Лабораторная работа №1

Git

Ким Илья Владиславович

Содержание

# Цель работы

Вспомнить основные функции git и markdown.

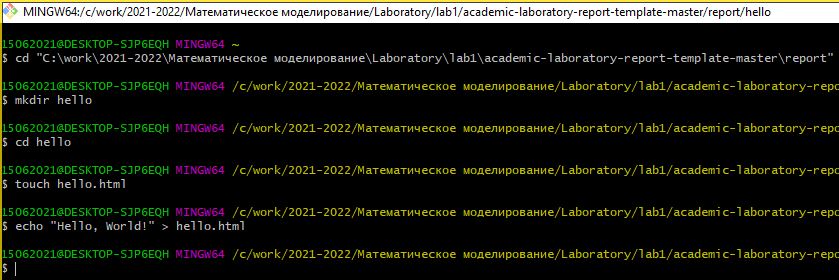
# Задание

* Сделайте отчёт по предыдущей лабораторной работе в формате Markdown.
* В качестве отчёта просьба предоставить отчёты в 3 форматах: pdf, docx и md (в архиве, поскольку он должен содержать скриншоты, Makefile и т.д.)

# Выполнение лабораторной работы

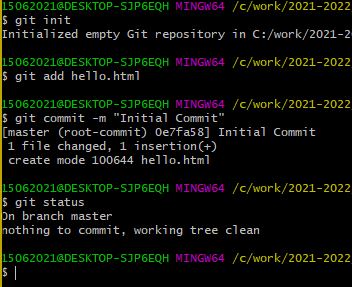
1.2 Создание проекта

1.2.1 Создание страницы "Hello, World" (рис. [-@fig:001])



Hello, World

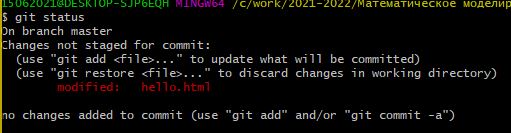
1.2.2 Создание репозитория  
  
1.2.3 Добавление файла в репозиторий  
  
1.2.4 Проверка состояние репозитория (рис. [-@fig:002])



Hello, World

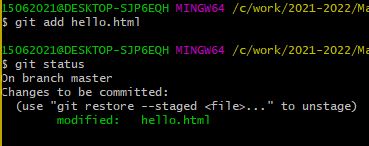
1.3 Внесение изменений

1.3.1 Изменение страницы «Hello, World» (рис. [-@fig:003])



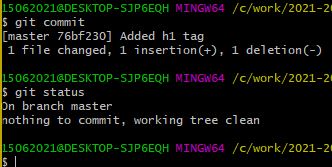
Hello, World

1.4 Индексация изменений (рис. [-@fig:004])



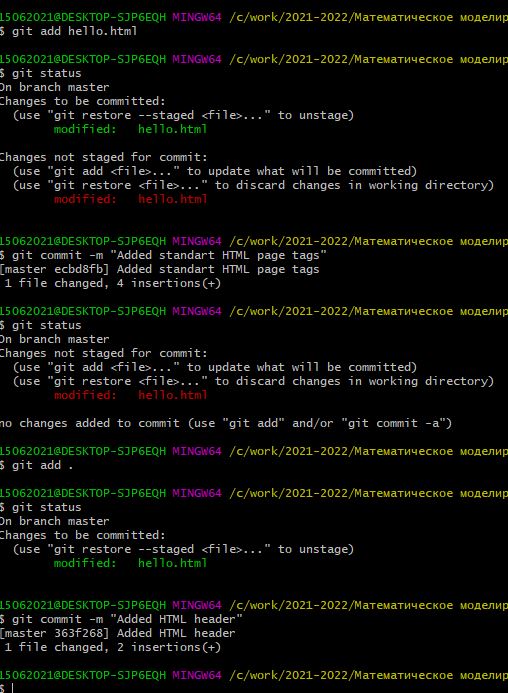
Hello, World

1.4.1 Коммит изменений (рис. [-@fig:005])



