# \_УП\_7

# Работа с Git

Для того, чтобы просмотреть список настроенных удалённых репозиториев, необходимо запустить команду git remote (рисунок 1).

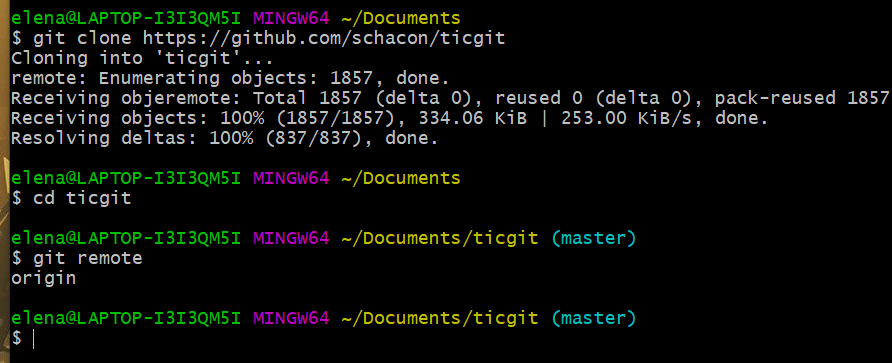


Рисунок 1 – Клонирование репозитория и просмотр удаленных репозиториев

Можно указать ключ -v, чтобы просмотреть адреса для чтения и записи, привязанные к репозиторию (рисунок 2).

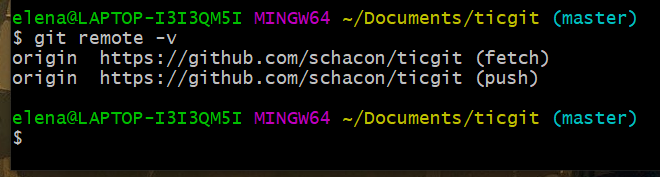


Рисунок 2 - Просмотр удаленных репозиториев с ключом -v

Для добавления удаленного репозитория с новым именем используется команда git remote add (рисунок 3).

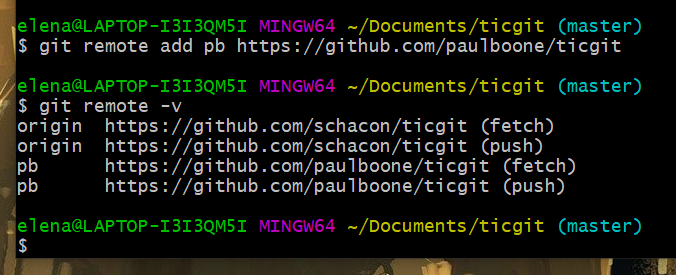


Рисунок 3 - Добавление удаленного репозитория

После задания имени репозиторию впоследствии его можно использовать вместо указания полного пути (рисунок 4).

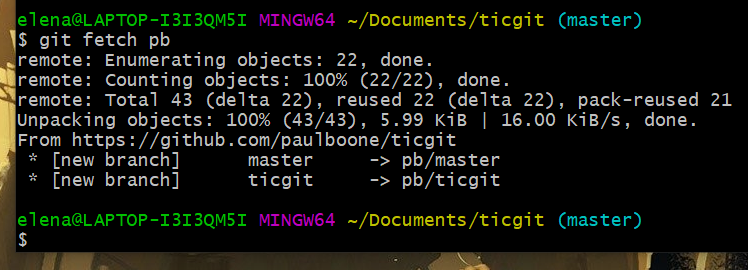


Рисунок 4 - Использование имени вместо пути

Для получения данных из удалённых проектов используется команда git fetch (рисунок 5).

