Лабораторная работа №1

Математическое моделирование

Юхнин Илья Андреевич

Содержание

# Цель работы

Научиться работе с Git и языком разметки Markdown

# Задание

Создание отчёта по предыдущей лабораторной работе Git в формате Markdown

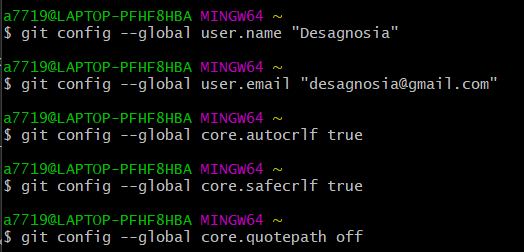
# Теоретическое введение

Git — это система контроля версий, которая позволяет отслеживать и фиксировать изменения в коде: вы можете восстановить код в случае сбоя или откатить до более ранних версий. С Git работают через командную строку или инструменты вроде GitHub. Команды Git принимают вид git , где аргументом может быть путь к файлу. В команды также включаются опции, которые обозначаются как –.

Markdown — это синтаксис (или набор правил), который форматирует текст на веб-страницах. Markdown упрощает форматирование текста для веб-страниц, поскольку теги проще, чем HTML, и они автоматически преобразуются в HTML. Это означает, что вам не нужно знать HTML, чтобы написать что-то для веб-страницы, потому что Markdown переводит ваши теги в HTML для вас. Он охватывает не все возможные теги HTML, а наиболее распространенные параметры форматирования.

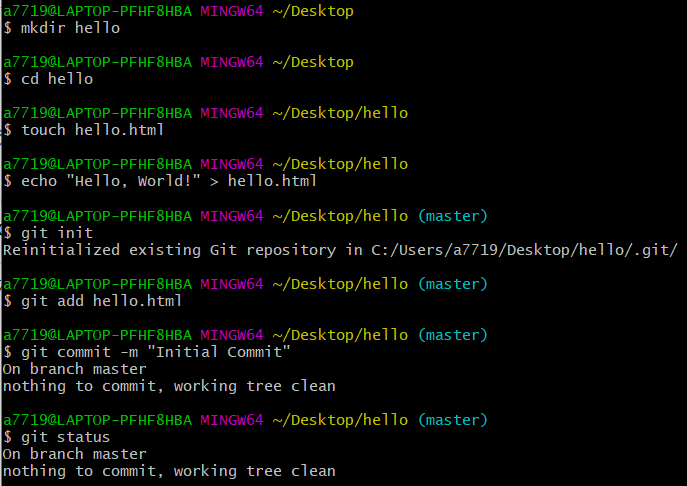
# Выполнение лабораторной работы

1.1 Подготовка работы с git(рис. 1.1)



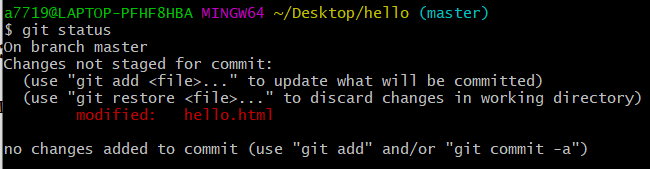
1.1

1.2 Создание проекта и добавление его в репозиторий(рис. 1.2)



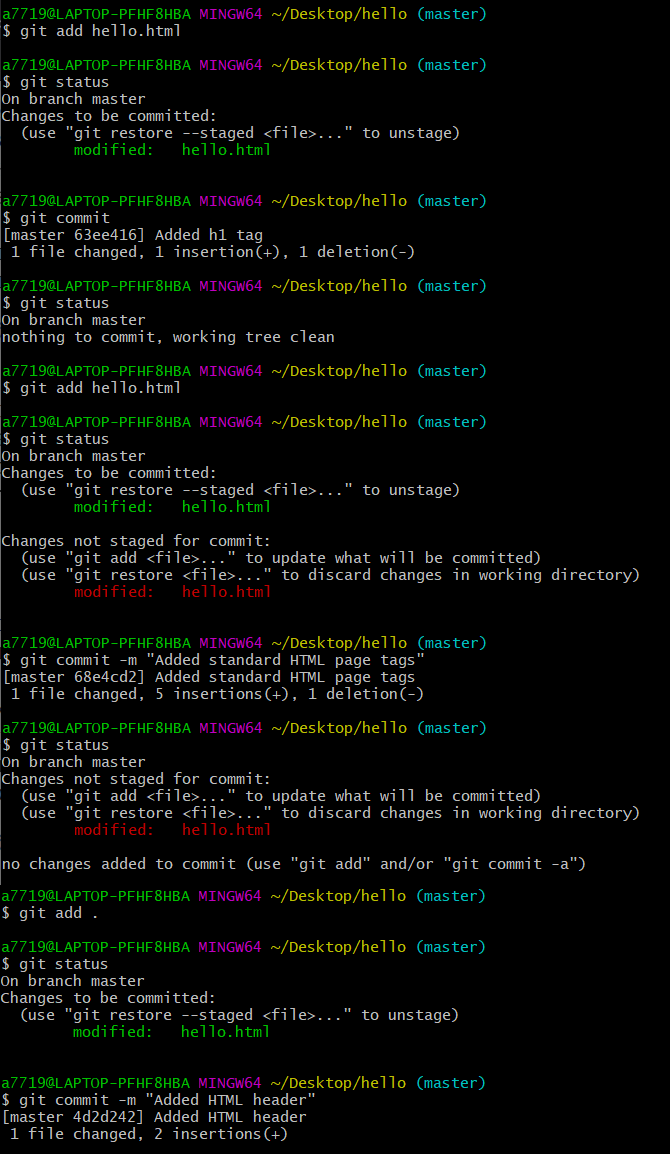
1.2

1.3 Внесение изменений и проверка состояния рабочего каталога(рис. 1.3)