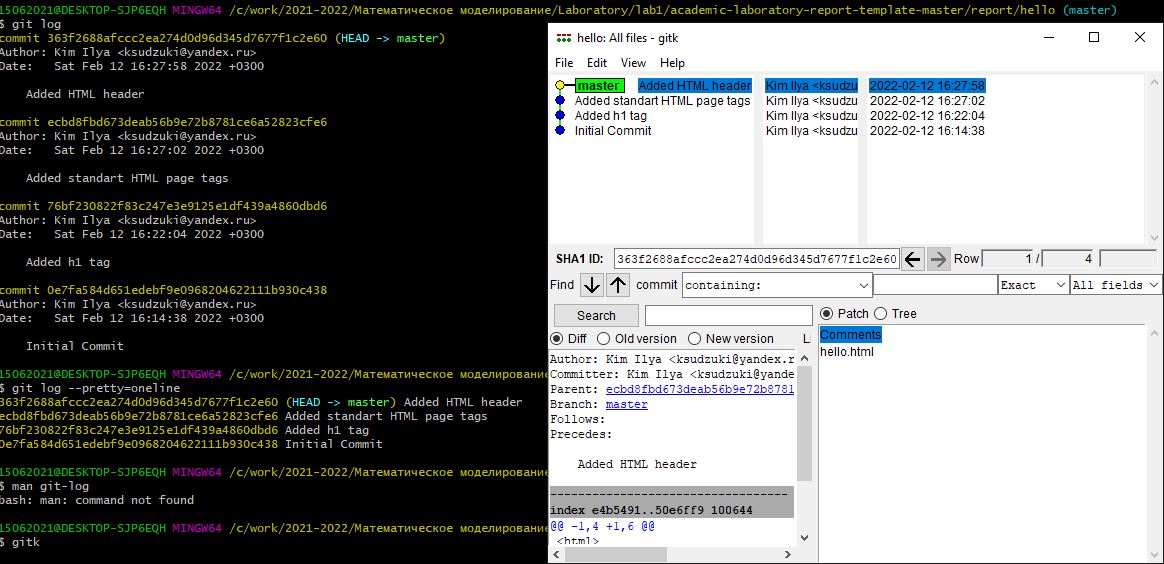
Hello, World

1.4.2 Добавление тегов (рис. [-@fig:006])



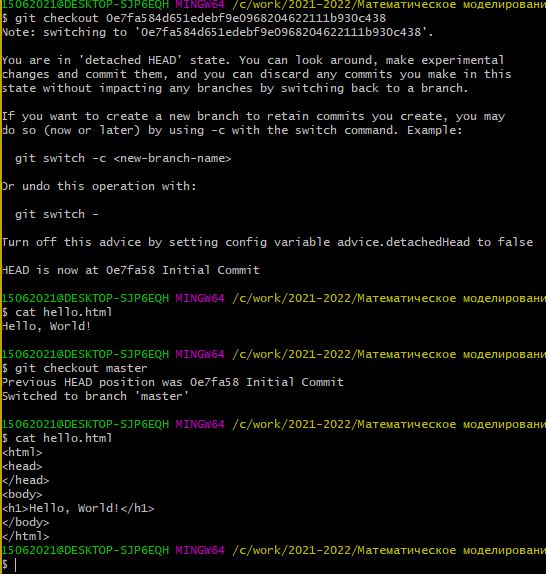
Hello, World

1.4.3 История (рис. [-@fig:007])



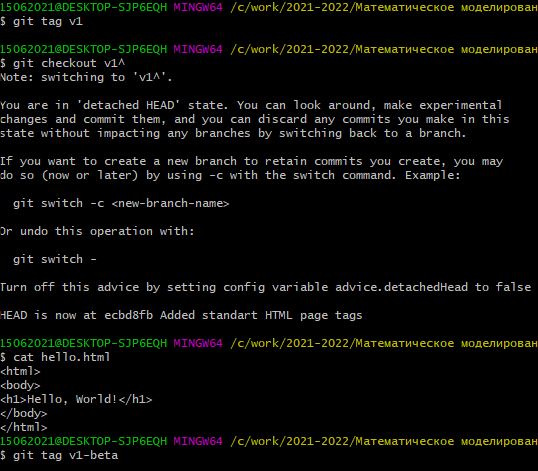
Hello, World

1.4.4 Получение старых версий (рис. [-@fig:008])



