**ОТЧЕТ**

**о выполнении лабораторной работы №4**

**Дисциплина «МДК 03.01 Моделирование и анализ программного обеспечения»**

**Тема: «Создание и изучение возможностей репозитория проекта»**

**Специальность: 09.02.07 Информационные системы и программирование**

**Квалификация: Специалист по информационным системам**

Выполнил: Стадников М.А.

Студент группы: ИС50-2-22

Проверил:

Преподаватель: Кретова Е.М.

Дата: «\_\_\_» \_\_\_\_\_\_\_\_\_ 2024 г.

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Теоретические сведения**

Команда **git config** позволяет настроить инсталляцию Git (или отдельный репозиторий) из командной строки. С помощью этой команды можно установить любые настройки: от информации о пользователе до его предпочтений и характеристик репозитория.

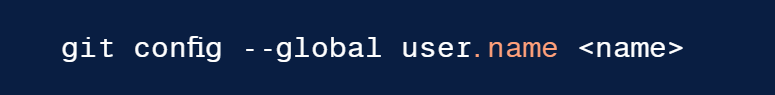
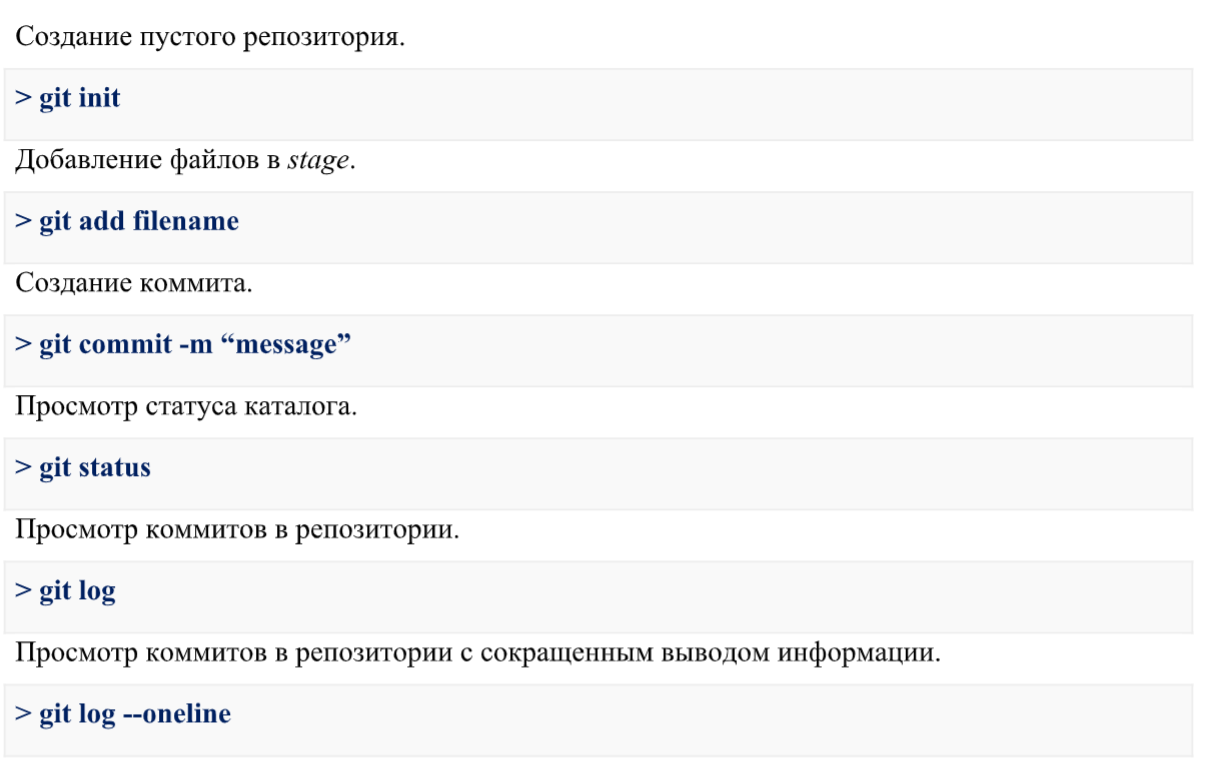


Рисунок 1 – Эта команда задает имя автора, которое будет использоваться для всех коммитов, выполненных текущим пользователем