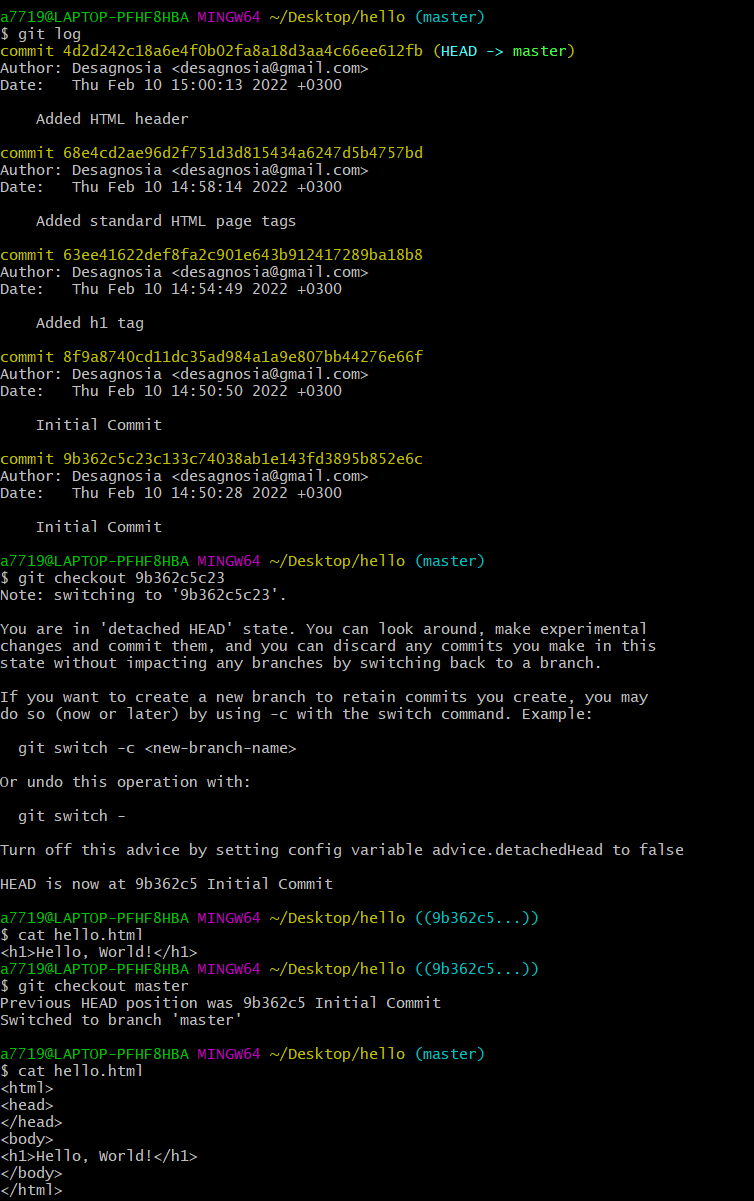


1.3

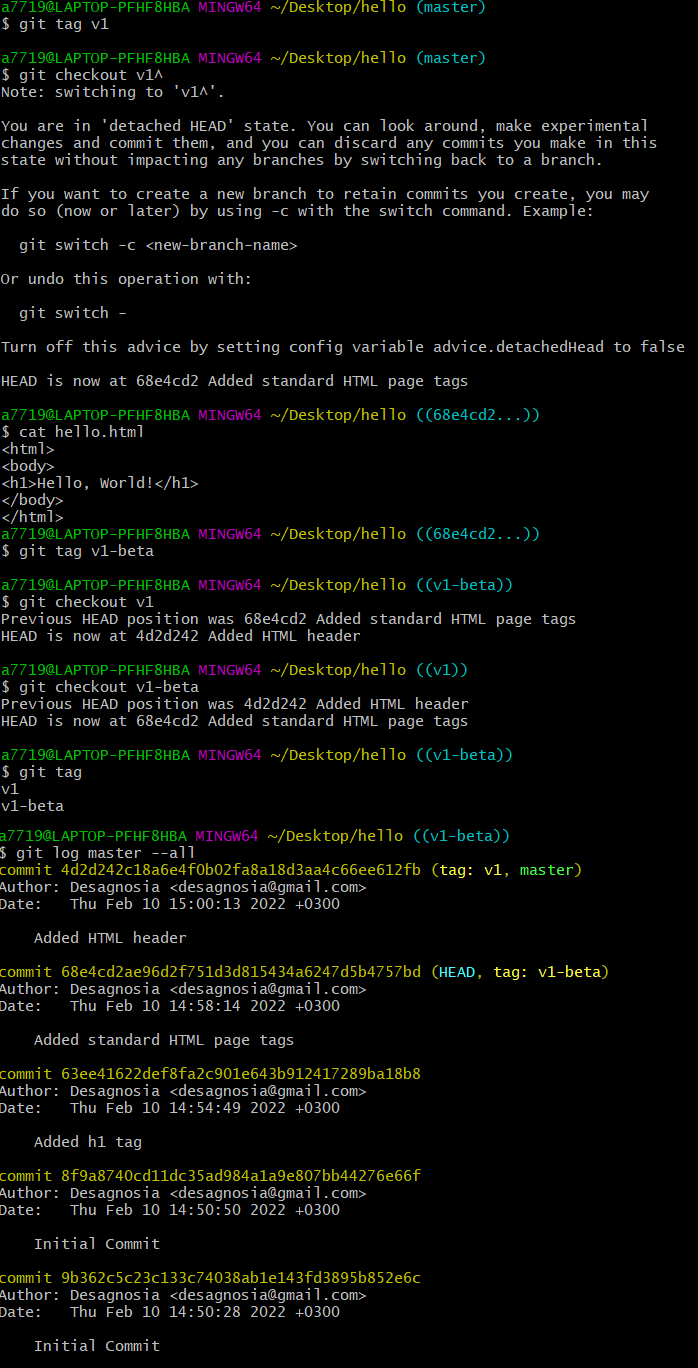
1.4 Индексация изменений и работа с историей и тегами(рис. 1.4.1-3 - 1.4.5-7)