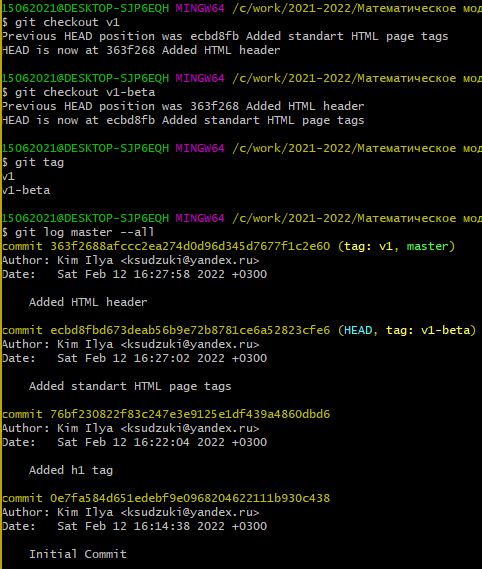
Hello, World

1.4.5 Создание тегов версий (рис. [-@fig:009])



Hello, World

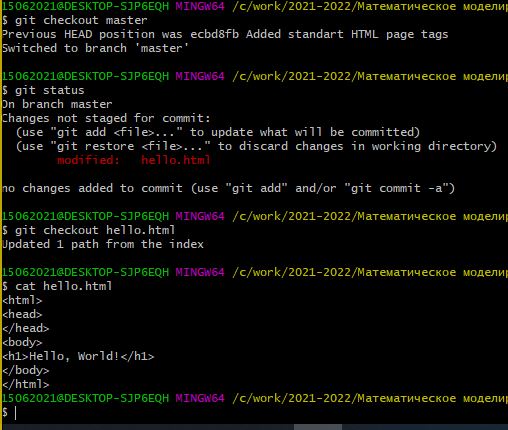
1.4.6 Переключение по имени тега  
  
1.4.7 Просмотр тегов с помощью команды tag (рис. [-@fig:010])



Hello, World

1.5 Отмена локальных изменений (до индексации)

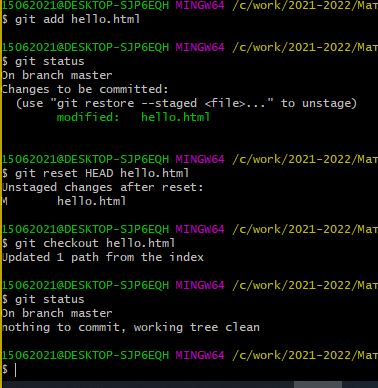
1.5.1 Переключитесь на ветку master  
  
1.5.2 Измените hello.html  
  
1.5.3 Проверьте состояние  
  
1.5.4 Отмена изменений в рабочем каталоге (рис. [-@fig:011])



Hello, World

1.6 Отмена проиндексированных изменений (перед коммитом)

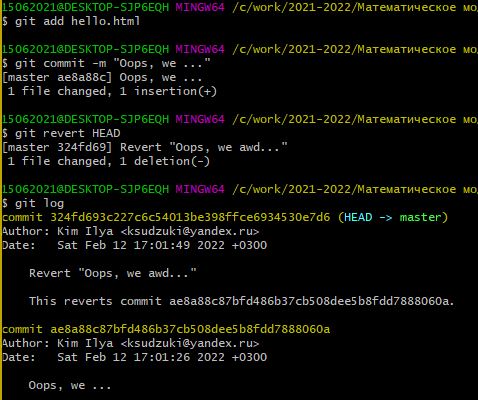
1.6.1 Измените файл и проиндексируйте изменения  
  
1.6.2 Проверьте состояние  
  
1.6.3 Выполните сброс буферной зоны  
  
1.6.4 Переключитесь на версию коммита (рис. [-@fig:012])



