Шаблон отчёта по лабораторной работе

Первая лабораторная работа

Аристид Жан Лоэнс Аристобуль

Содержание

# 1 Цель работы

Знакомство с git Hub и его фундаментальными понятиями.

# 2 Теоретическое введение

B табл. 1 приведено базовые команды Git.

Таблица 1: Описание некоторых команд Git.

| Команды | Описание Командов |
| --- | --- |
| git config | Чтобы настроить git в нашем компьютере. |
| git init | Чтобы переобразывать каталог в репозытории добавляя файл .git в выбранном каталоге |
| git add | Чтобы индексировать наши изменение |
| git commit | Чтобы сокранить наши изменение в репозытории |
| git remote | Позваляет просмотрить удаленный из которого наш локальный репозыторый извлечил данный или позволяет уточнить репозыторый к которому собираем отправить файли |
| git push | Чтобы отправить наши измениние в удалённый репозыторый |
| git clone | Чтобы клонировать удалённый репозыторый в наш компьютер |
| git pull | Чтобы клонировать и слиться новые измениние удалённого репозыторый в наш локалный репозыторый |

# 3 Выполнение лабораторной работы

В этом изображении, мы устроим наш git (рис. 1).

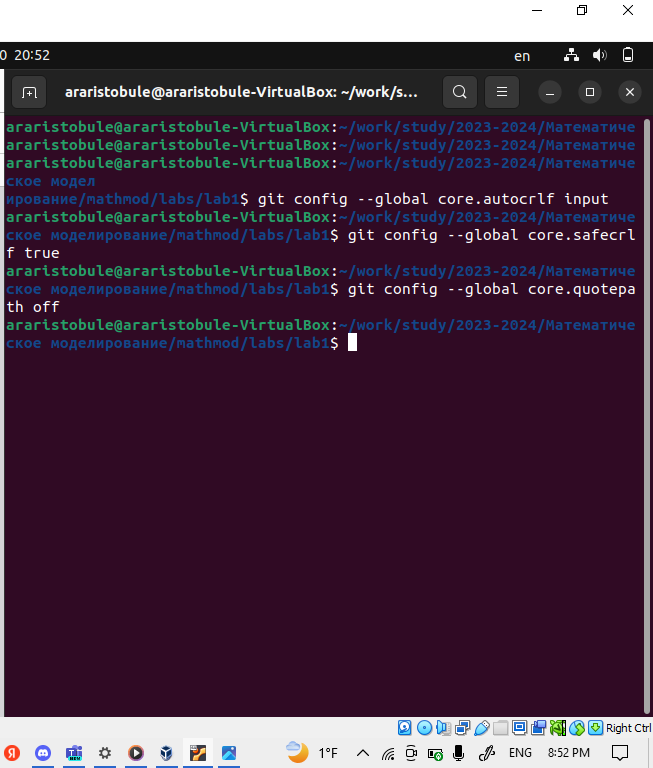


Рис. 1: Устроить Git

Создание каталога hello и внутри его файл hello которой мы модифировали с помошью команда echo (рис. 2).

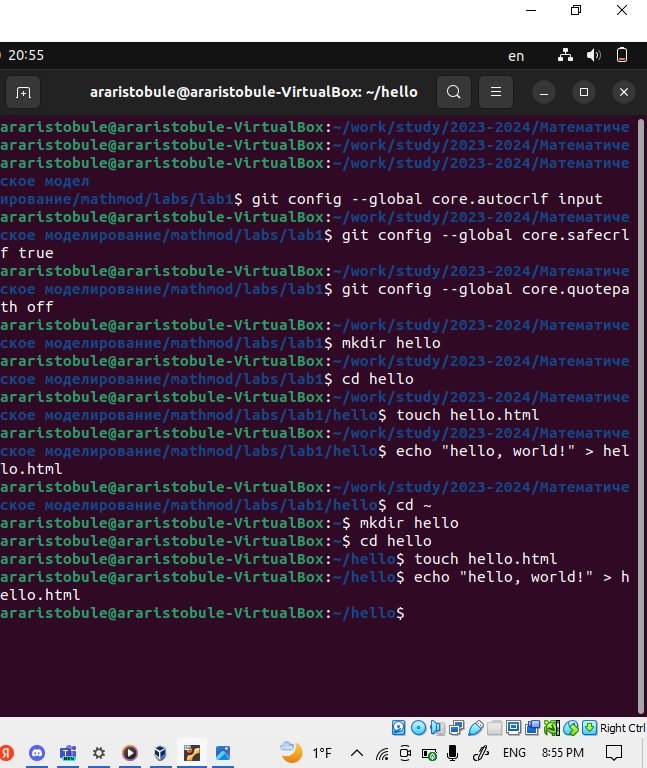


Рис. 2: Новый каталог и файл

Преобразавание документ hello в репозитории. Первый индексирование и первый коммит (рис. 3).