1.4.1-3

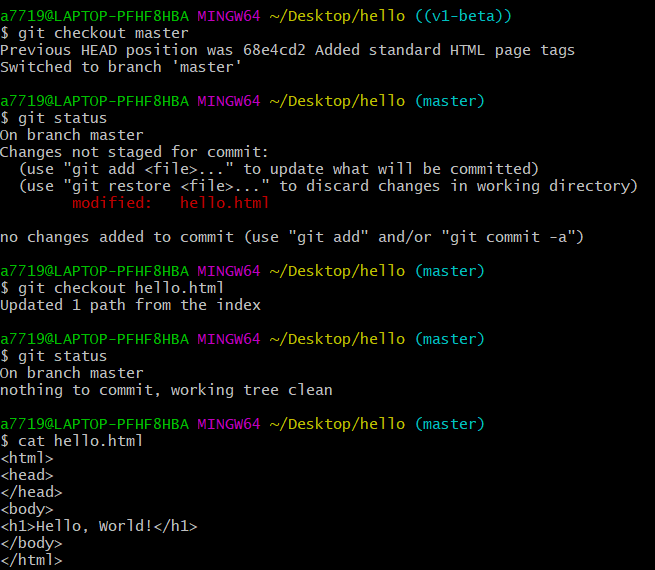


1.4.4



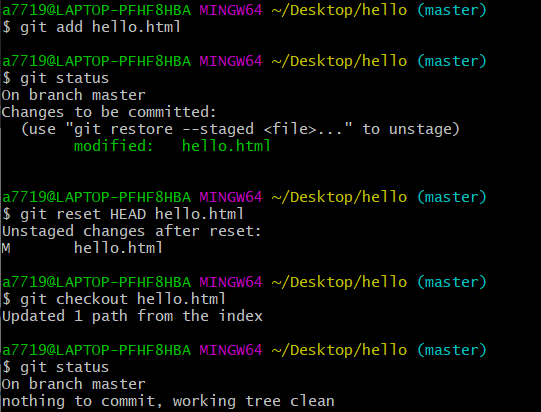
1.4.5-7

1.5 Отмена локальных изменений (до индексации) (рис. 1.5)



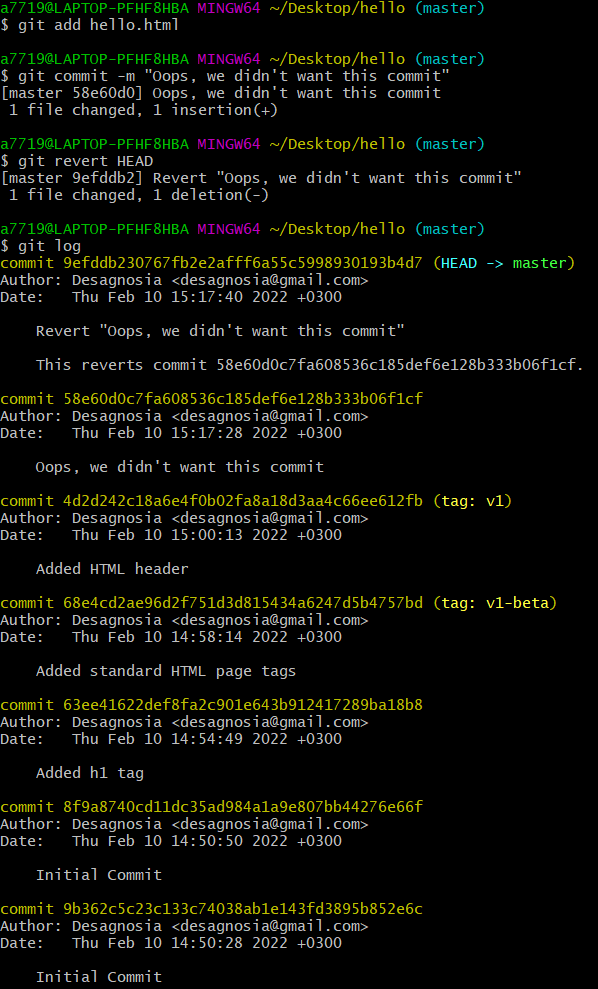
1.5

1.6 Отмена проиндексированных изменений (перед коммитом)(рис. 1.6)



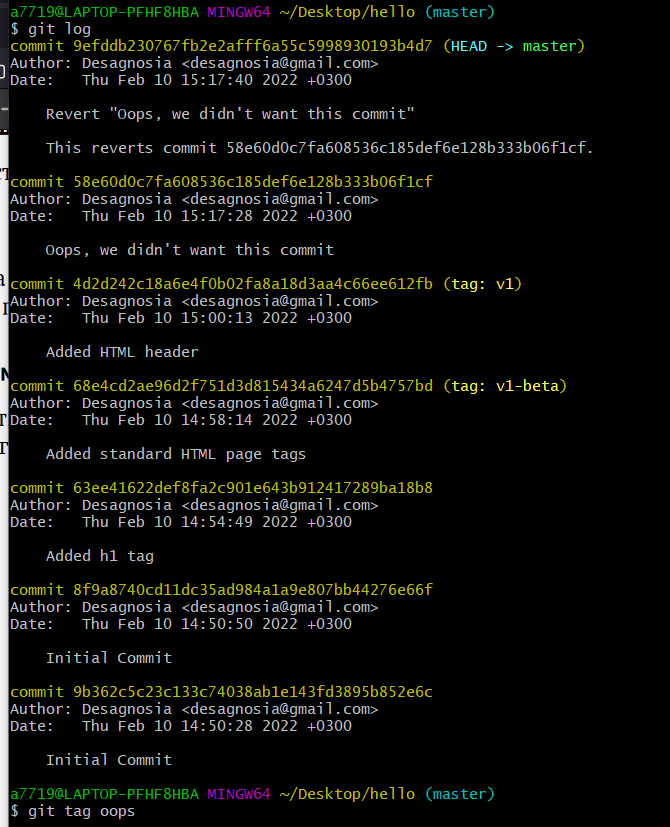
1.6

1.7 Отмена коммитов(рис. 1.7)