Hello, World

1.7 Отмена коммитов

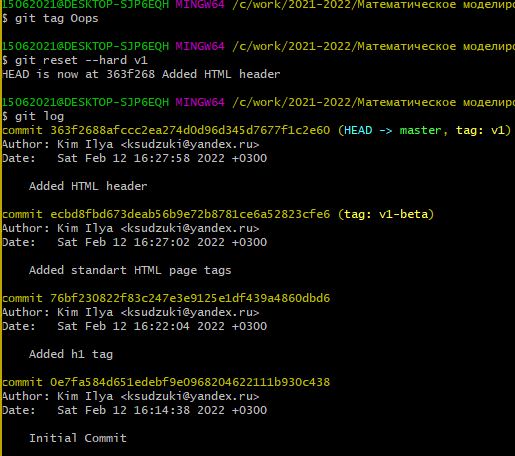
1.7.1 Отмена коммитов  
  
1.7.2 Измените файл и сделайте коммит  
  
1.7.3 Сделайте коммит с новыми изменениями, отменяющими предыдущие  
  
1.7.4 Проверьте лог (рис. [-@fig:013])



Hello, World

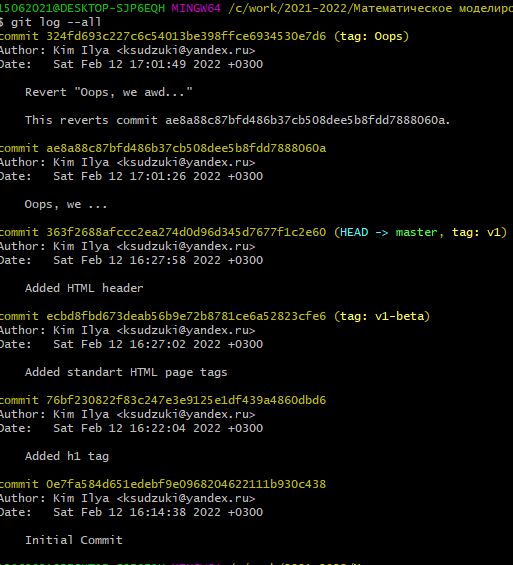
1.8 Удаление коммиттов из ветки

1.8.1 Команда git reset  
  
1.8.2 Проверьте нашу историю  
  
1.8.3 Для начала отметьте эту ветку  
  
1.8.4 Сброс коммитов к предшествующим коммиту Oops (рис. [-@fig:014])



Hello, World

1.8.5 Ничего никогда не теряется (рис. [-@fig:015])

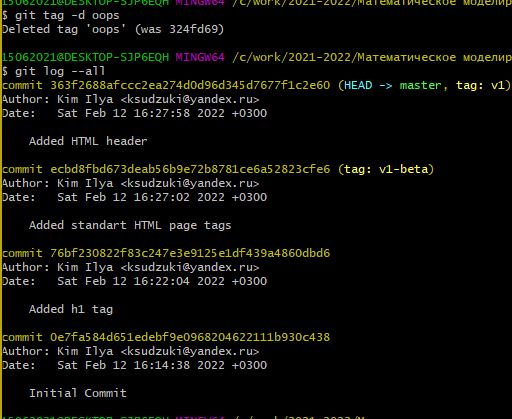


Hello, World

1.8.6 Опасность сброса

1.9 Удаление тега oops

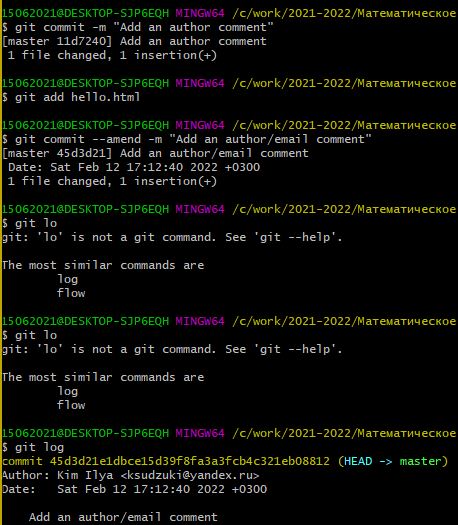
1.9.1 Удаление тега oops (рис. [-@fig:016])