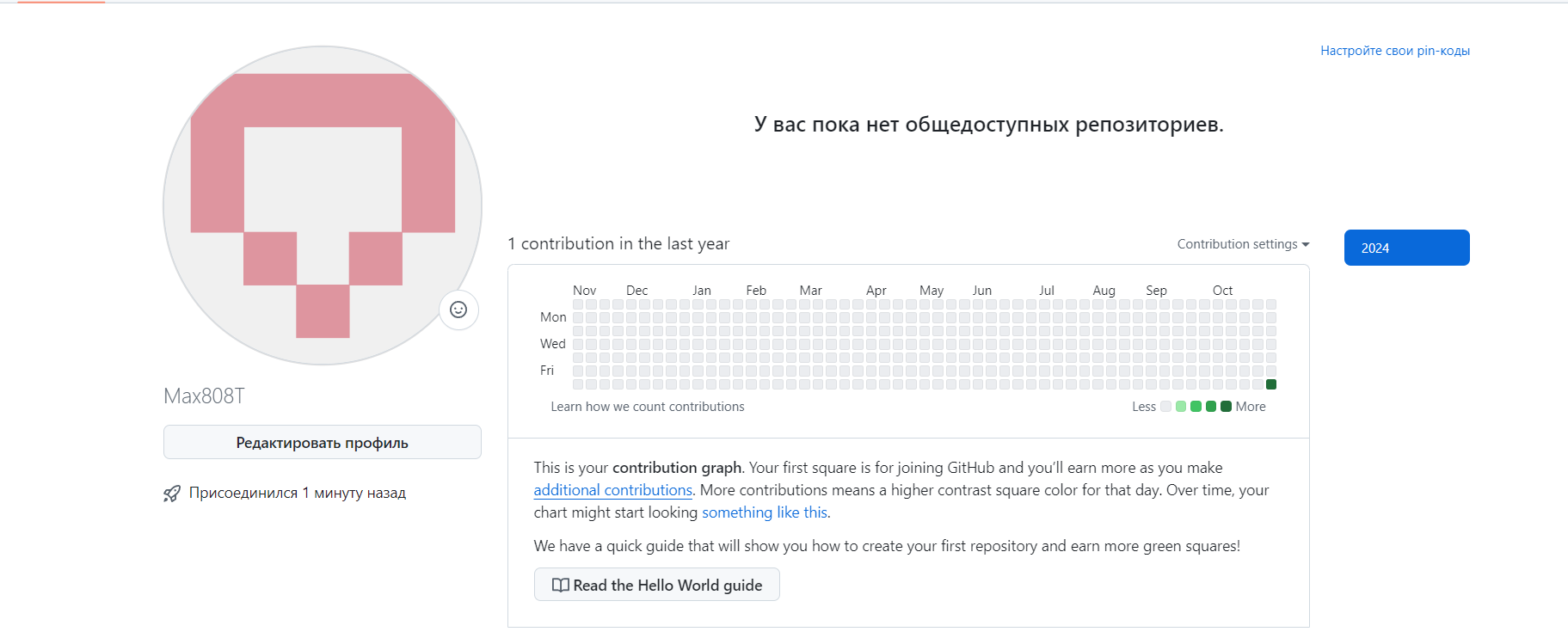


Рисунок 2 – Эта команда задает адрес электронной почты автора, который будет использоваться для всех коммитов, выполненных текущим пользователем



**Порядок выполнения работы**

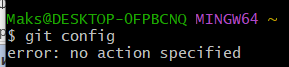
1. Необходимо зарегистрироваться на GitHub <https://github.com/>



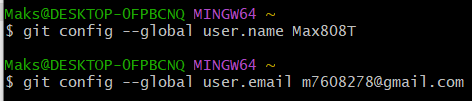
1. Запускаем Git Bash



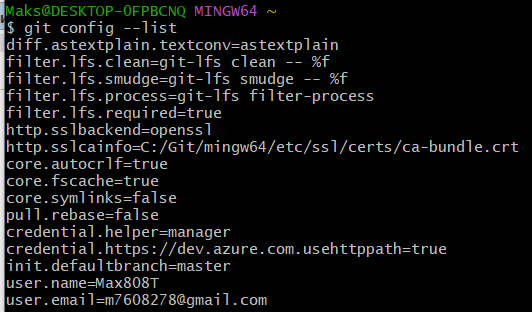
1. Пишем команду **git config**, которая позволяет посмотреть доступные команды