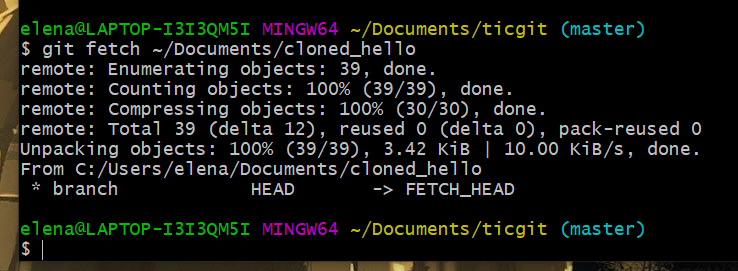


Рисунок 5 - Получение данных из удаленных проектов

Для отправки изменений в удаленный репозиторий используется команда git push (рисунок 6).

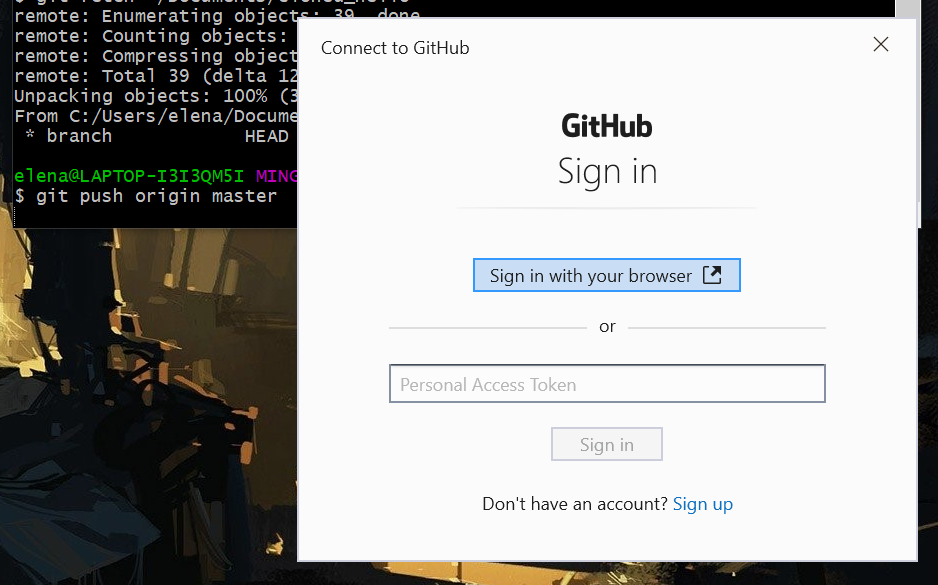


Рисунок 6 - Отправка изменений в удаленный репозиторий

Для получения информации об одном из удалённых репозиториев, можно использовать команду git remote show (рисунок 7).

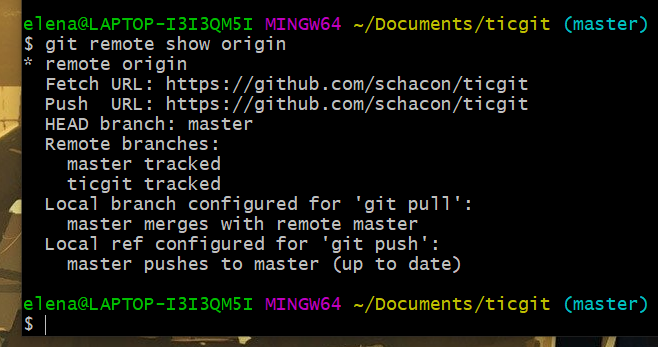


Рисунок 7 - Информация об удаленном репозитории

Для переименования удаленных репозиториев используется команда git remote rename (рисунок 8).

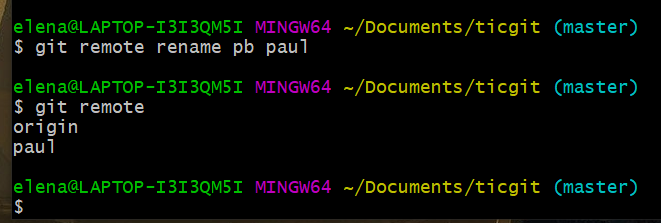


Рисунок 8 - Переименование удаленного репозитория

Для удаления удаленного репозитория нужно выполнить команду git remote remove (рисунок 9).