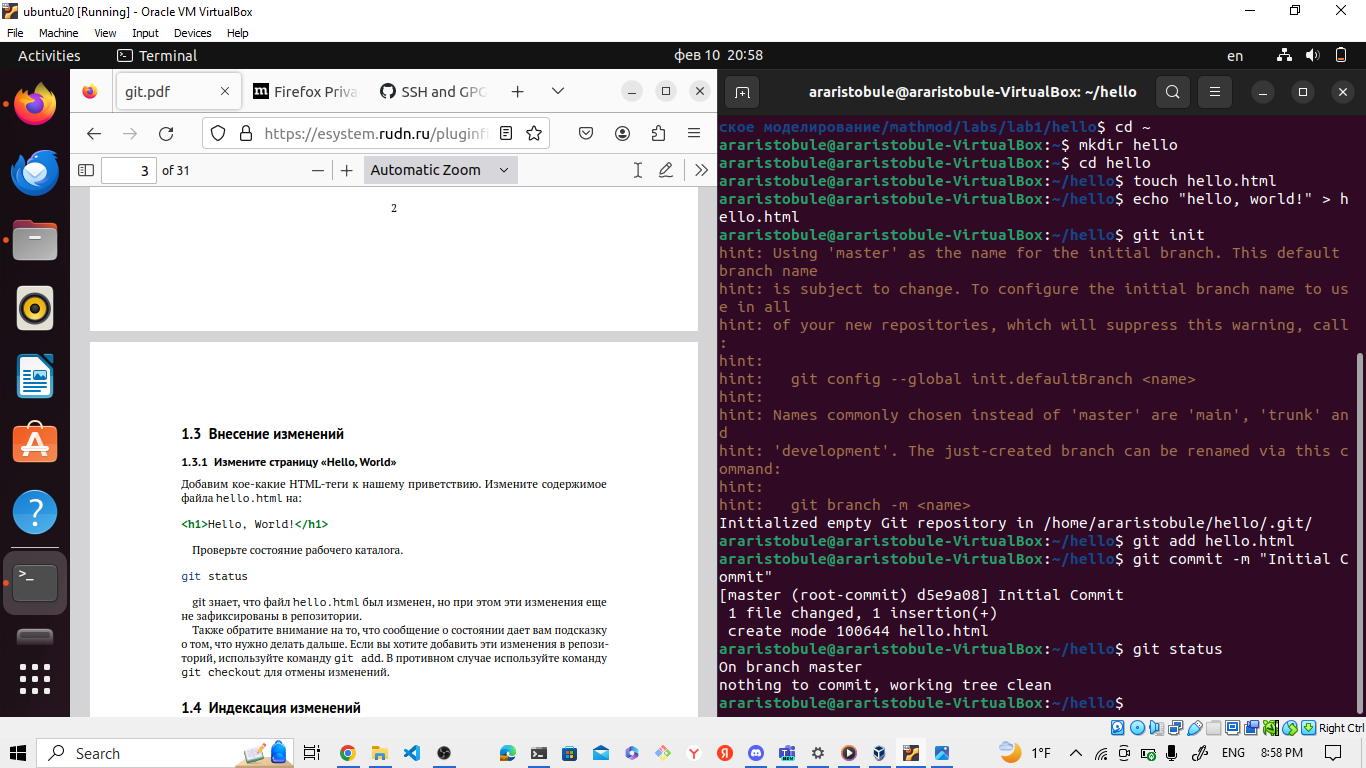


Рис. 3: Новый репозиторый

Когда коммит измениние надо писать комментарии с помощью метки -m иначе editor возникается чтобы там уточнить комментари для коммита. (рис. 4).

