



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 09.03.01 Информатика и вычислительная техника

О Т Ч Е Т

по лабораторной работе № 1

Название: Основы Git & GitHub

Дисциплина: Языки интернет-программирования

Студент

ИУ6-33Б

(Группа)

(Подпись, дата)

Н.Р. Гусниев

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

(И.О. Фамилия)

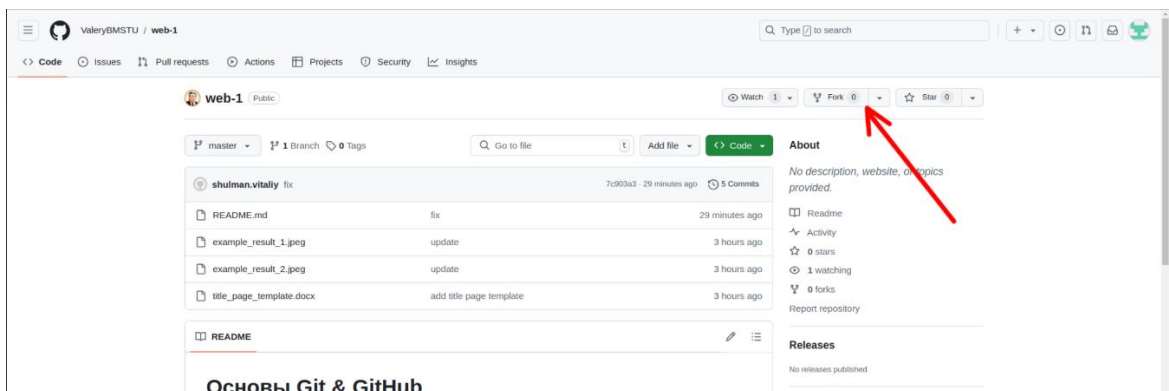
Москва, 2024

ЦЕЛЬ РАБОТЫ

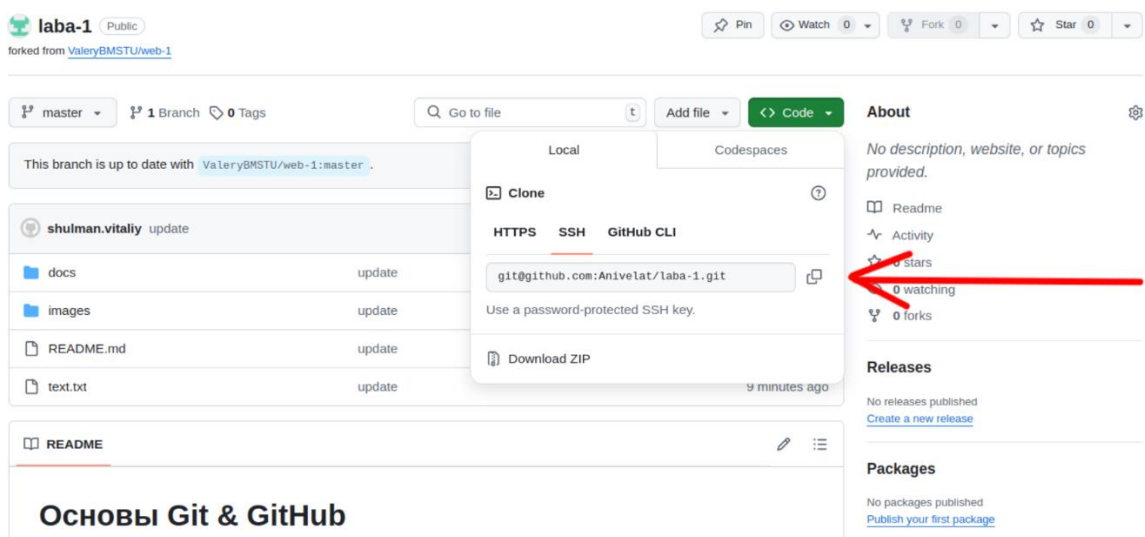
знакомство с системой контроля версий Git, получение и закрепление практических навыков управления git-репозиторием с помощью базовых команд.