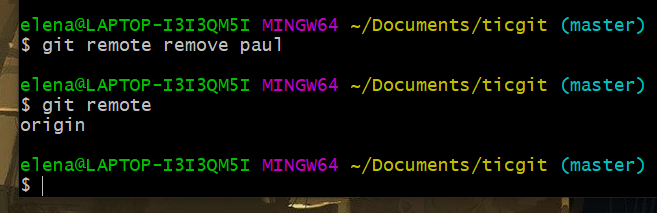


Рисунок 9 - Удаление удаленного репозитория

Просмотреть существующие теги можно с помощью команды git tag (рисунок 10).

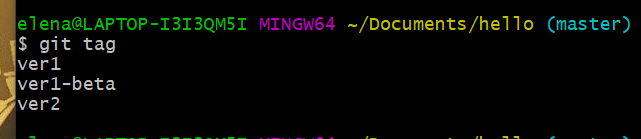


Рисунок 10 - Просмотр тегов

Для создания аннотированной метки нужно выполнить команду, показанную на рисунке 11.

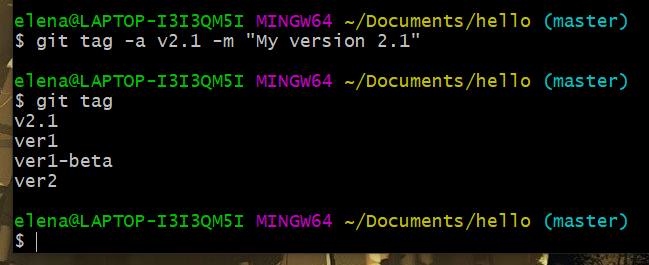


Рисунок 11 - Создание аннотированной метки

Команда git show осуществляет просмотр данных тегов вместе с коммитом (рисунок 12).

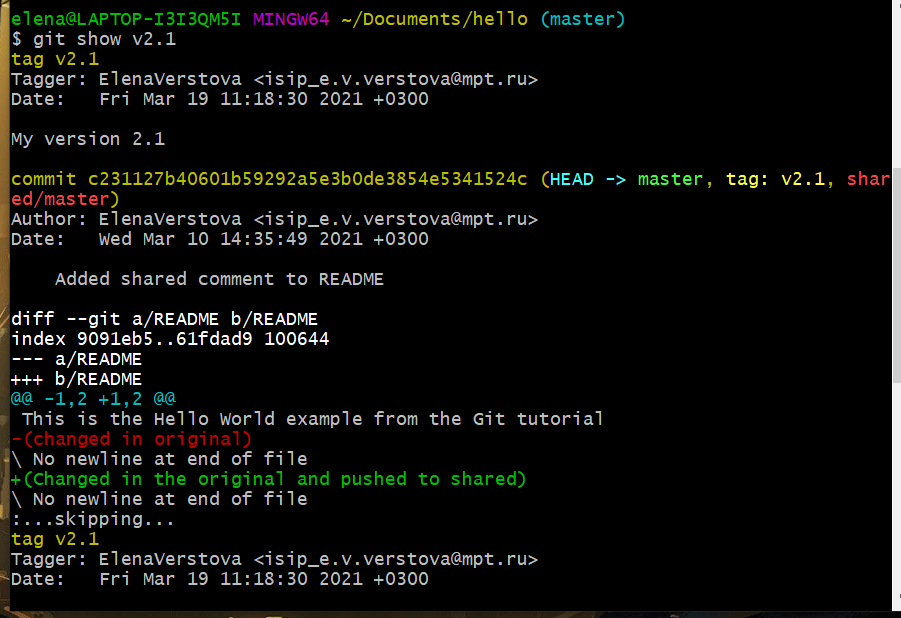


Рисунок 12 - Просмотр данных тега

Для создания легковесной метки не нужно передавать опции -a, -s и -m, надо указать только название (рисунок 13). Просмотр данных такой метки осуществляется также с помощью git show (рисунок 14).