Hello, World

1.10 Внесение изменений в коммиты

1.10.1 Измените страницу, а затем сделайте коммит  
  
1.10.3 Измените предыдущий коммит  
  
1.10.4 Просмотр истории (рис. [-@fig:017])



Hello, World

1.11 Перемещение файлов

1.11.1 Переместите файл hello.html в каталог lib

1.12 Второй способ перемещения файлов

1.12.1 Коммит в новый катало

1.13 Подробнее о структуре

1.13.1 Добавление index.html

1.14 Git внутри: Каталог .git

1.14.1 Каталог .git  
  
1.14.2 База данных объектов  
  
1.14.3 Углубляемся в базу данных объектов  
  
1.14.4 Config File  
  
1.14.5 Ветки и теги  
  
1.14.6 Файл HEAD

1.15 Работа непосредственно с объектами git

1.15.1 Поиск последнего коммита  
  
1.15.2 Вывод последнего коммита с помощью SHA1 хэша  
  
1.15.3 Поиск дерева  
  
1.15.4 Вывод каталога lib  
  
1.15.5 Вывод файла hello.html  
  
1.15.6 Исследуйте самостоятельно

1.16 Создание ветки

1.16.1 Создайте ветку  
  
1.16.2 Добавьте файл стилей style.css  
  
1.16.3 Измените основную страницу  
  
1.16.4 Измените index.html

1.17 Навигация по веткам

1.17.1 Переключение на ветку master  
  
1.17.2 Вернемся к ветке style

1.18 Изменения в ветке master

1.18.1 Создайте файл README в ветке master

1.19 Сделайте коммит изменений README.md в ветку master.

1.19.1 Просмотр отличающихся веток  
  
1.19.2 Просмотрите текущие ветки

1.20 Слияние

1.20.1 Слияние веток

1.21 Создание конфликта

1.21.1 Вернитесь в master и создайте конфликт  
  
1.21.2 Просмотр веток

1.22 Разрешение конфликтов

1.22.1 Слияние master с веткой style  
  
1.22.2 Решение конфликта  
  
1.22.3 Сделайте коммит решения конфликта  
  
1.22.4 Перебазирование как альтернатива слиянию

1.23 Сброс ветки style

1.23.1 Сброс ветки style  
  
1.23.2 Проверьте ветку.

1.24 Сброс ветки master

1.24.1 Сброс ветки master

1.25 Перебазирование

1.25.1 Слияние VS перебазирование

1.26 Слияние в ветку master

1.26.1 Слияние style в master  
  
1.26.2 Просмотрите логи

1.27 Клонирование репозиториев

1.27.1 Перейдите в рабочий каталог  
  
1.27.2 Создайте клон репозитория hello

1.28 Просмотр клонированного репозитория

1.28.1 Давайте взглянем на клонированный репозиторий.  
  
1.28.2 Просмотрите историю репозитория  
  
1.28.3 Удаленные ветки

1.29 Что такое origin?

1.30 Удаленные ветки

1.30.1 Список удаленных веток

1.31 Изменение оригинального репозитория

1.31.1 Внесите изменения в оригинальный репозиторий hello  
  
1.31.2 Извлечение изменений  
  
1.31.3 Проверьте README.md

1.32 Слияние извлеченных изменений

1.32.1 Слейте извлеченные изменения в локальную ветку master  
  
1.32.2 Еще раз проверьте файл README.md

1.33 Добавление ветки наблюдения

1.33.1 Добавьте локальную ветку, которая отслеживает удаленную ветку

1.34 Чистые репозитории

1.35 Создайте чистый репозиторий

1.36 Добавление удаленного репозитория

1.37 Отправка изменений

1.38 Извлечение общих изменений

# Выводы

Вспомнили как использовать основные функции Git.