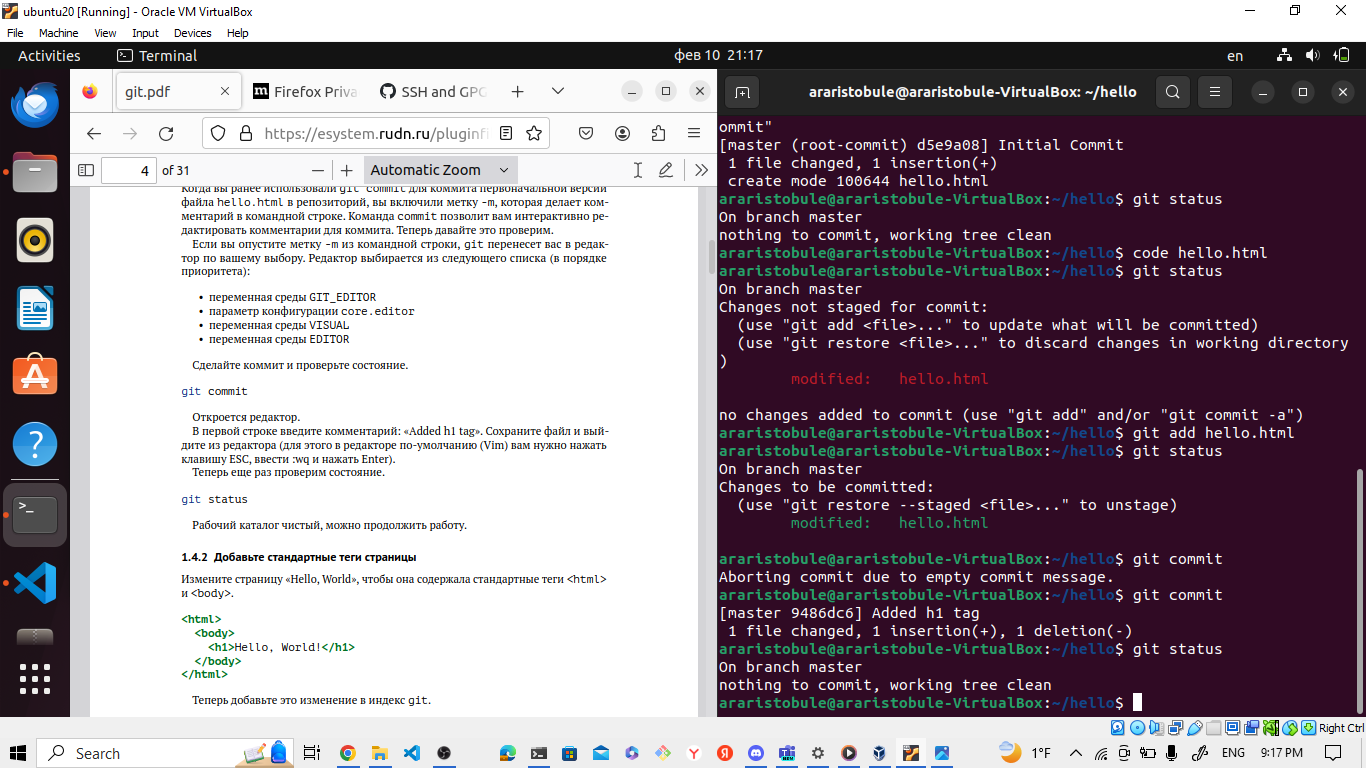


Рис. 4: Коммит без комментарии

Посмотреть все коммити ваше репозитории с помощью git log (рис. 5).

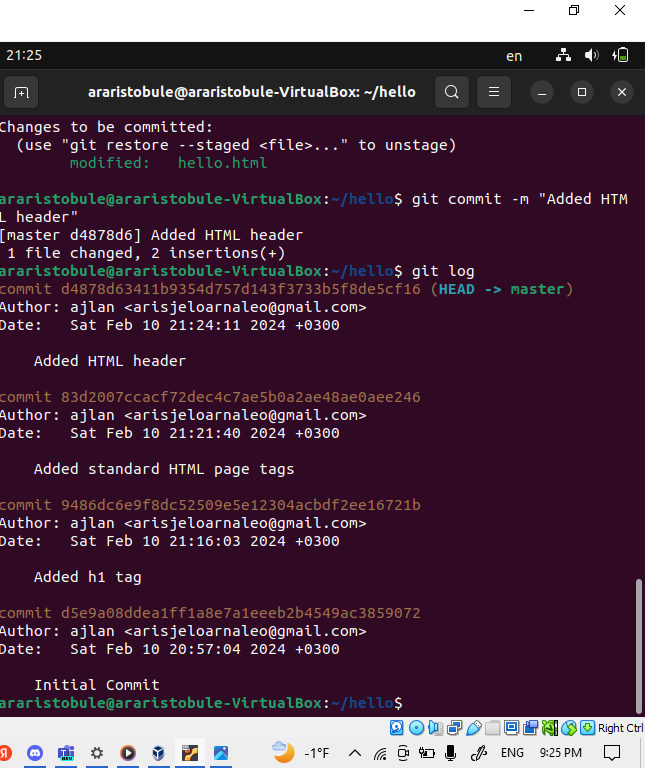


Рис. 5: Git Log

Другой вариант чтобы смотреть списку коммитов с помощью git log –pretty=oneline (рис. 6).