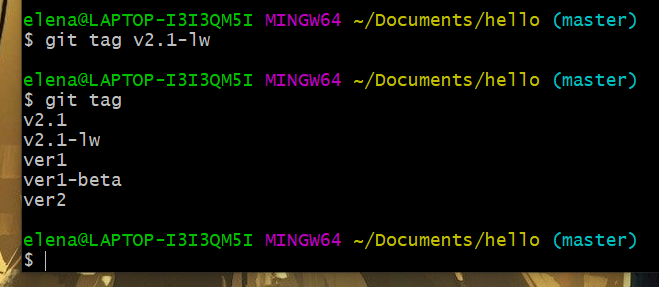


Рисунок 13 - Создание легковесной метки

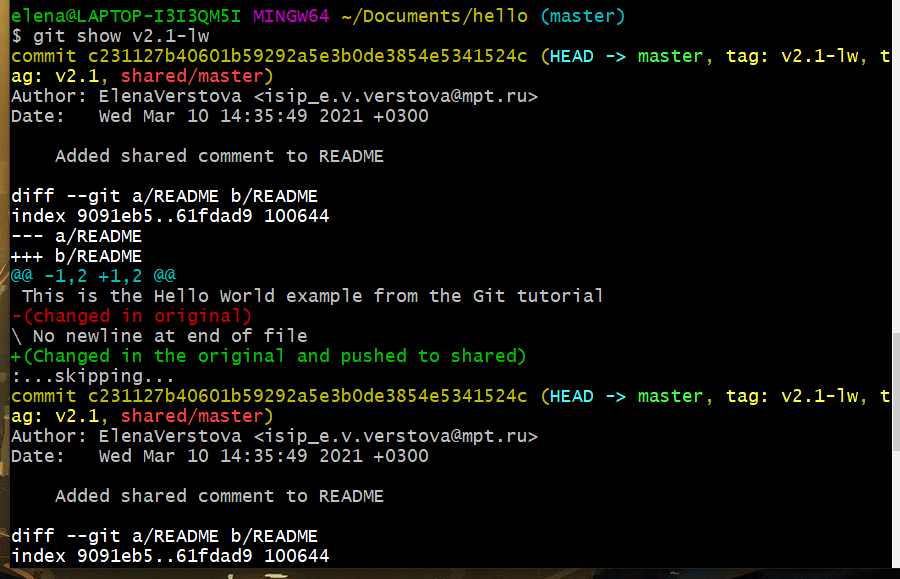


Рисунок 14 - Просмотр данных тега

Для отметки определенного коммита тегом надо указать его хэш (рисунки 15-17).

