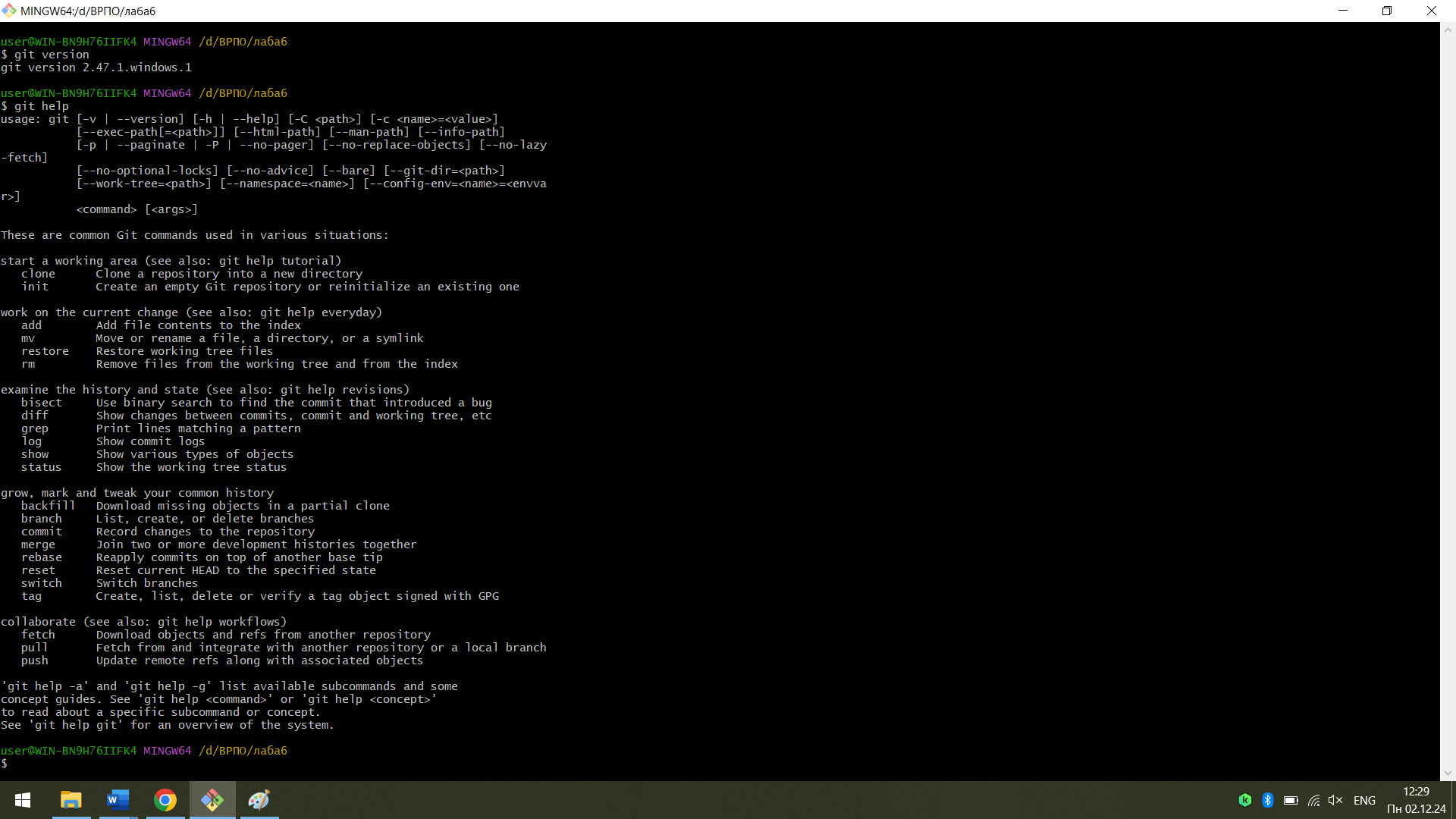
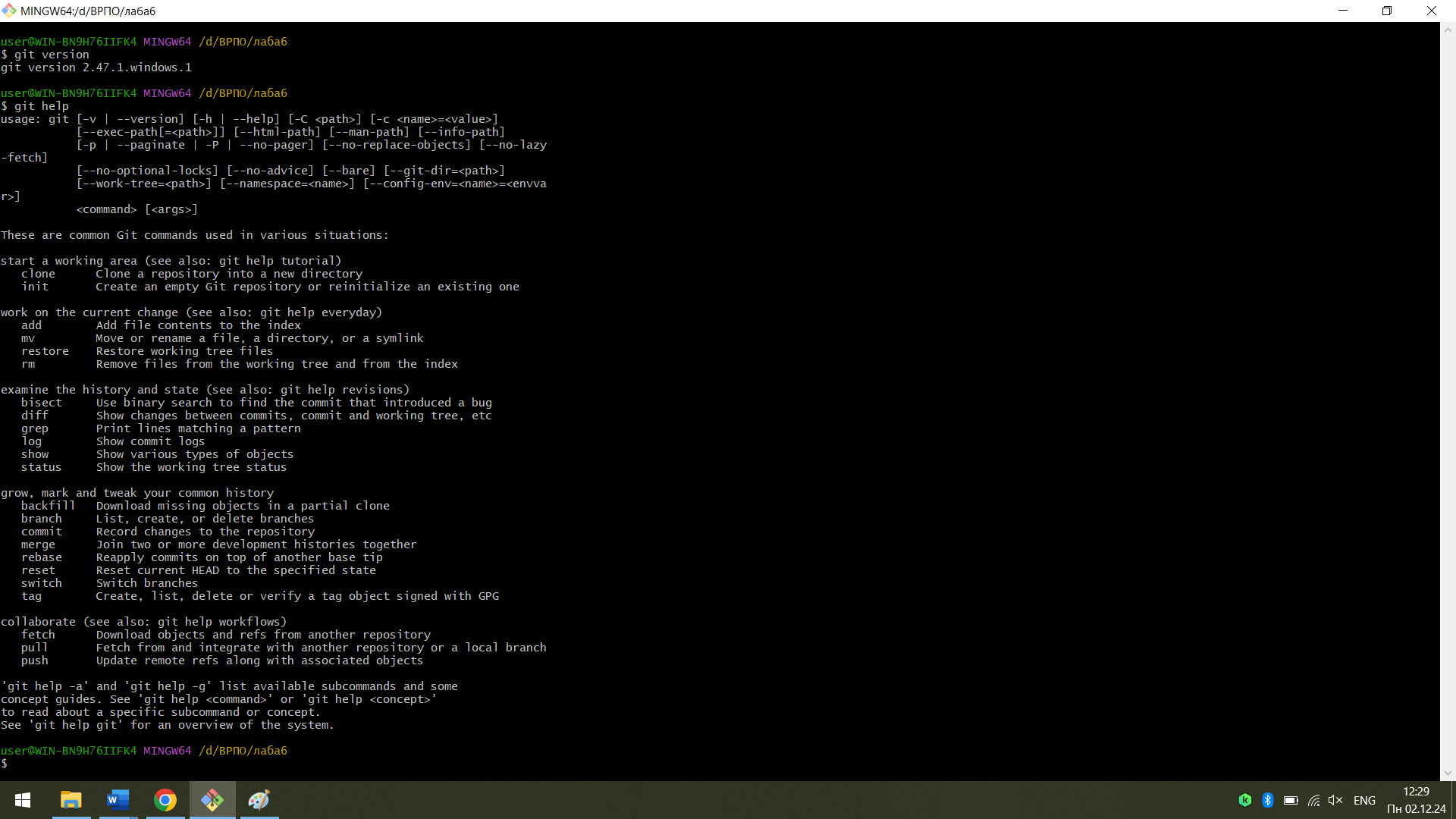
**Лабораторная работа №6**