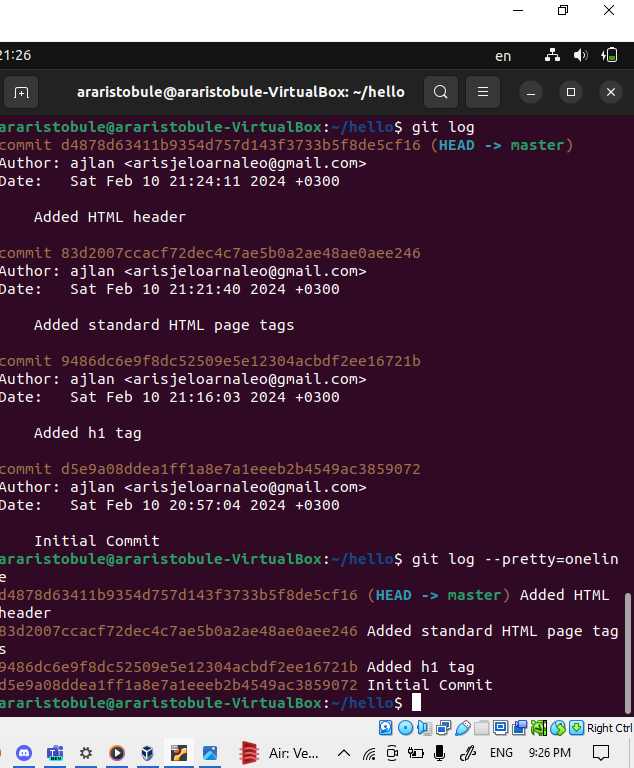


Рис. 6: Списка коммитов

С помощью git checkout , мы можешь перейти в предыдущый коммит. (рис. 7).

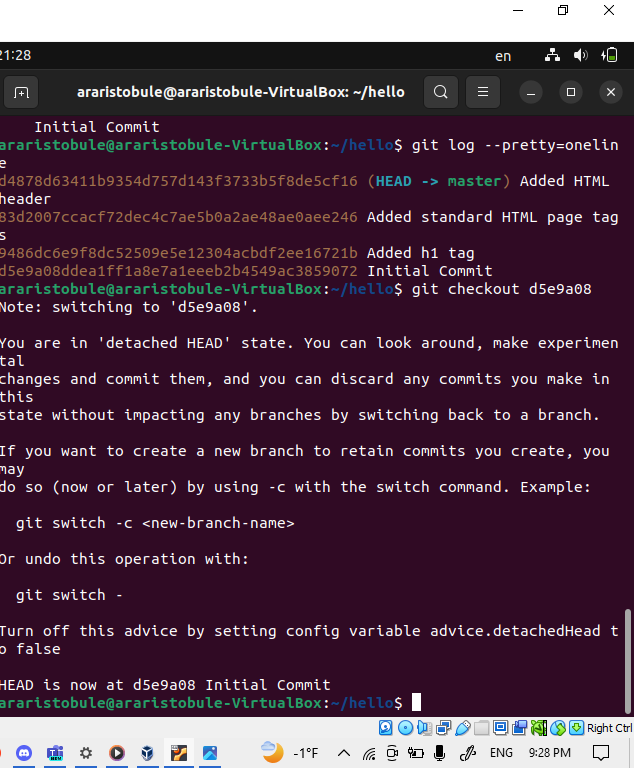


Рис. 7: Git Checkout

И чтобы потом вернуться на наш последный коммит в ветке master, просто набирать git checkout master (рис. 8).

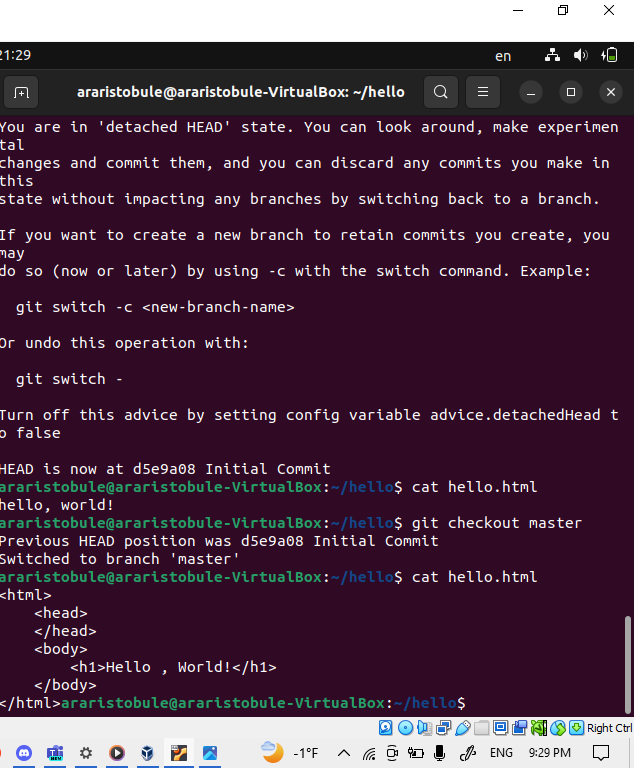


Рис. 8: Последный коммит

Можно исползовать git tap чтобы называть наши коммити. (рис. 9).