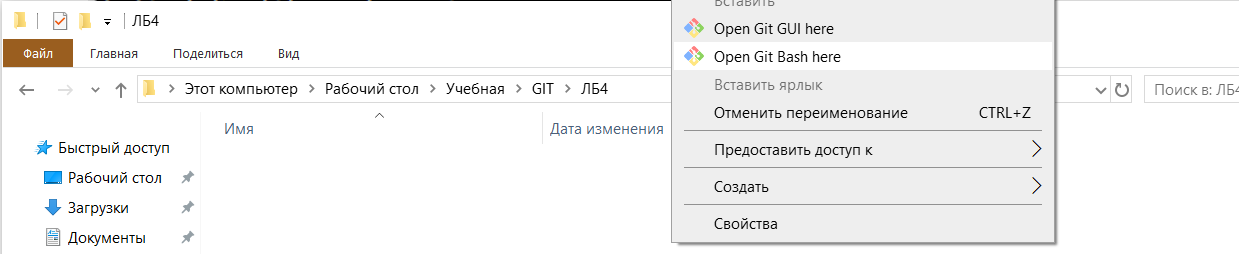
1. Добавляем в Git контактные данные (это необходимо для идентификации изменений. Эти данные должны совпадать с аккаунтом, зарегистрированным на GitHub – **ваша почта и ваш ник**

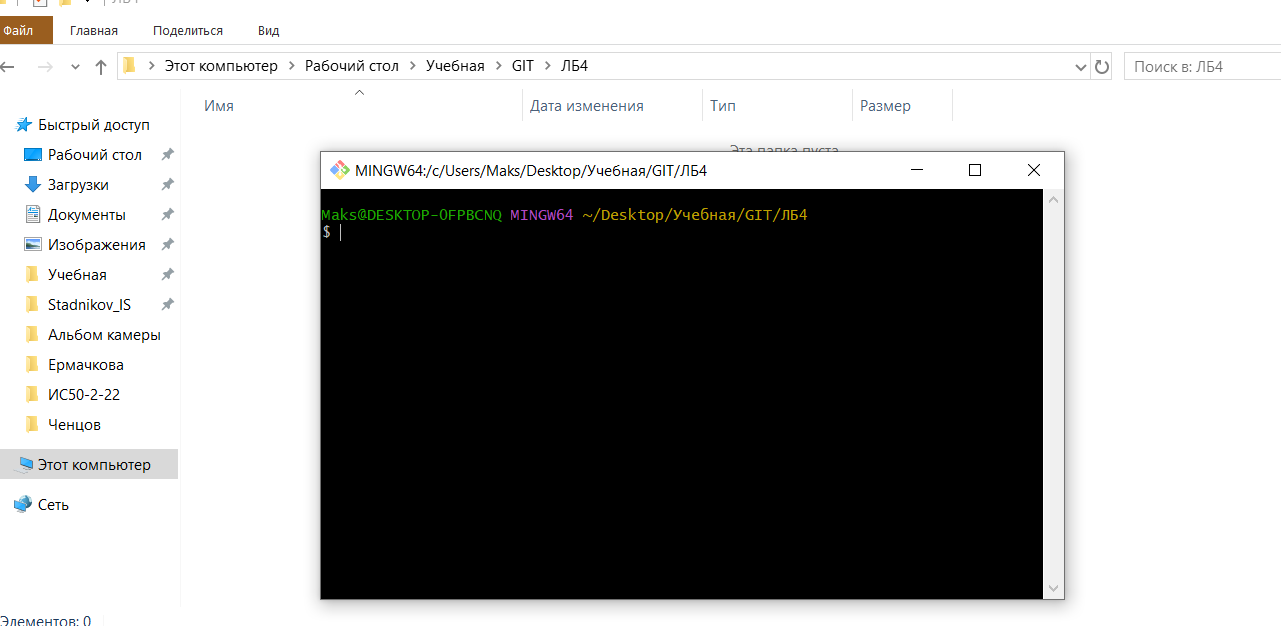


1. Получаем все заданные параметры при помощи команды **git config –list**

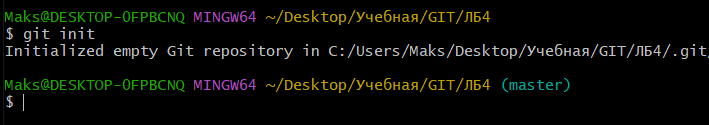


1. Для того чтобы создать репозиторий, для начала, создайте папку, в которой он будет располагаться. Далее переходим в папку, правой кнопкой мыши вызываем контекстное меню и выбираем GitBash Here