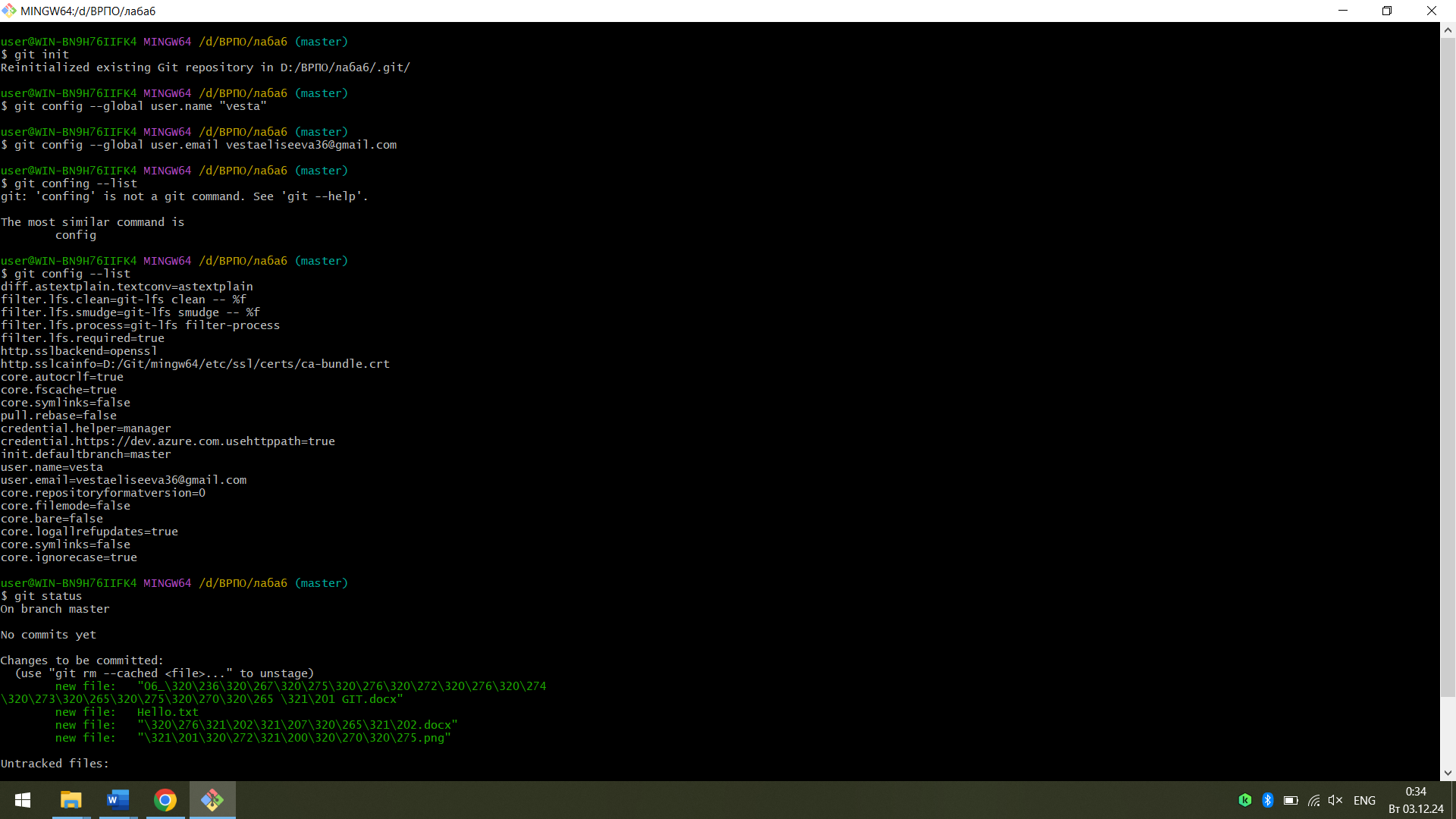
Проверка установленной версии



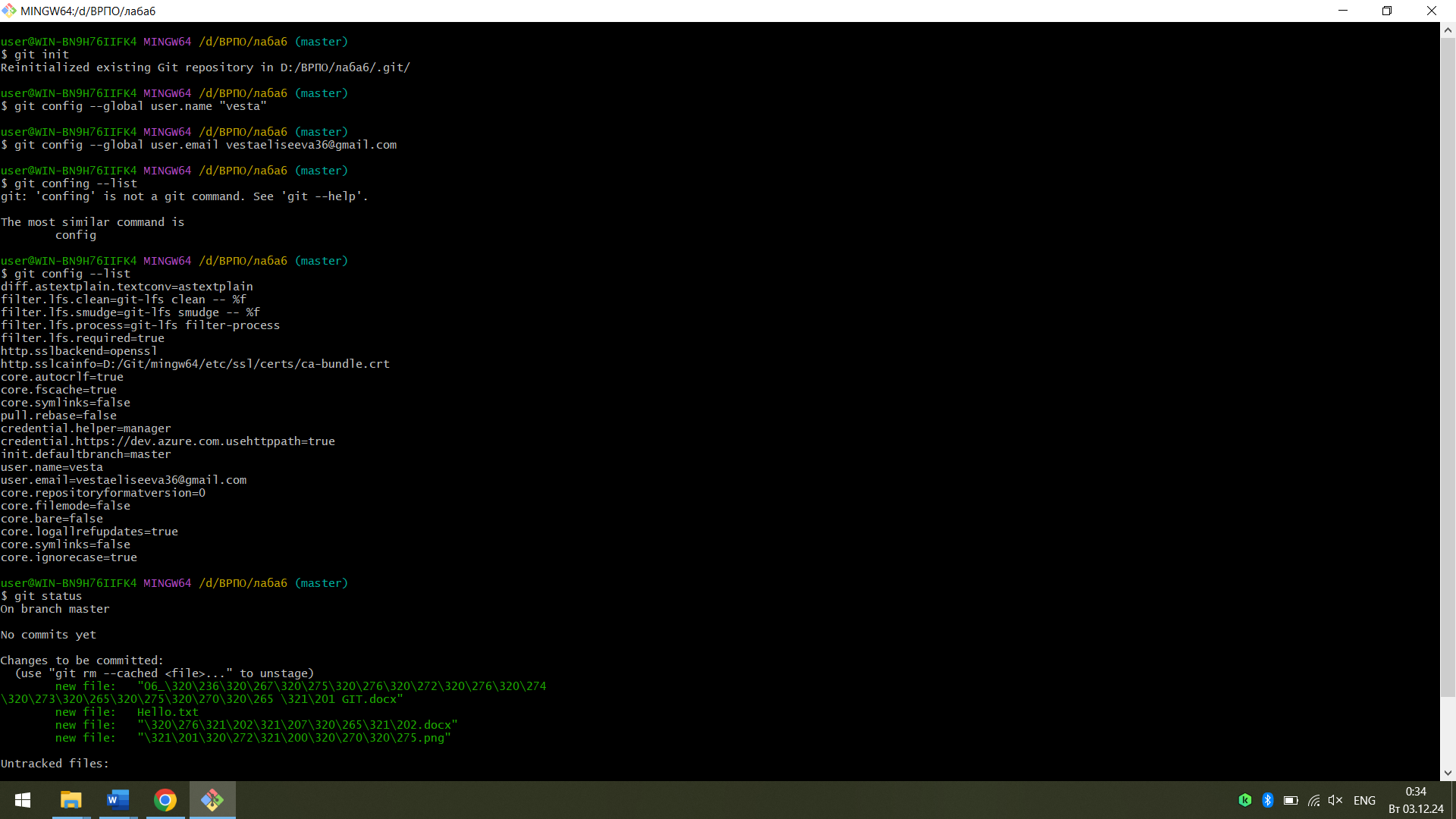
Для получения справки по Git