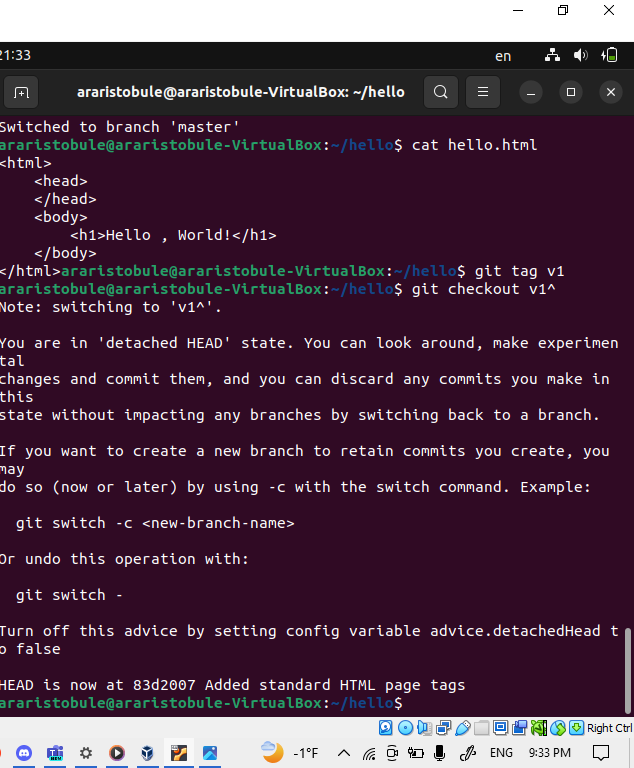


Рис. 9: Tag

Чтобы сбросить из буфера измениние который не собираешь сократить можно използовать git checkout (рис. 10).

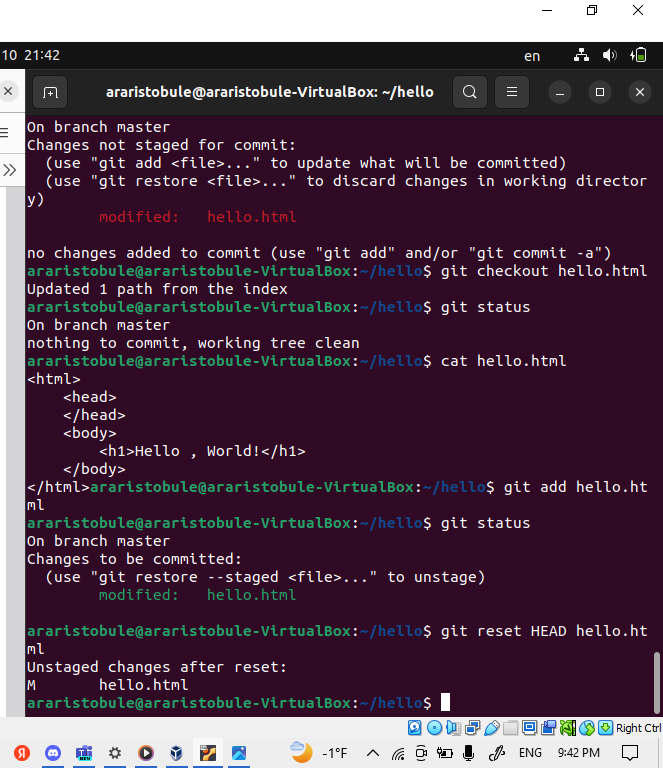


Рис. 10: Удалить из буфера

Чтобы сбросить из индексированных файлов измениние который не собираешь сократить можно използовать git reset HEAD (рис. 11).