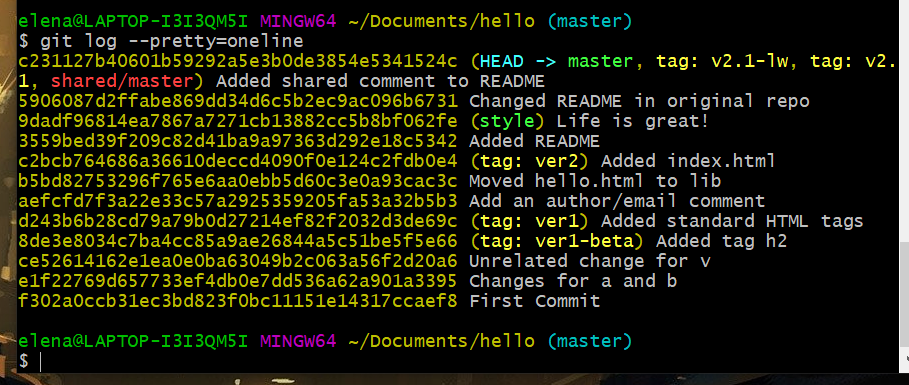


Рисунок 15 - Просмотр истории

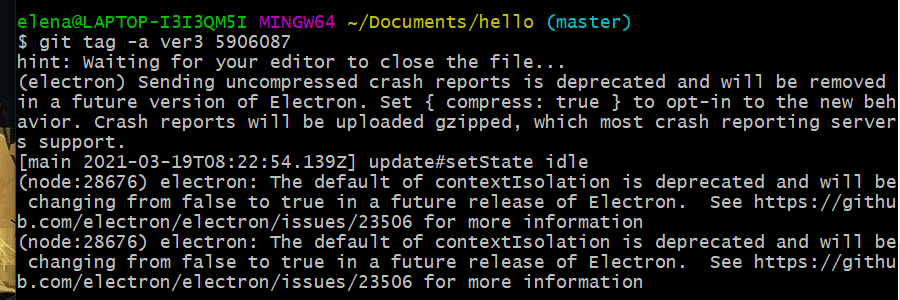


Рисунок 16 - Создание тега определенному коммиту

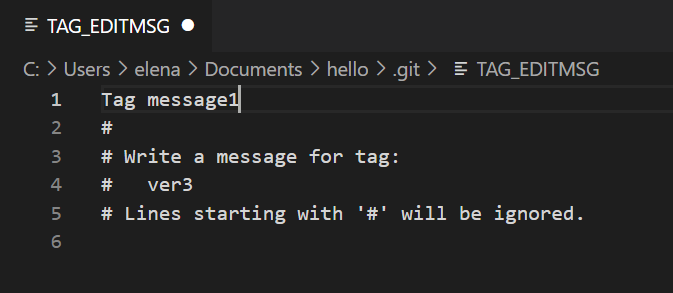


Рисунок 17 - Ввод сообщения в текстовом редакторе

Данные этого тега можно просмотреть аналогичным образом (рисунок 18).

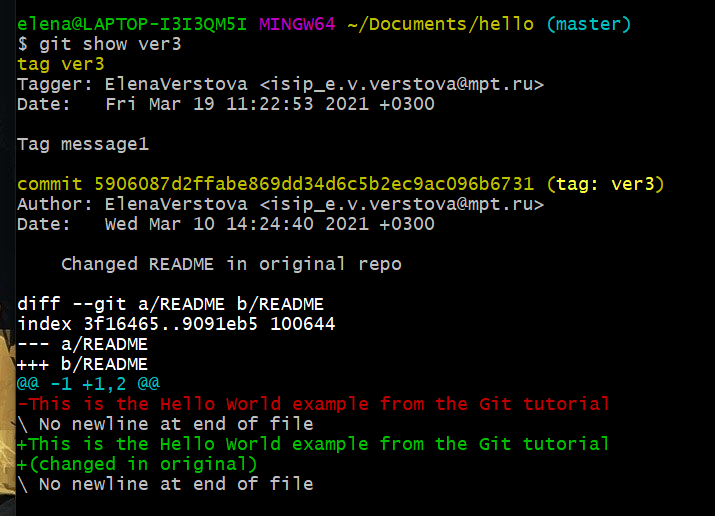


Рисунок 18 - Просмотр данных тега

По умолчанию, команда git push не отправляет теги на удалённые сервера. Нужно выполнить команду git push shared (рисунок 19).