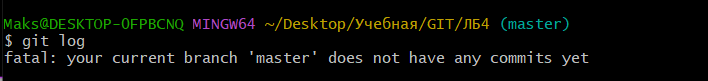




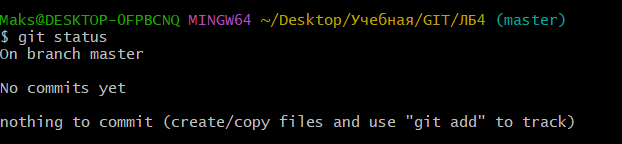
1. Инициализируем новый репозиторий при помощи команды **git init**



1. Если мы посмотрим на список коммитов, которые были отправлены в репозиторий, то увидим, что он пустой – это правильно, т.к. мы пока только создали репозиторий и ничего ещё туда не отправляли при помощи команды **git log**



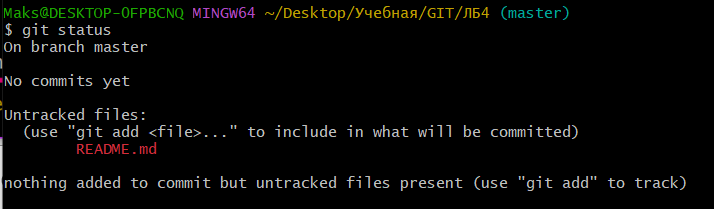
1. Просматриваем текущий статус репозитория при помощи команды **git status**

****

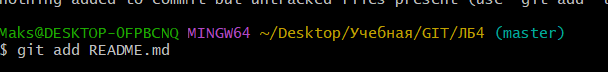
1. Создадим в нашем каталоге пустой файл при помощи команды **touch** README.md (название файла)



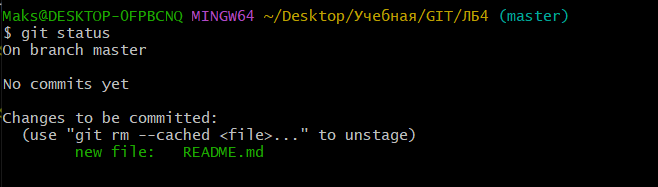
1. Теперь, если мы выполним команду git status, то увидим, что в нашем каталоге появился один неотслеживаемый файл: README.md



1. Добавим, созданный файл в stage. Stage (или cache) – это хранилище для файлов с изменениями, информация о которых попадет в единый коммит. Для добавления файла README.md в stage необходимо воспользоваться командой **git add** README.md

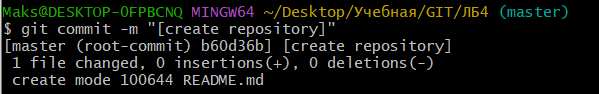


1. Выполним **git status** для того, чтобы посмотреть на то, что сейчас происходит в нашем каталоге

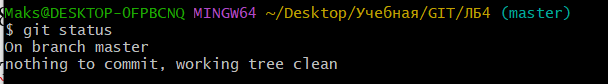


1. Как видно, в stage был добавлен один файл с именем README.md и теперь представленный набор изменений готов к отправке в репозиторий – т.е. к коммиту

(**git commit -m “[create repository]”**)



1. Проверим статус каталога. Как видно с момента последнего коммита никаких изменений в рабочем каталоге не производилось (**git status**)



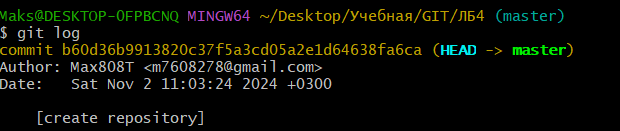
1. Теперь взглянем на список коммитов.

Из приведенной информации видно, что был отправлен один коммит, который имеет

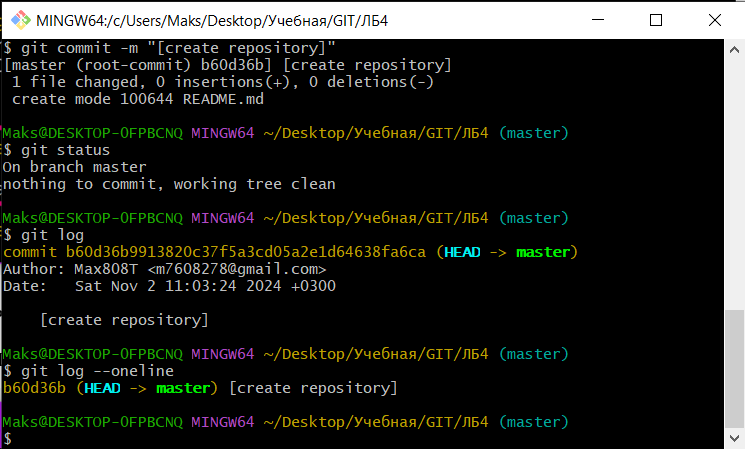
ID: 11051cdedb629fc38b367894bb919628269298d1.