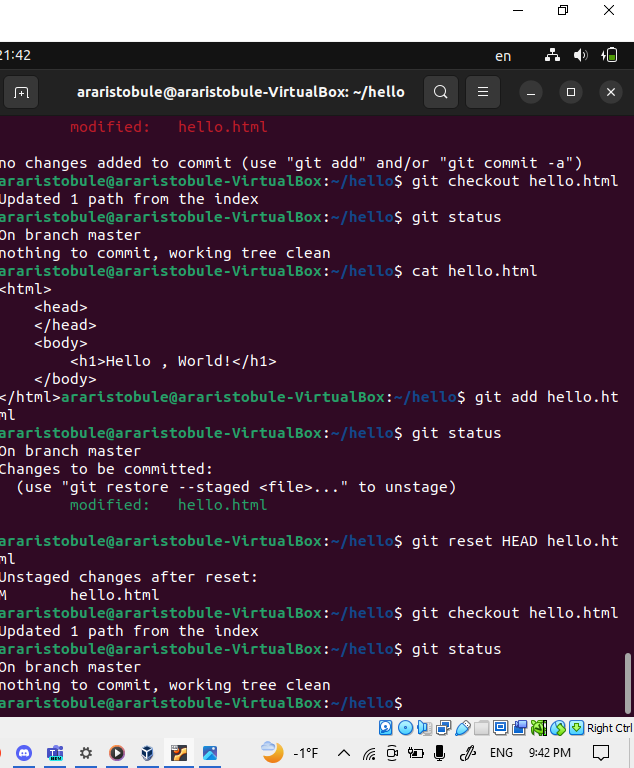


Рис. 11: Удалить из индексированных файлов

Чтобы модифировать последный коммит можно исползовать git commit –amend (рис. 12).

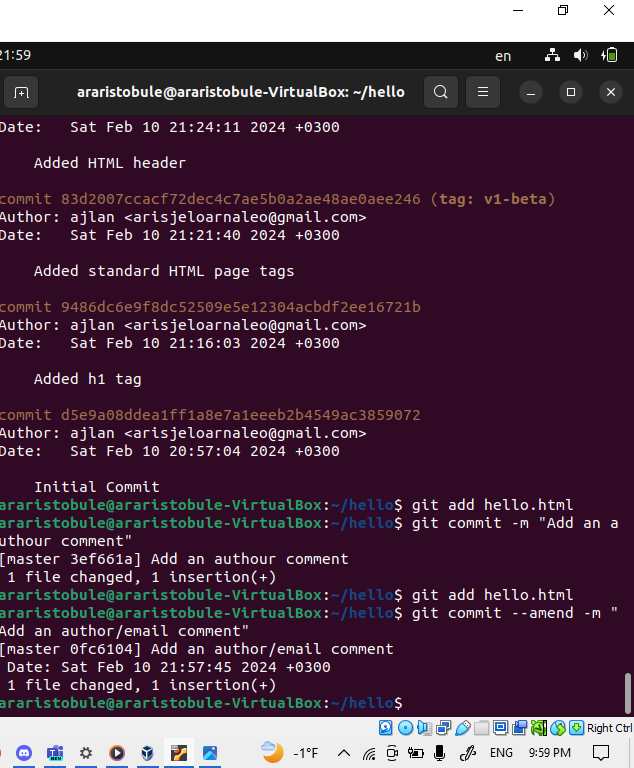


Рис. 12: Модифирование коммита

Чтобы переместить файл в репозитории лучше исползовать git mv (рис. 13).

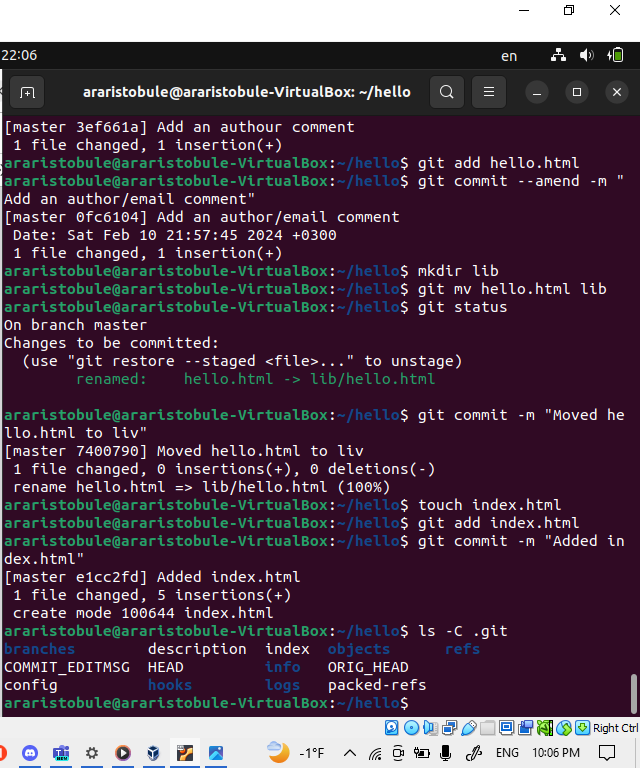


Рис. 13: Переместить файл

Создание новую ветку style с помощбю git checkout -b style (рис. 14).