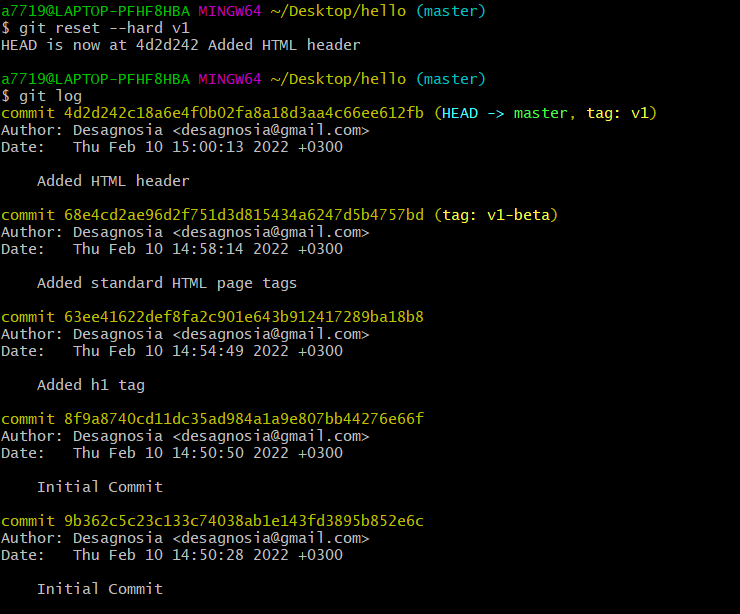


1.7

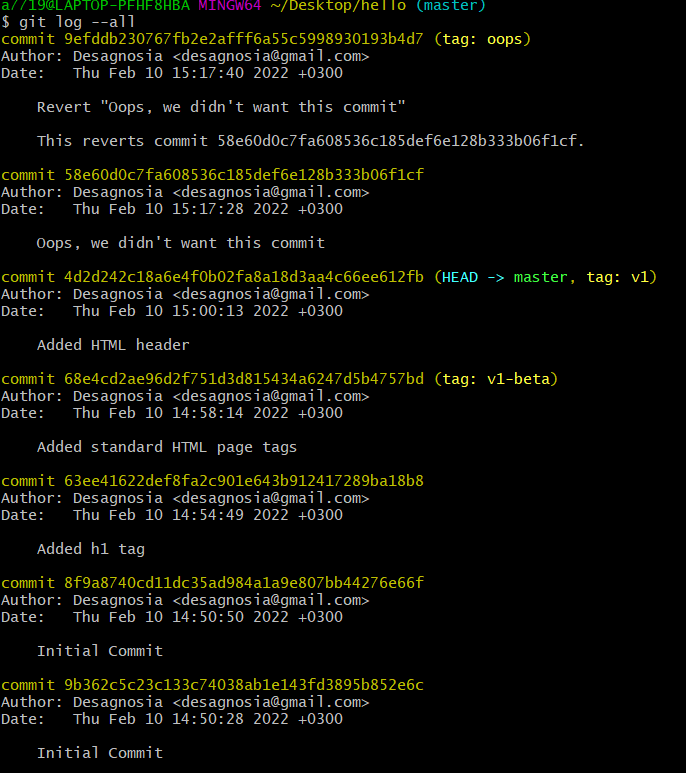
1.8 Удаление коммиттов из ветки(рис. 1.8.2-3 - 1.8.5)



1.8.2-3

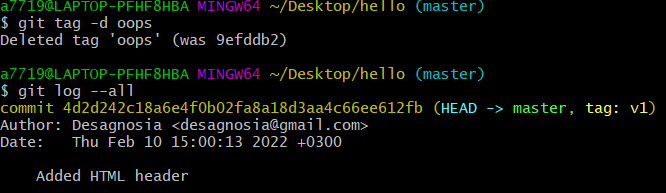


1.8.4



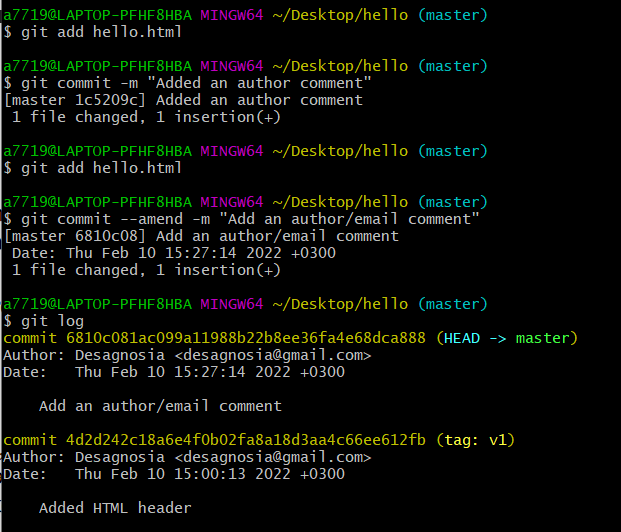
1.8.5

1.9 Удаление тега oops (рис. 1.9)



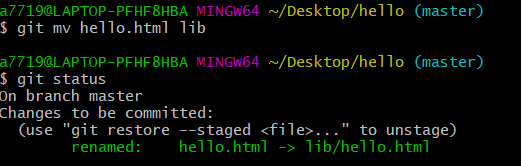
1.9

1.10 Внесение изменений в коммиты(рис. 1.10)



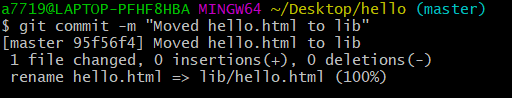
1.10

1.11 Перемещение файлов(рис. 1.11)



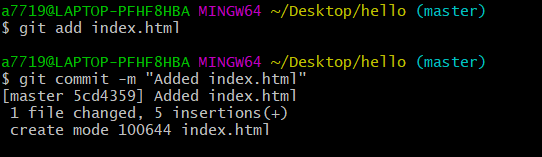
1.11

1.12 Второй способ перемещения файлов(рис. 1.12)



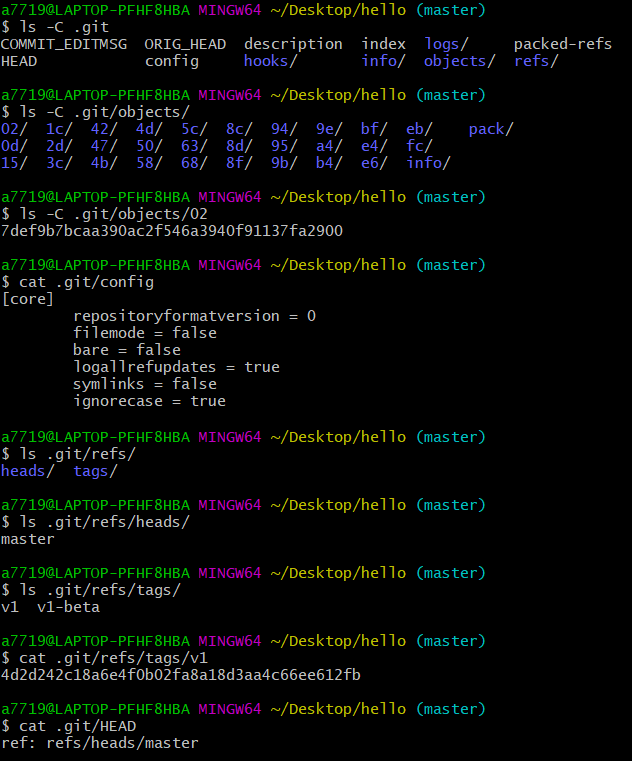
1.12

1.13 Подробнее о структуре(добавление index.html)(рис. 1.13)