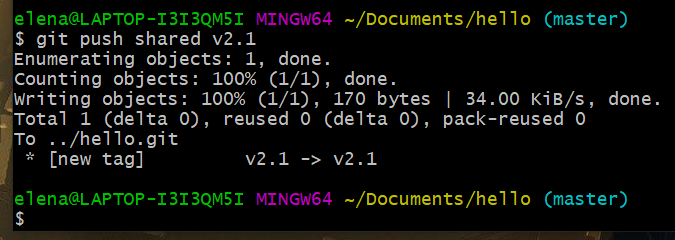


Рисунок 19 - Отправка тега на удаленный сервер

Можно использовать опцию --tags для команды git push. В таком случае все теги отправятся на удалённый сервер (рисунок 20).

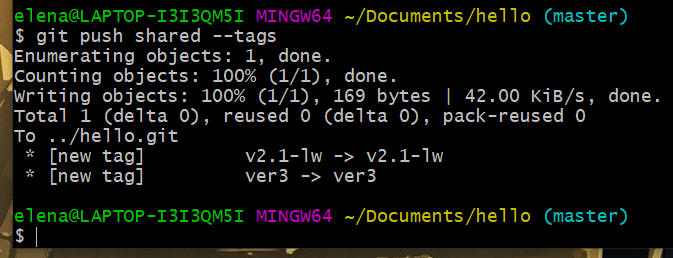


Рисунок 20 - Отправка всех тегов на сервер

Для того, чтобы удалить тег, надо использовать команду git tag с параметром -d (рисунок 21).

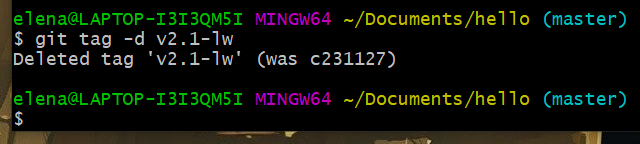


Рисунок 21 - Удаление тега

Для удаления тега с сервера используется команда, показанная на рисунке 22.

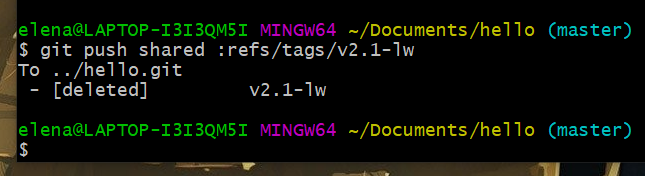


Рисунок 22 - Удаление тегов с сервера

Для того, чтобы получить версии файлов, на которые указывает тег, можно выполнить git checkout для тега. Однако, это переведёт репозиторий в состояние «detached HEAD». Если в состоянии «detached HEAD» внести изменения и сделать коммит, то тег не изменится, при этом новый коммит не будет относиться ни к какой из веток, а доступ к нему можно будет получить только по его хэшу. Поэтому в таком случае следует создать новую ветку (рисунки 23-24).