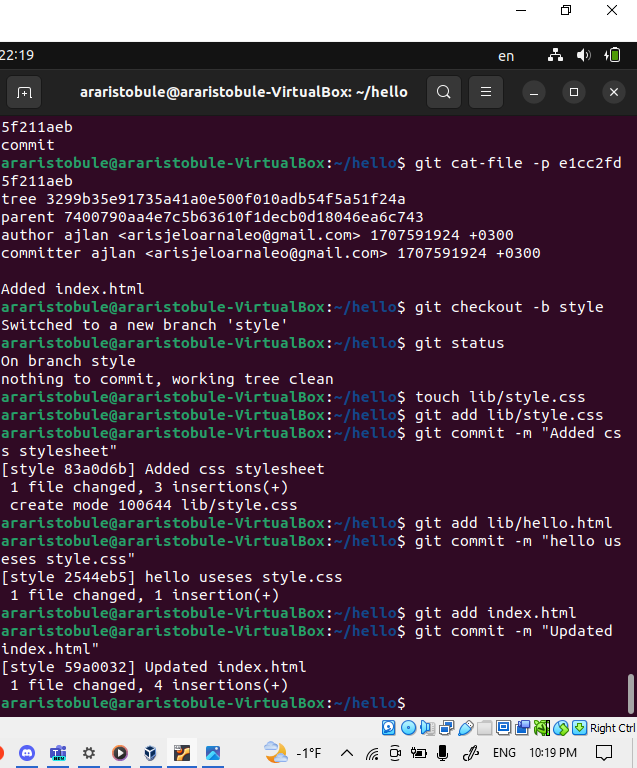


Рис. 14: Ветка style

Слияние ветки с помощью git merge и просмотр эти ветки графический с помощью git log –graph –alд (рис. 15).