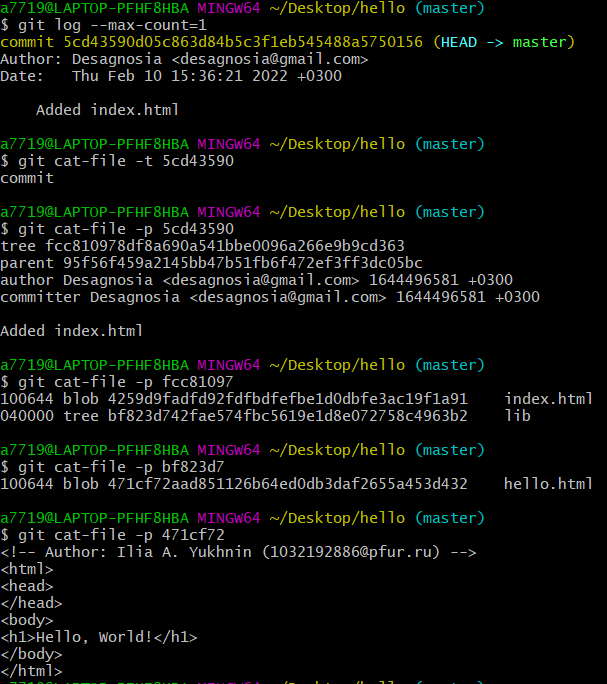
1.13

1.14 Каталог .git(рис. 1.14)

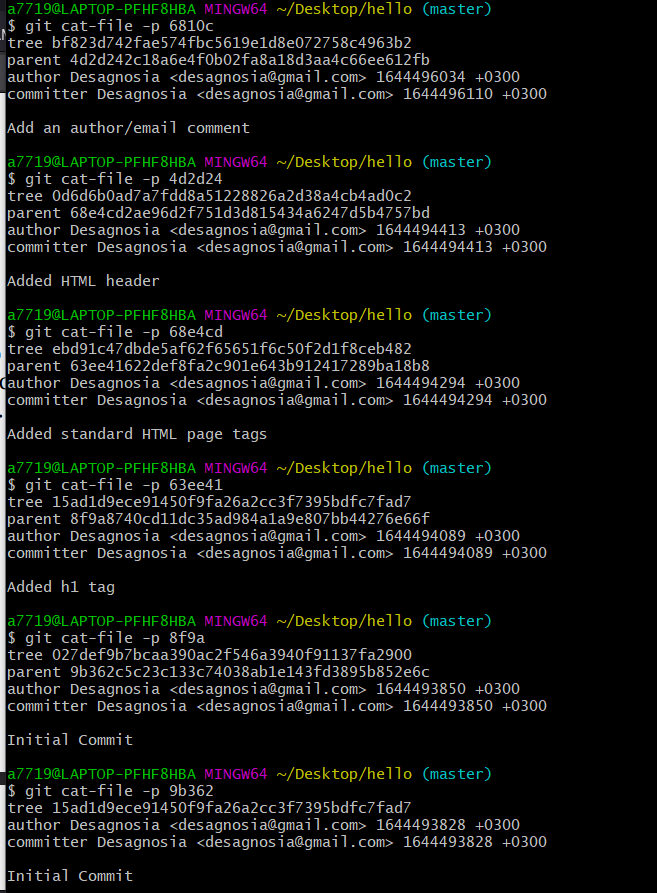


1.14

1.15 Работа непосредственно с объектами git(рис. 1.15а - 1.15b)

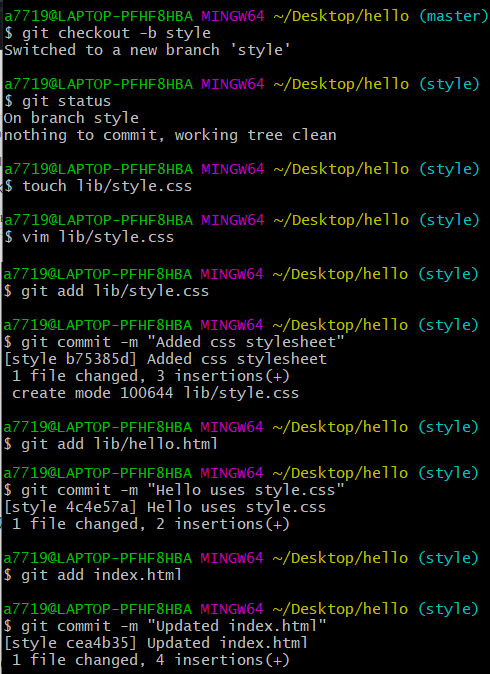


1.15a



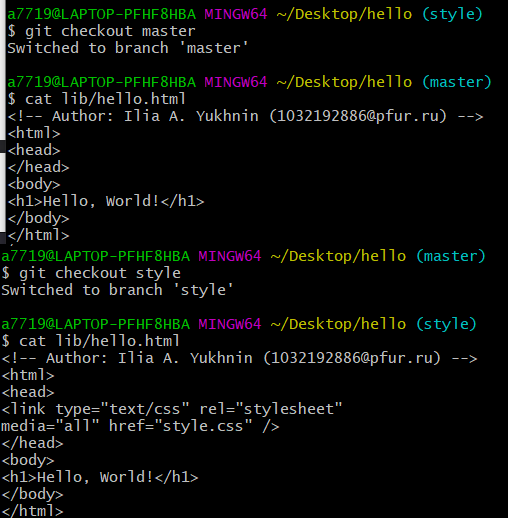
1.15b

1.16 Создание ветки(рис. 1.16)



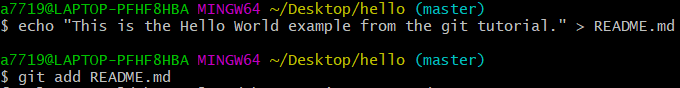
1.16

1.17 Навигация по веткам(рис. 1.17)