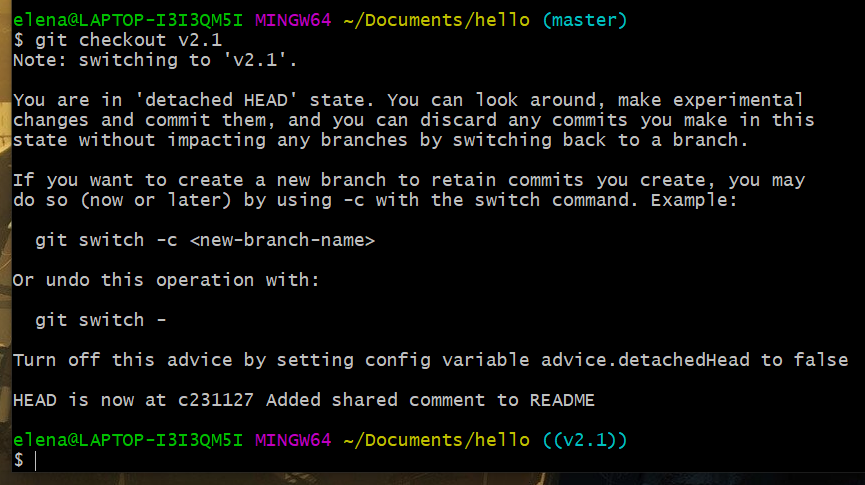


Рисунок 23 - Переключение на метку

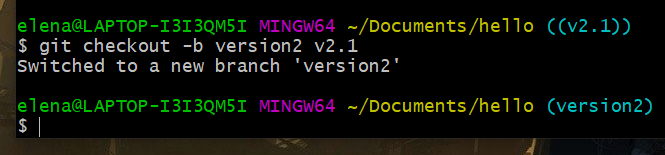


Рисунок 24 - Создание новой ветки

Можно создать псевдонимы (алиасы) для команд. Создание алиасов и примеры их использования показаны на рисунках 25-30.

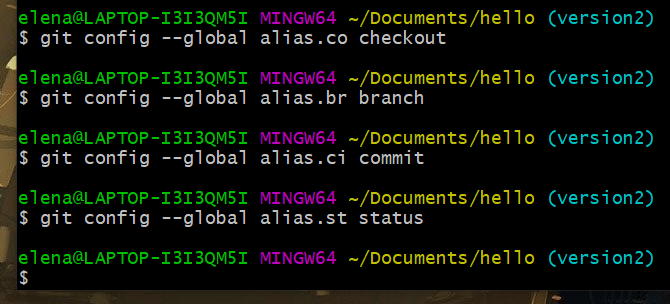


Рисунок 25 - Задание алиасов

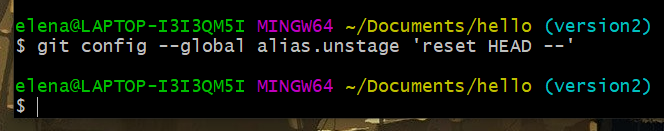


Рисунок 26 - Создание псевдонима исключения файла из индекса

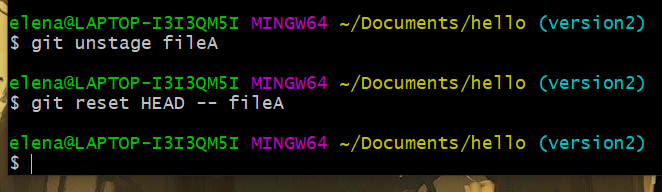


Рисунок 27 - Использование созданного псевдонима

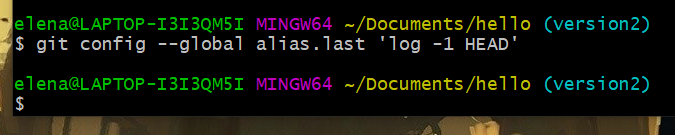


Рисунок 28 - Создание алиаса для просмотра последнего коммита

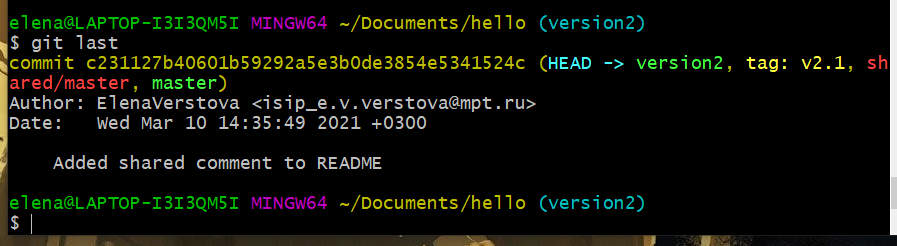


Рисунок 29 - Результат работы созданного алиаса

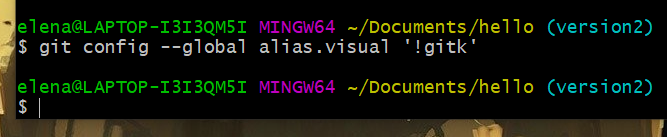


Рисунок 30 - Создание псевдонима внешней команды