используем команду **git help**



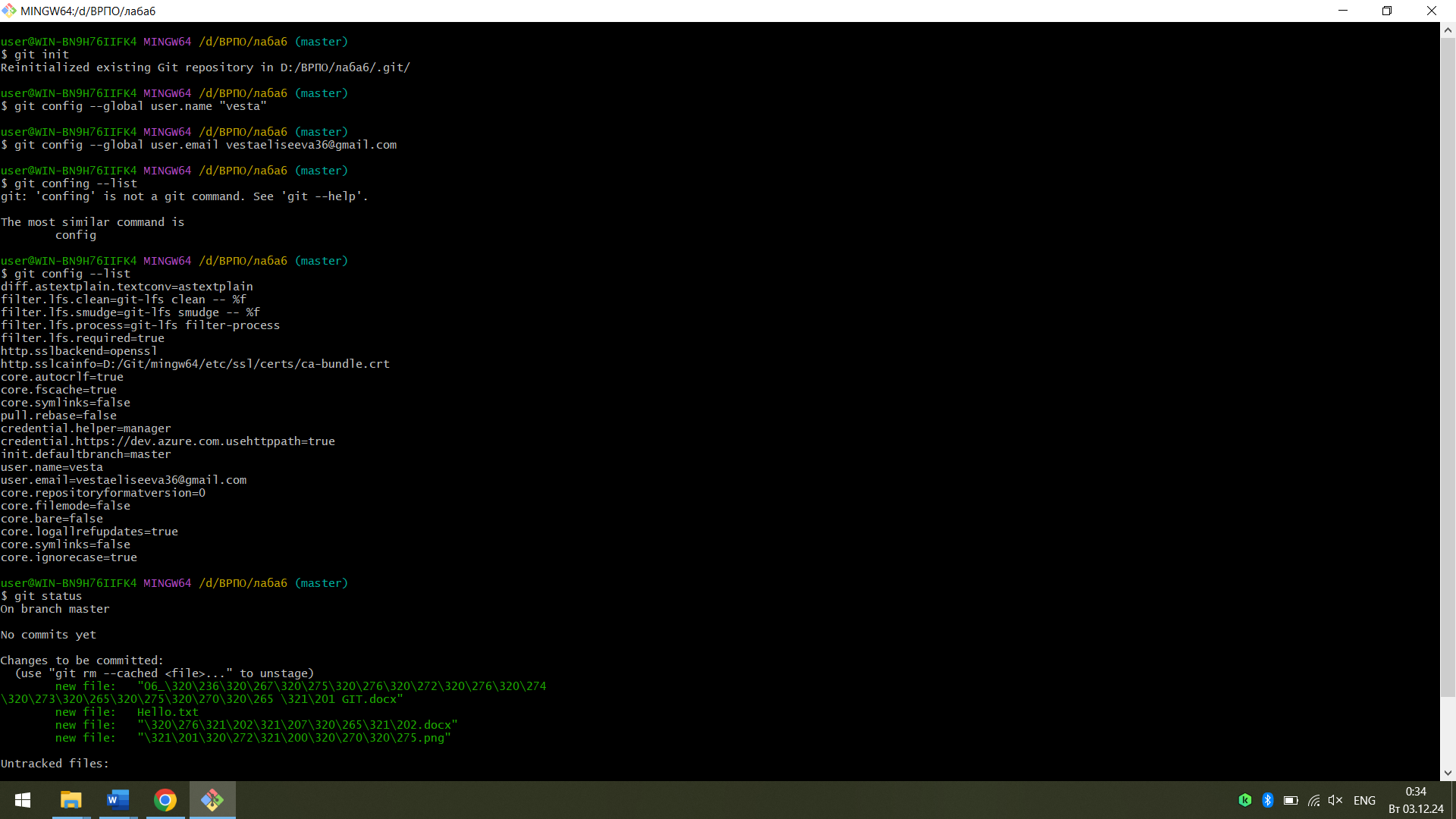
Создание локального репозитория с помощью команды **git init**