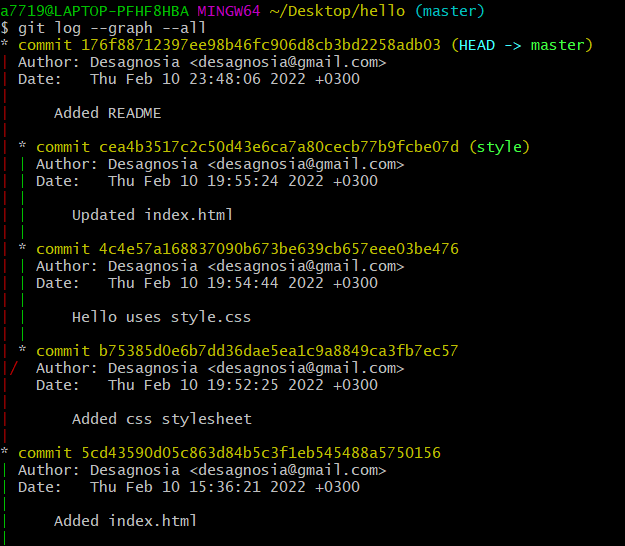
1.17

1.18 Изменения в ветке master(рис. 1.18)



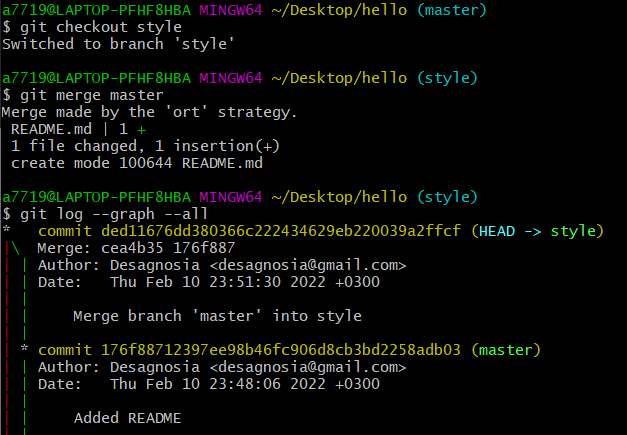
1.18

1.19 Создание коммит изменений README.md в ветку master(рис. 1.19)



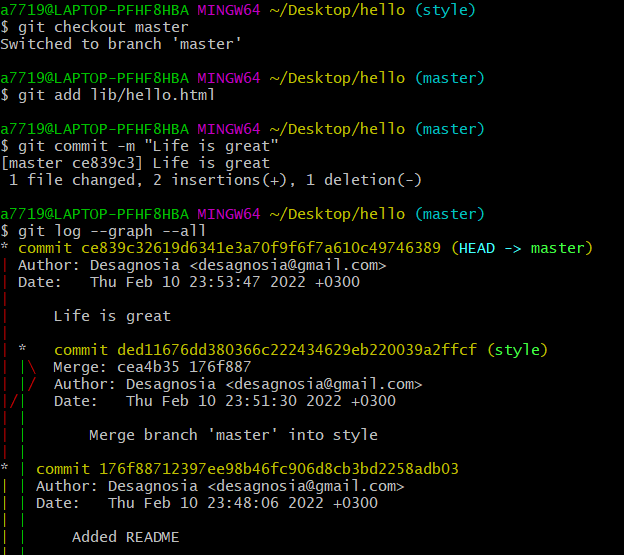
1.19

1.20 Слияние веток(рис. 1.20)



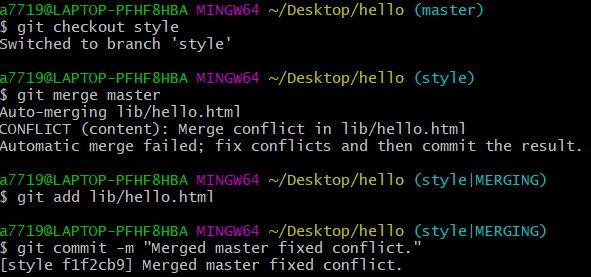
1.20

1.21 Создание конфликта(рис. 1.21)



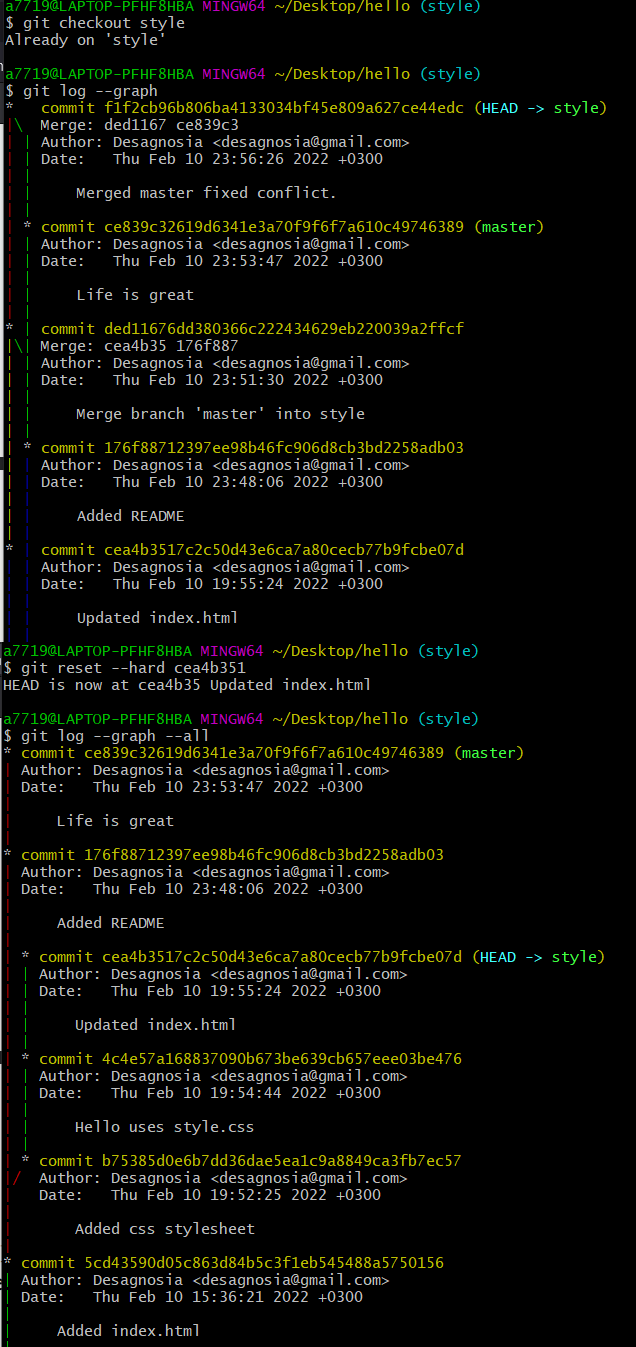
1.21

1.22 Разрешение конфликтов(рис. 1.22)