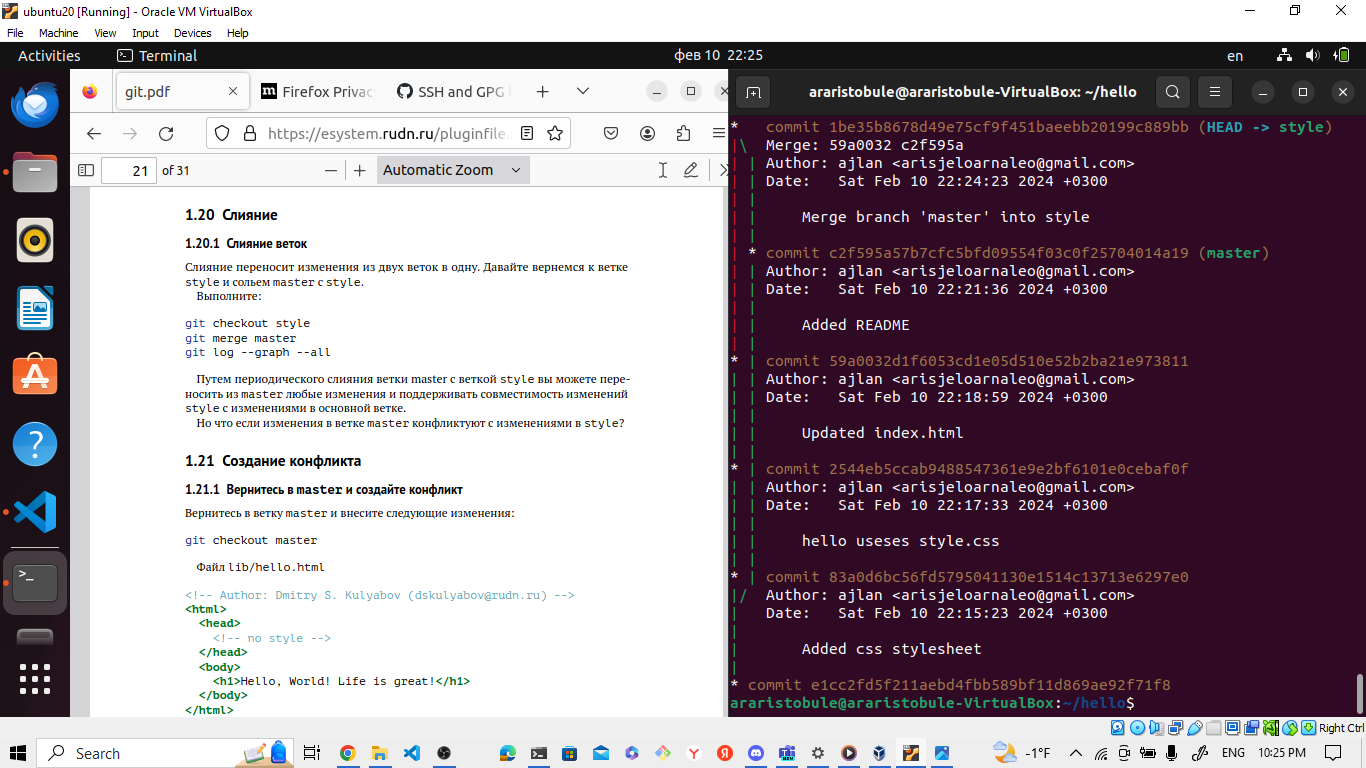


Рис. 15: Слияние ветки

Чтобы rebase измениние ветки можно испозовать команды git Rebase (рис. 16).

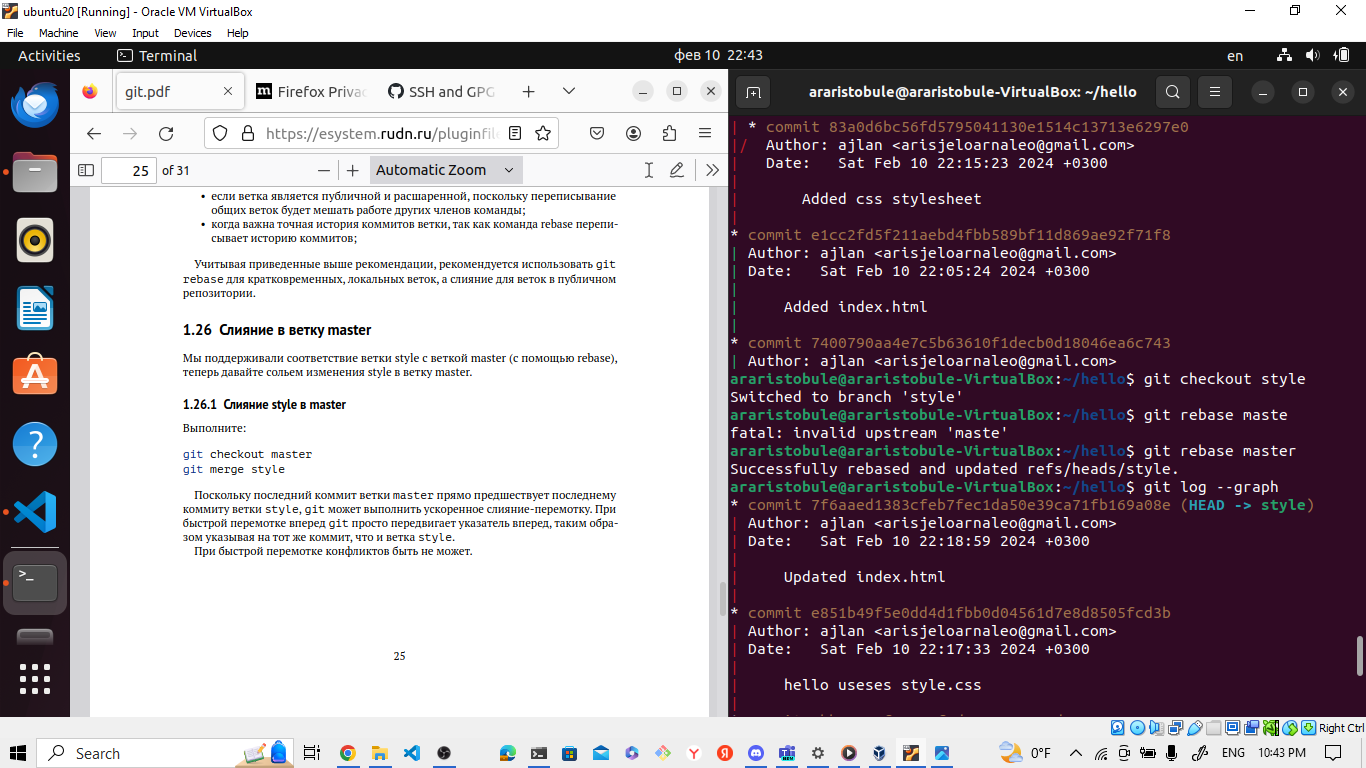


Рис. 16: Rebase a branch

# 4 Выводы

Git Удобно способ чтобы сохранить разные этапы данного проэкта.