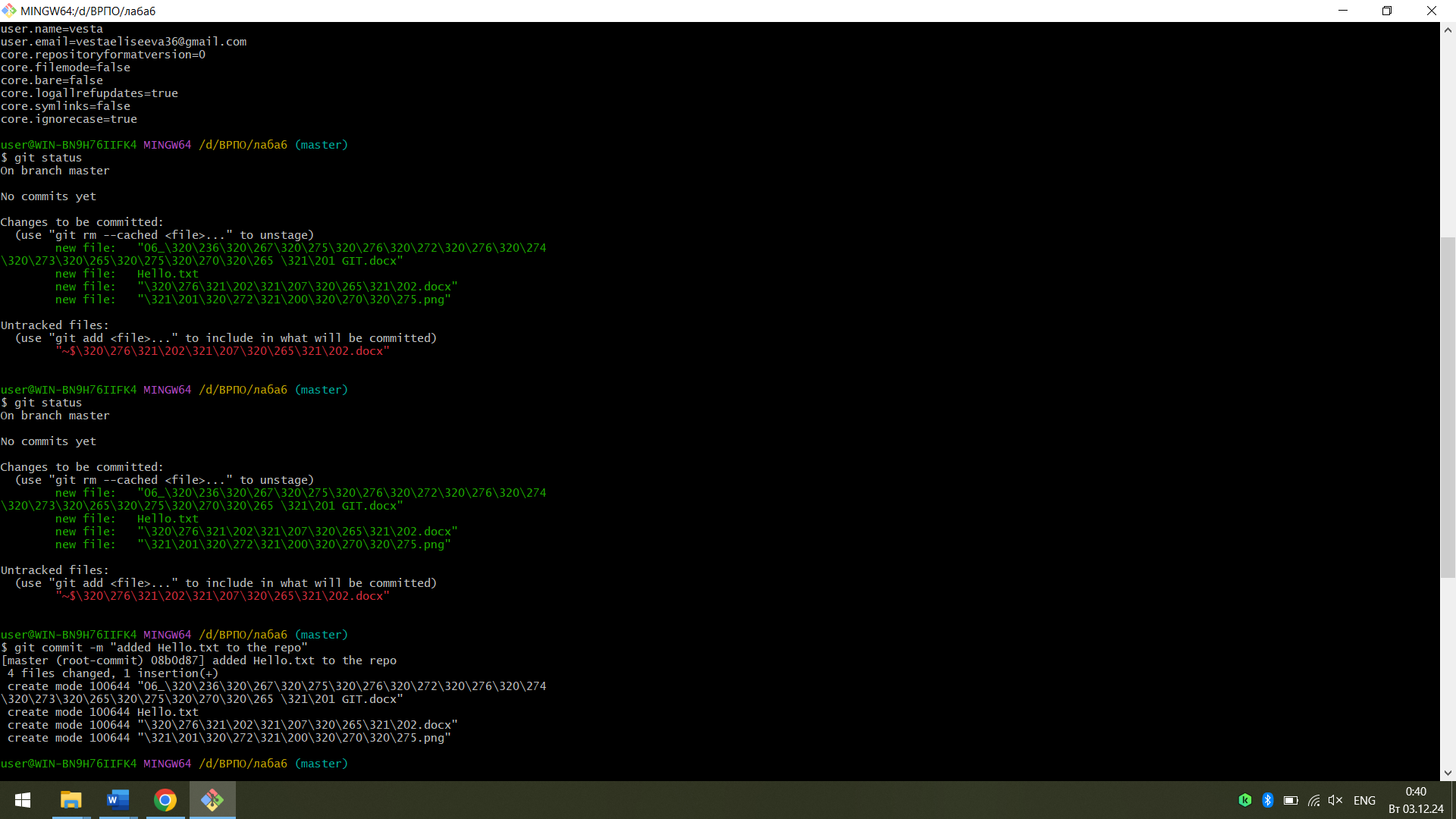
Настройка конфигурационного файла:



Проверка настройки конфигурационного файла командой **git config –list**

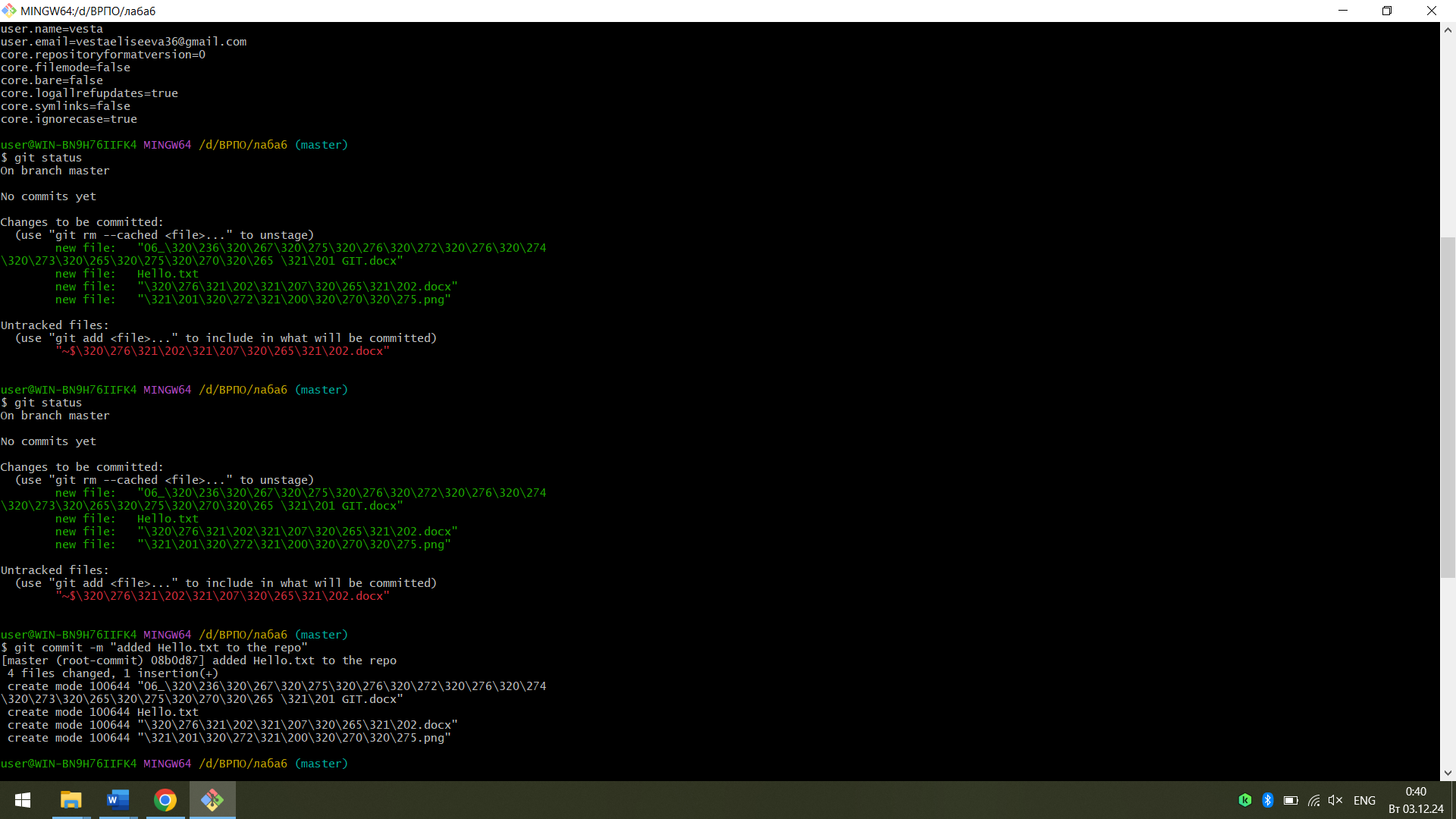


Получение информации о статусе репозитория с помощью **git status**



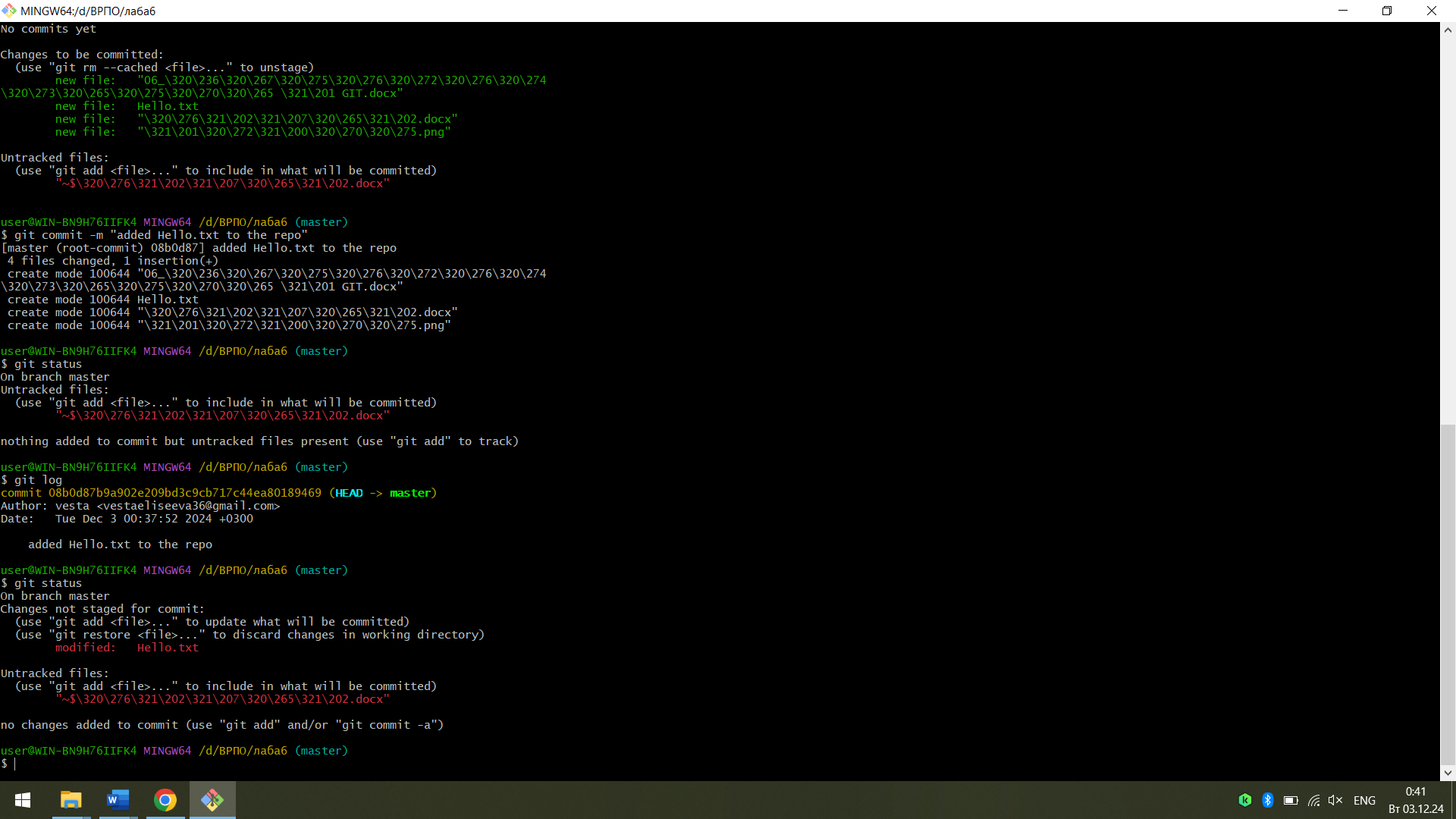
Наш файл Hello.txt подсвечивается зелёным цветом, это значит, что он проиндексирован

Зафиксируем изменения индексированных файлов в репозитории с помощью команды **git commit –m “added Hello.txt to the repo”**:



Ключ –m позволяет добавить комментарий, описывающий, что именно было изменено в коммите ("added Hello.txt to the repo")

Проверим изменения в репозитории:

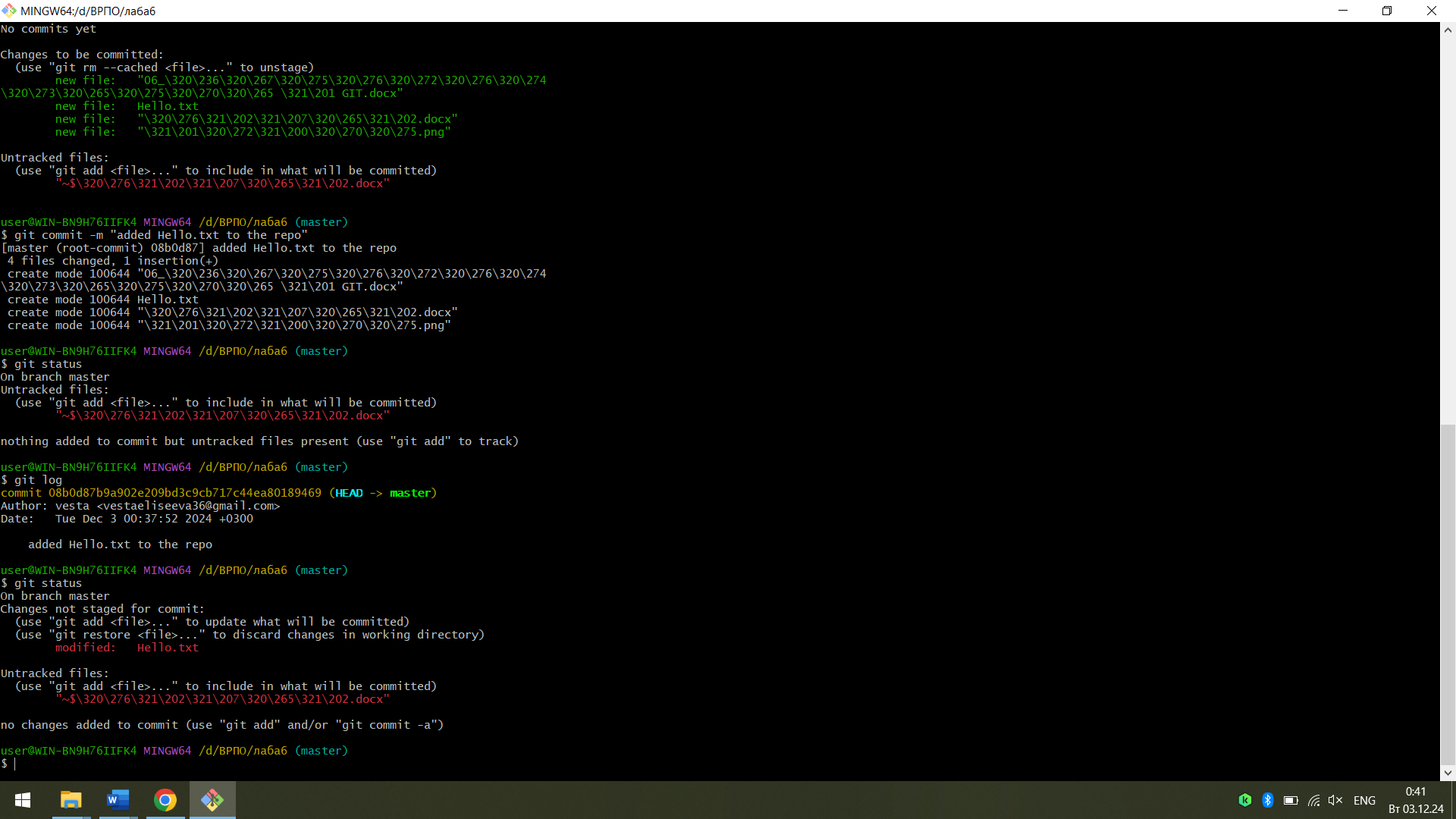


Git информирует об успешном создании нового коммита, в ветку **master** добавлен 1 файл:

[master (root-commit) 4993c08] added Hello.txt to the repo

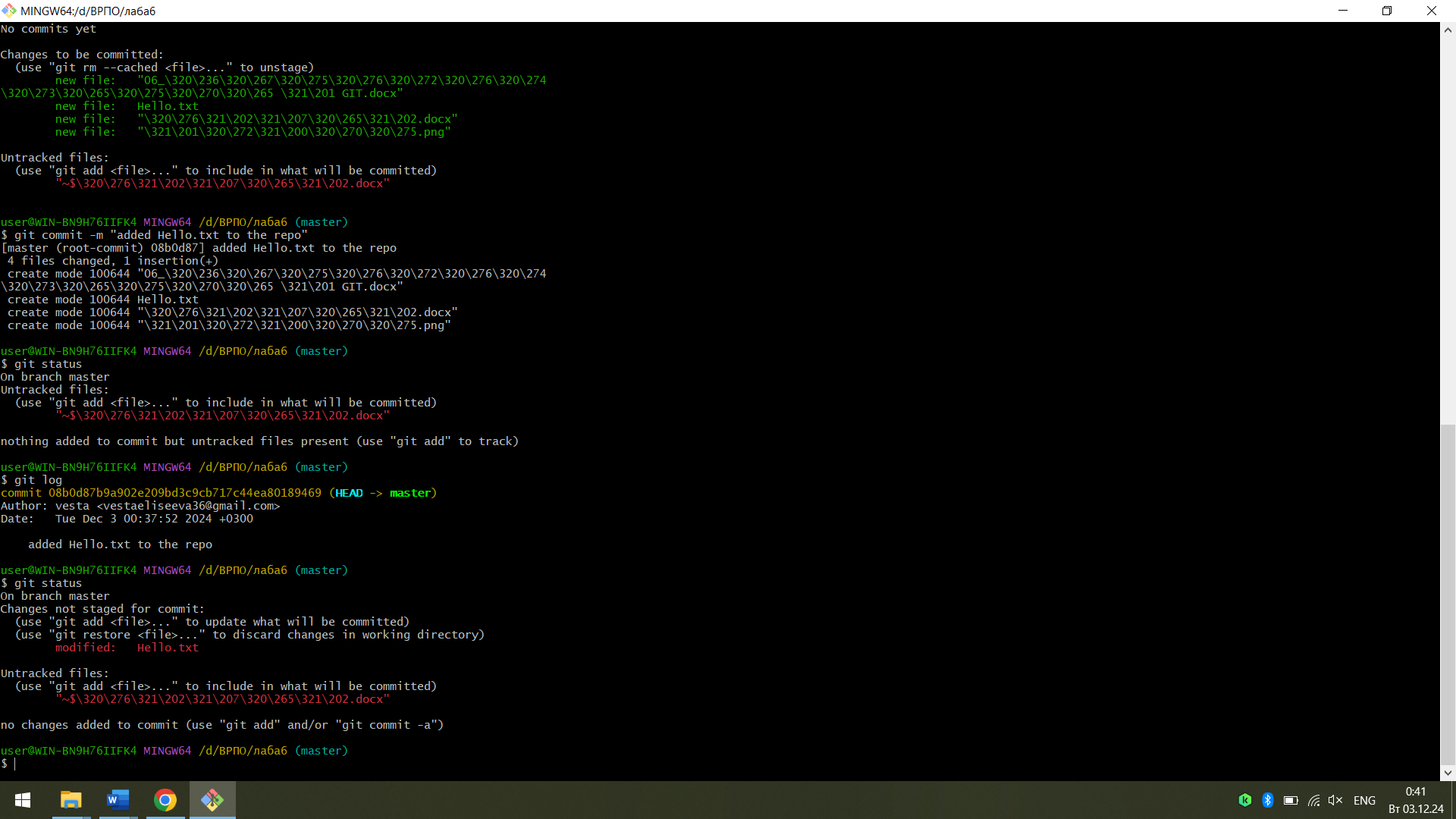
1 file changed, 4 insertions(+)