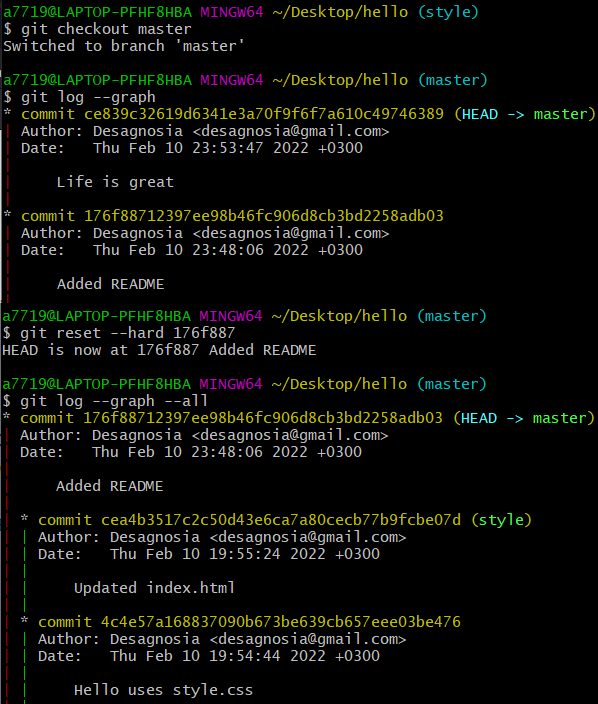
1.22

1.23 Сброс ветки style(рис. 1.23)



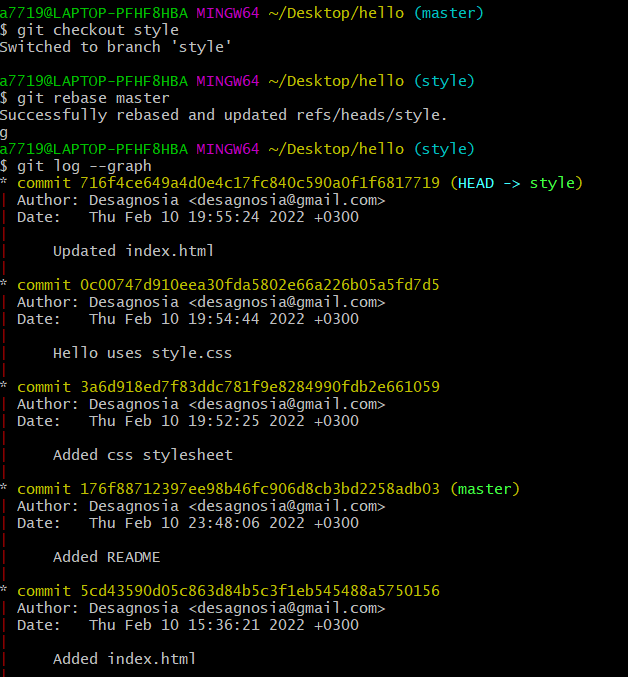
1.23

1.24 Сброс ветки master(рис. 1.24)



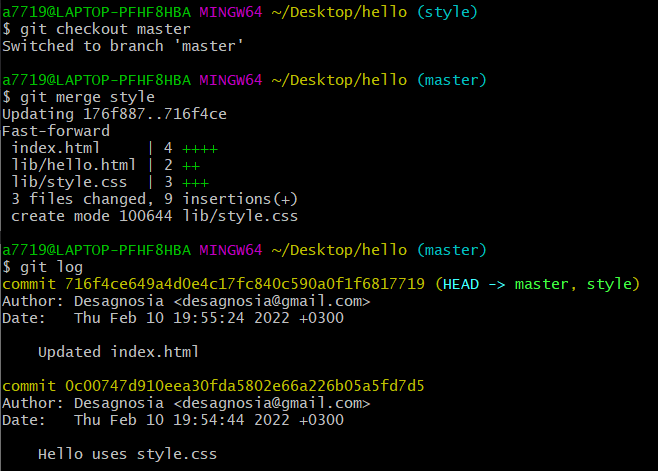
1.24

1.25 Перебазирование(рис. 1.25)



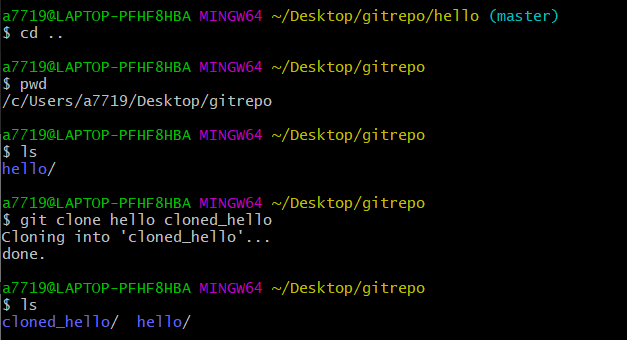
1.25

1.26 Слияние в ветку master(рис. 1.26)



1.26

1.27 Клонирование репозиториев(рис. 1.27)



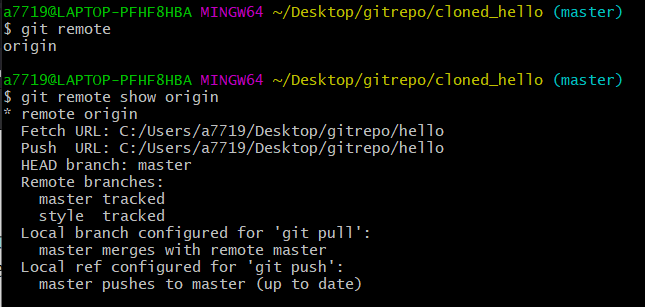
1.27

1.28 Просмотр клонированного репозитория(рис. 1.28)



