git clone - Клонировать существующий репозиторий из удаленного источника (например, GitHub, GitLab) на локальную машину. (**git clone https://github.com/user/repo.git)**

Git add - Добавить изменения в индекс (stage) для последующего коммита. (**git add .)**

Git pull - Обновить локальный репозиторий, скачав последние изменения из удаленного репозитория и объединив их с локальными изменениями. (**git pull origin master)**

Git commit - Зафиксировать изменения в репозитории, создав новый коммит.
(**git commit -m "Commit message")**

Git push - Отправить локальные изменения в удаленный репозиторий. (**git push origin master)**

Git merge - Объединить две ветки, создав новый коммит, который объединяет изменения из обеих веток. (**git merge feature/new-feature)**

Git rebase - Повторно применить коммиты из одной ветки на другую, создав линейную историю коммитов. (**git rebase master)**

1. **mkdir myproject** - Создайте новую директорию для вашего проекта
2. **cd myproject** - Перейдите в созданную директорию
3. **git init** - Инициализируйте Git-репозиторий
4. **git add .** - Добавьте файлы в репозиторий
5. **git commit -m "Initial commit"** - Создайте первый коммит