ЗАДАНИЕ

1. Установите утилиту Git: <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>
2. Следуя инструкции <https://dev.to/joshhortt/how-to-generate-and-add-an-ssh-key-to-github-1fe1>, зарегистрируйтесь на <https://github.com>, сгенерируйте ssh-ключ и положите его в ваш личный кабинет
3. Далее, вам необходимо сделать свою собственную копию репозитория с данной лабораторной работой. Через интерфейс GitHub сделайте Fork текущего репозитория



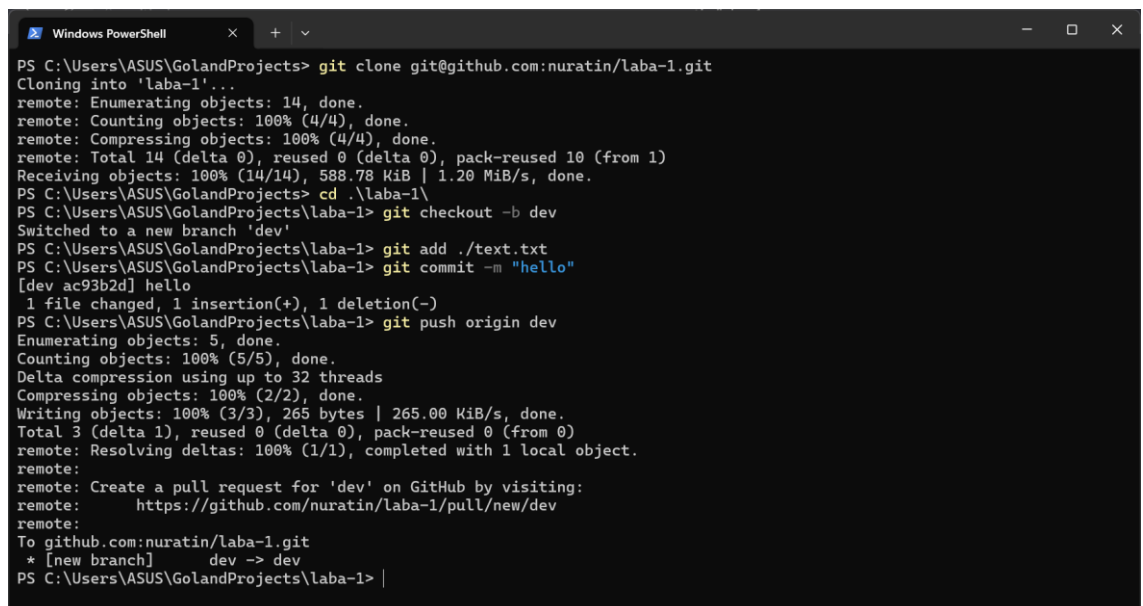
4. Склонируйте форкнутый репозиторий на свою локальную машину по ssh с помощью команды `git clone git@github.com:<адрес вашего репозитория>.git` и перейдите в соответствующую директорию



5. Создайте локально от ветки master ветку dev и переключитесь на неё с помощью команды `git checkout -b dev`
6. Модифицируйте файл `text.txt`, заменив слово "world" на ваше имя, после чего проиндексируйте изменения с помощью команды `git add ./text.txt` и зафиксируйте изменения с помощью команды `git commit -m "hello"`
7. По аналогии с предыдущим шагом, добавьте отчёт по лабораторной работе в директорию `docs` в формате pdf (шаблон титульника находится там же)
8. Запушьте локальную dev-ветку в свой удаленный репозиторий GitHub с помощью команды `git push origin dev` и создайте Pull request из dev в master с помощью интерфейса GitHub
9. Продемонстрируйте результаты в ходе защиты лабораторной работы