Просмотр журнала с помощью команды **git log**:



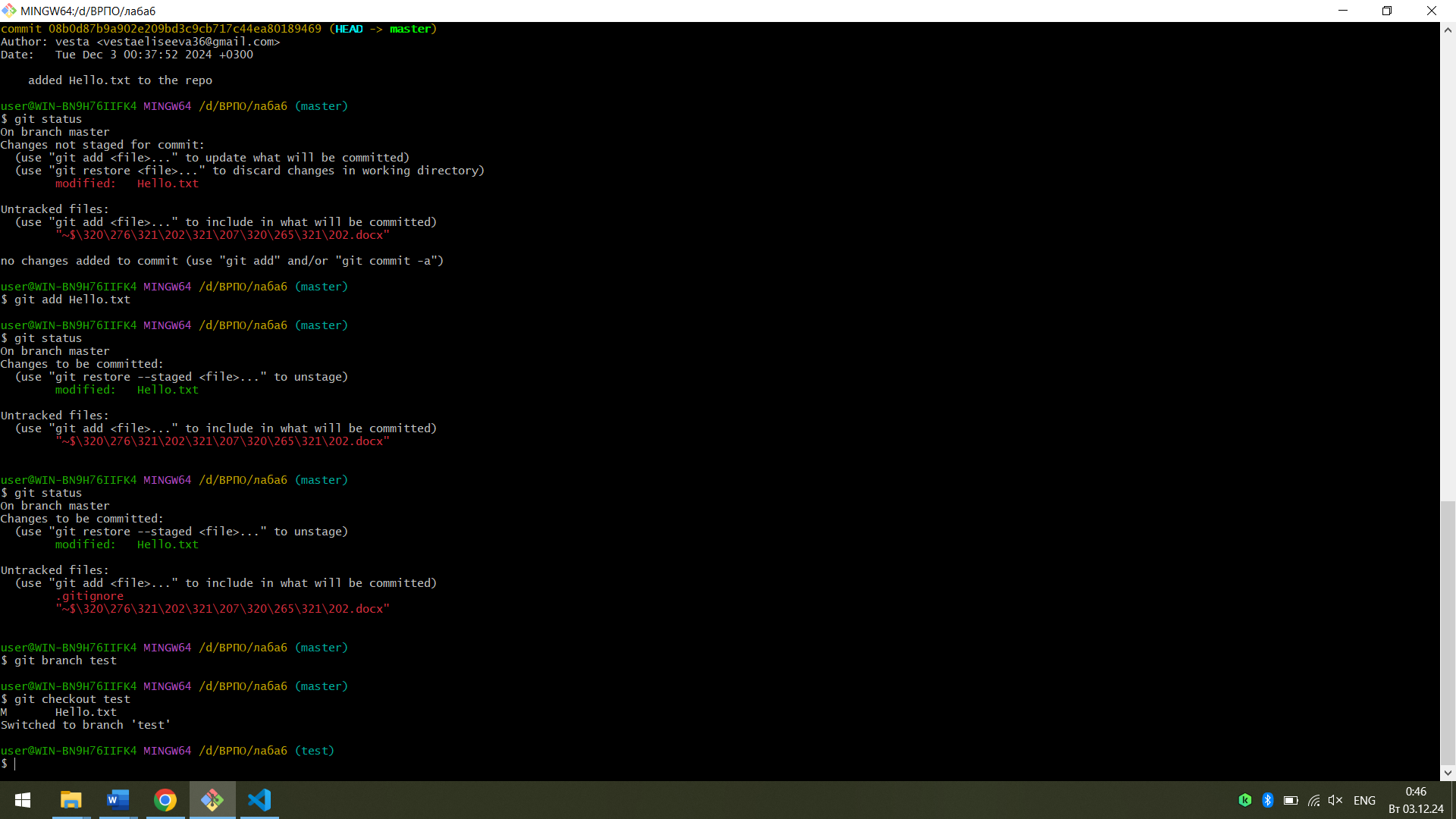
Получение информации о проиндексированных и неиндексированных файлах.

1. Изменим уже существующий файл Hello.txt и сохраним изменения
2. Просмотрим статус нашего репозитория



Можем заметить, что наш файл больше не подсвечивается зелёным цветом, и modified: Hello.txt, что указывает на изменения файла