Автор данного коммита rompersStomper

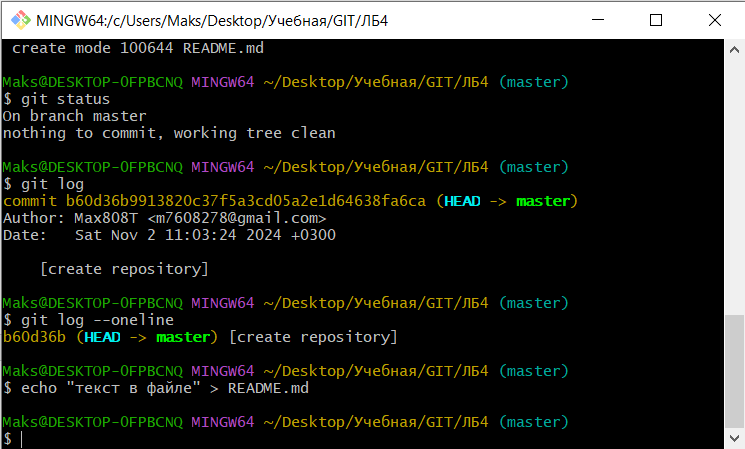
Коммит был создан Wed Oct 25 01:06:04 2023 +0300, с сообщением: [create repository].



1. Это довольно подробная информация, когда коммитов станет много, такой формат вывода будет не очень удобным, сокращенный вариант выглядит так (**git log --oneline**)



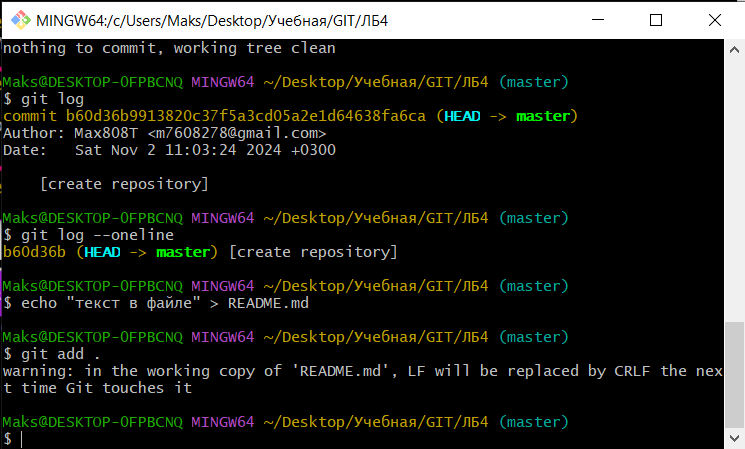
1. Внесем еще несколько изменений в наш репозитории. Добавим текст в файл README.md. (**echo** “текст в файле» > README.md)

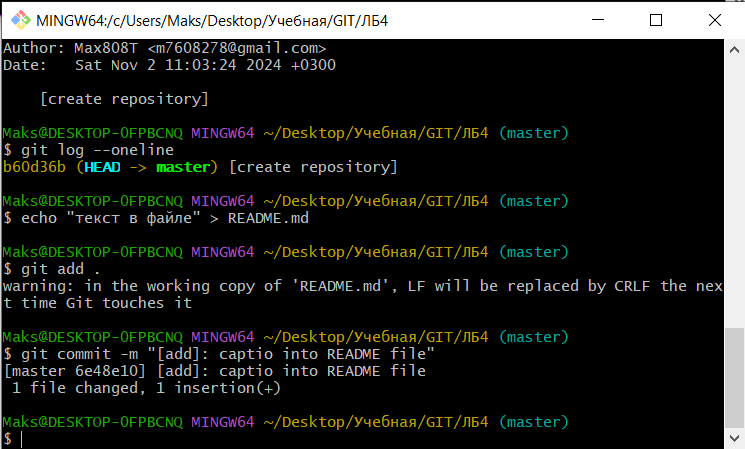


1. Зафиксируем эти изменения в репозитории

git add .

git commit -m “[add]: caption into README file”





1. Изменения в файле