ХОД РАБОТЫ

Задание успешно выполнено, скриншоты приведены ниже:



```
PS C:\Users\ASUS\GolandProjects> git clone git@github.com:nuratin/laba-1.git
Cloning into 'laba-1'...
remote: Enumerating objects: 14, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 14 (delta 0), reused 0 (delta 0), pack-reused 10 (from 1)
Receiving objects: 100% (14/14), 588.78 KiB | 1.20 MiB/s, done.
PS C:\Users\ASUS\GolandProjects> cd .\laba-1\
PS C:\Users\ASUS\GolandProjects\laba-1> git checkout -b dev
Switched to a new branch 'dev'
PS C:\Users\ASUS\GolandProjects\laba-1> git add ./text.txt
PS C:\Users\ASUS\GolandProjects\laba-1> git commit -m "hello"
[dev ac93b2d] hello
1 file changed, 1 insertion(+), 1 deletion(-)
PS C:\Users\ASUS\GolandProjects\laba-1> git push origin dev
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 32 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 265 bytes | 265.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'dev' on GitHub by visiting:
remote:   https://github.com/nuratin/laba-1/pull/new/dev
remote:
To github.com:nuratin/laba-1.git
 * [new branch]      dev -> dev
PS C:\Users\ASUS\GolandProjects\laba-1> |
```

Рисунок 1

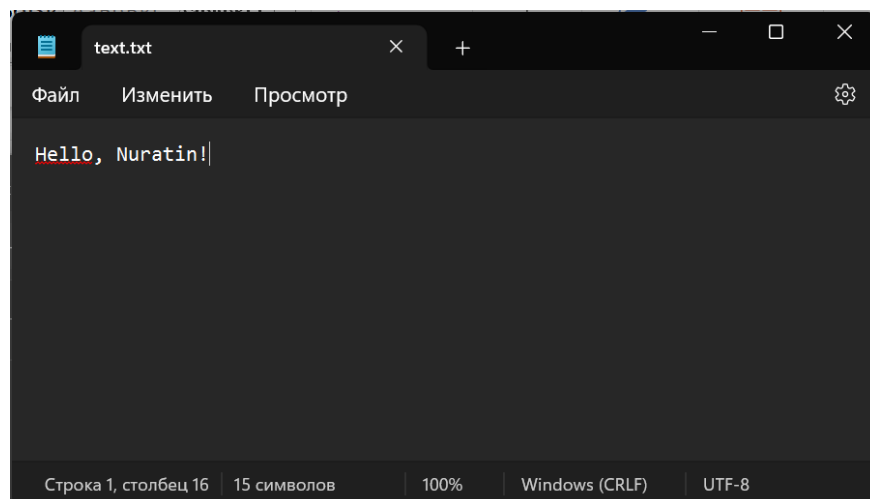


Рисунок 2

ЗАКЛЮЧЕНИЕ

В ходе лабораторной работы были успешно выполнены все этапы задания: от установки Git и настройки SSH-ключа до работы с удалённым репозиторием на GitHub. Были освоены базовые команды Git для управления ветками и фиксации изменений. Полученные навыки позволили продемонстрировать понимание принципов работы с системой контроля версий, что подтверждается созданием Pull request и его успешной интеграцией.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Разница между Git и GitHub:

Git — это система контроля версий, которая позволяет управлять изменениями в коде локально. GitHub — это веб-платформа для хранения Git-репозитория, их совместного использования и управления проектами в облаке.

2. Как объединить несколько коммитов в один коммит?

Для объединения нескольких коммитов в один можно использовать команду `git rebase -i` (интерактивный ребейз), где выбирается команда `squash` для нужных коммитов.

3. Для чего нужен `git rebase`, если есть `git merge`?

`git merge` сохраняет историю изменений с сохранением всех веток, тогда как `git rebase` переписывает историю, линейно объединяя изменения, что делает историю более чистой и прямолинейной.

4. Назначение команд:

- `clone`: копирует удалённый репозиторий на локальный компьютер.
- `add`: добавляет файлы в индекс для последующего коммита.
- `pull`: скачивает изменения с удалённого репозитория и объединяет их с текущей веткой.
- `commit`: сохраняет изменения в локальный репозиторий.
- `push`: отправляет коммиты в удалённый репозиторий.
- `merge`: объединяет изменения из одной ветки в другую.
- `rebase`: переписывает историю коммитов, применяя изменения одной ветки поверх другой.

5. Как локально создать Git-репозиторий с нуля?

Для создания локального репозитория с нуля используется команда `git init`, которая инициализирует пустой репозиторий в выбранной директории.