1.28

1.29 Origin(рис. 1.29)



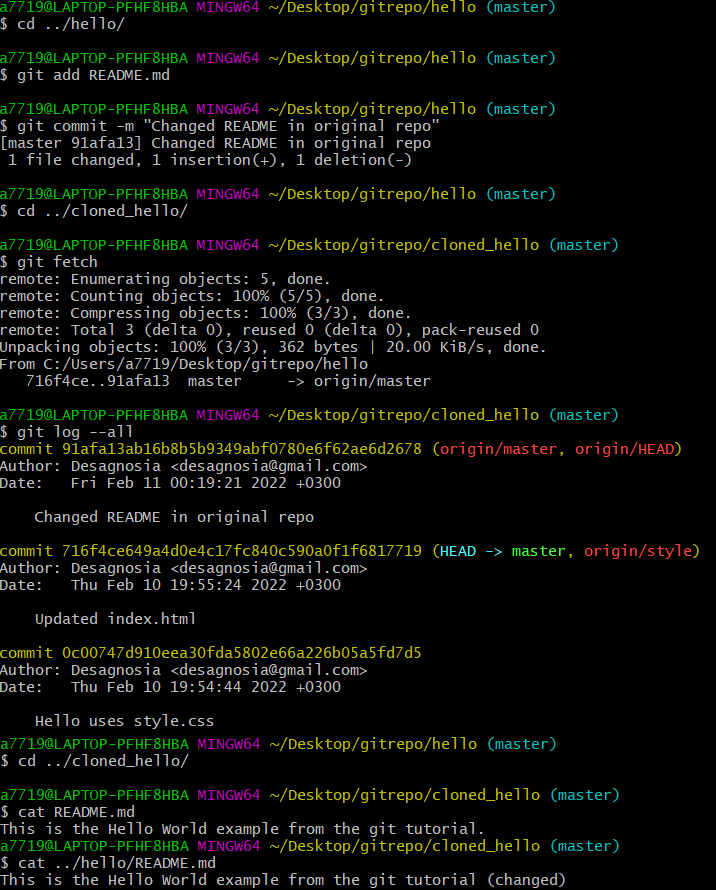
1.29

1.30 Удаленные ветки(рис. 1.30)



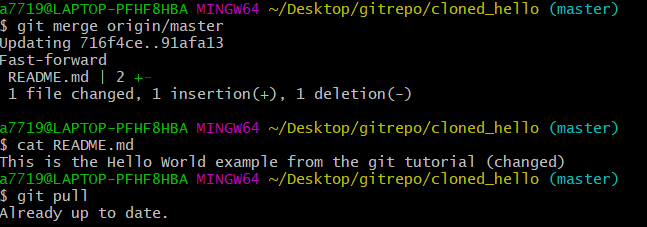
1.30

1.31 Изменение оригинального репозитория(рис. 1.31)



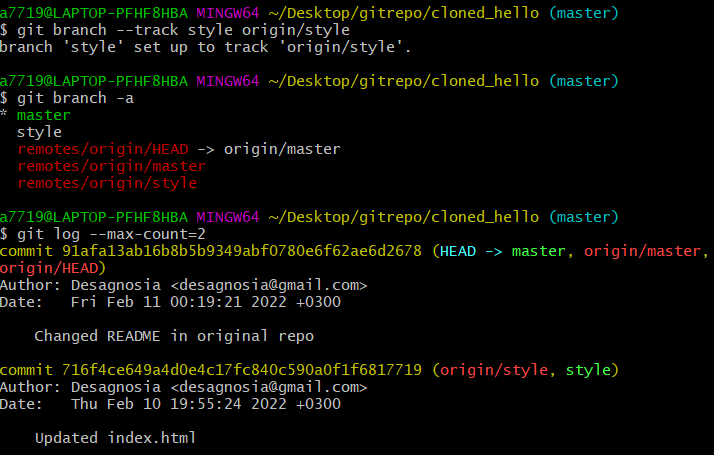
1.31

1.32 Слияние извлеченных изменений(рис. 1.32)



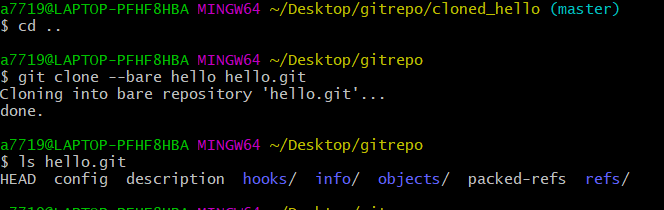
1.32

1.33 Добавление ветки наблюдения(рис. 1.33)



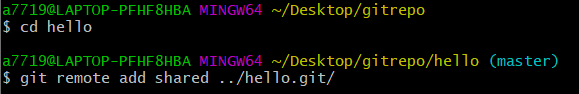
1.33

1.34-1.35 Создание чистого репозитория(рис. 1.35)



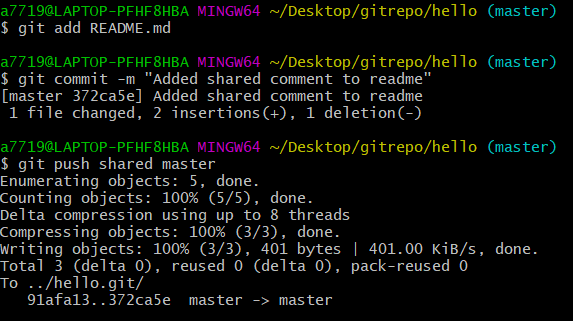
1.35

1.36 Добавление удаленного репозитория(рис. 1.36)



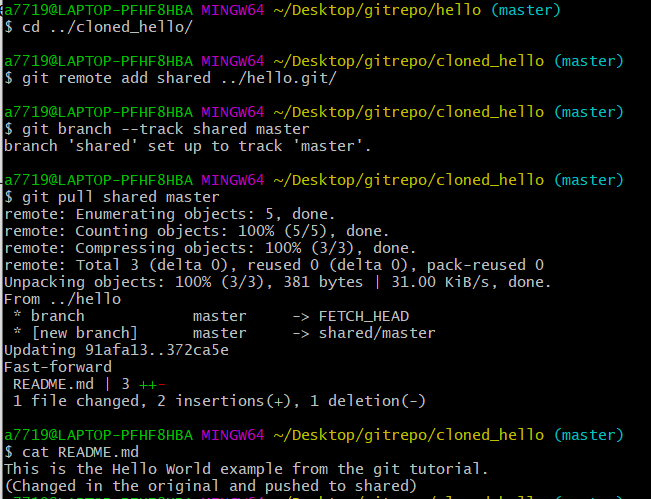
1.36

1.37 Отправка изменений(рис. 1.37)



1.37

1.38 Извлечение общих изменений(рис. 1.38)



1.38

# Выводы

В результате проделанной лабораторной работы, познакомился с работой распределённой системы управления версиям Git, а также с языком разметки Markdown