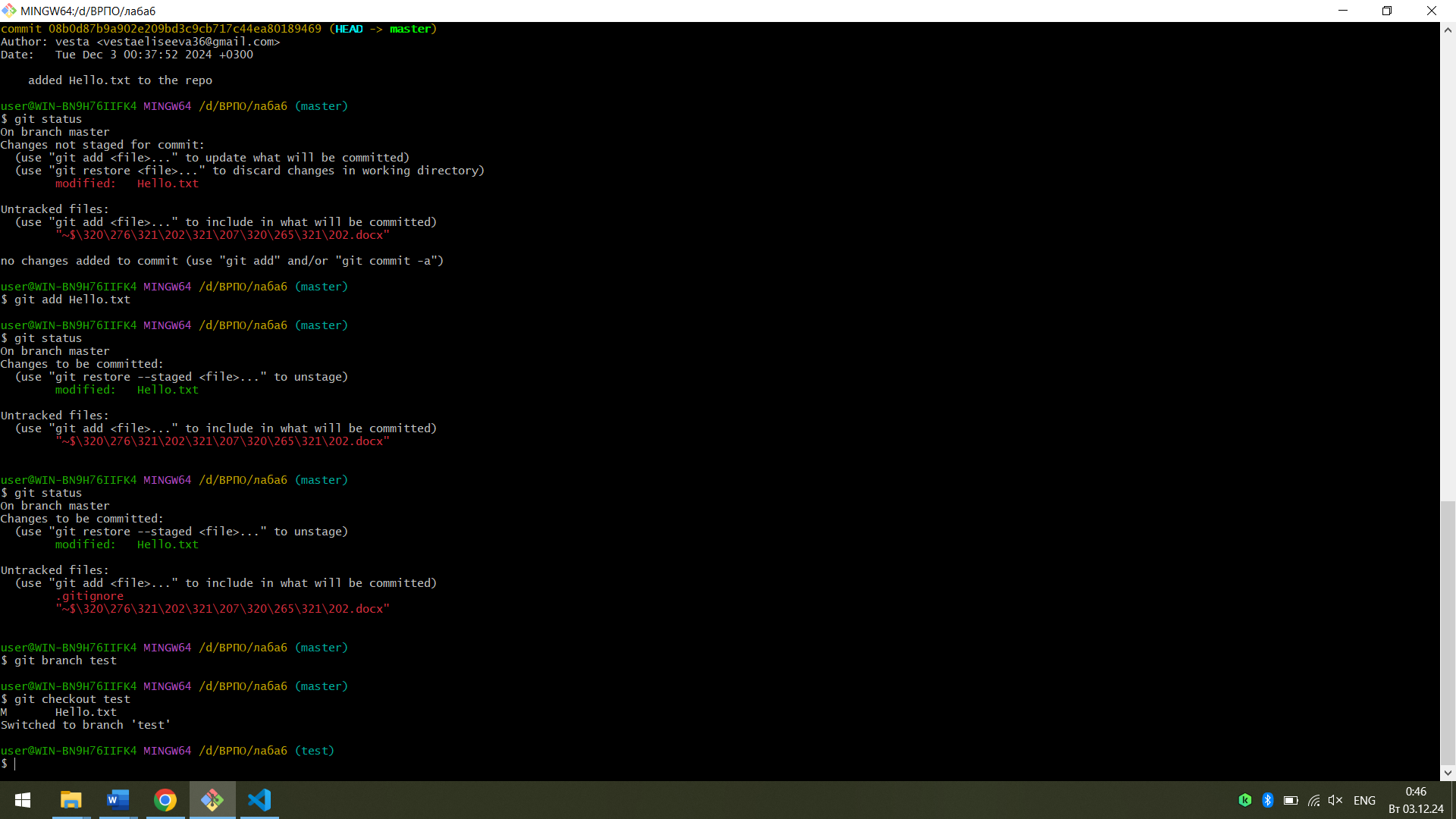
3) При добавлении этого изменённого файла получим



Теперь оно подсвечивается зелёным цветом

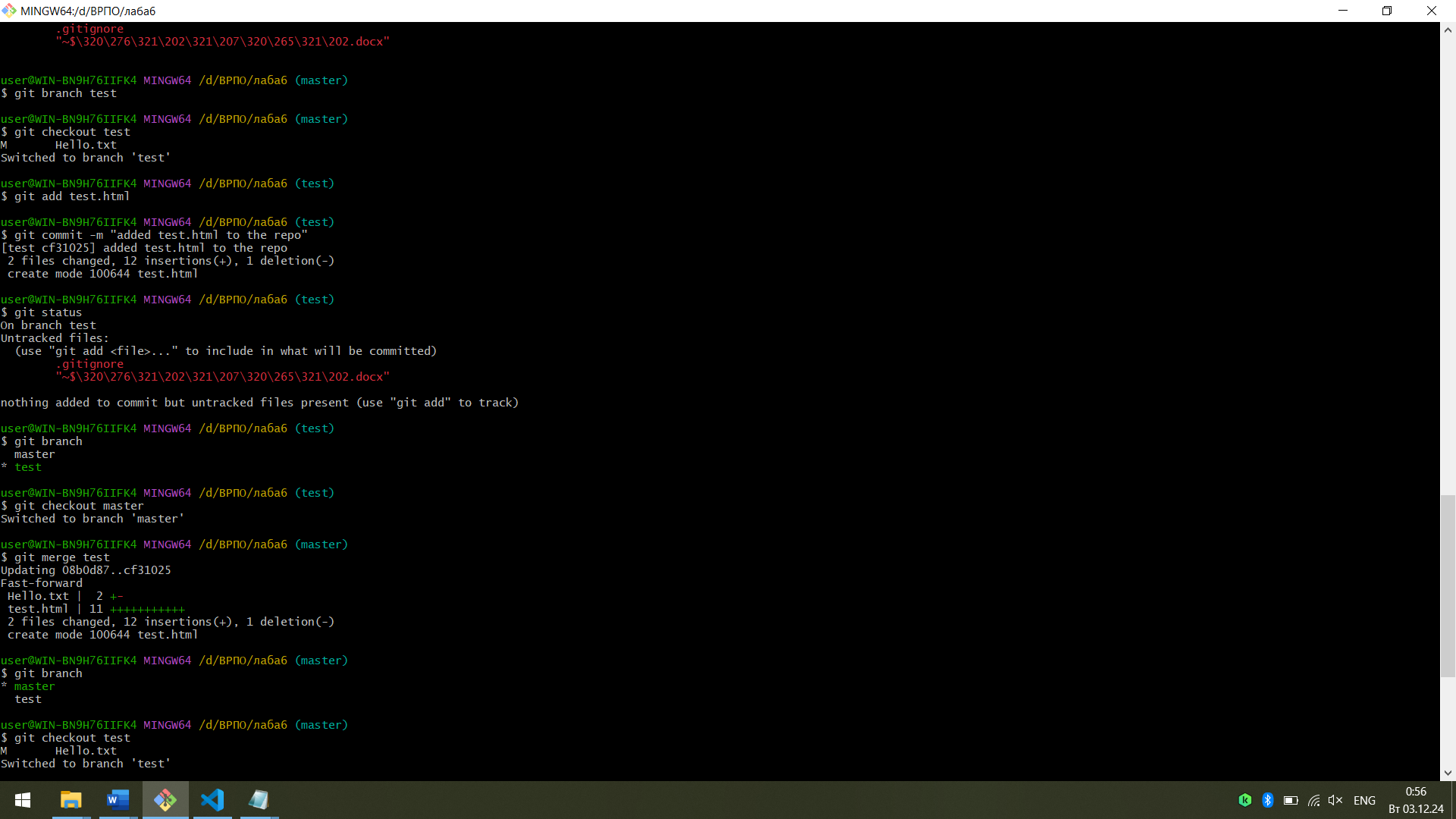
Настройка файла .gitignore .Создаём текстовый файл и переименовываем наш файл в файл с названием **.gitignore** убирая расширение **.txt** и сохраняем

Далее просматриваем статус нашего репозитория и видим, что файл с расширением **.docx** больше не отображаются

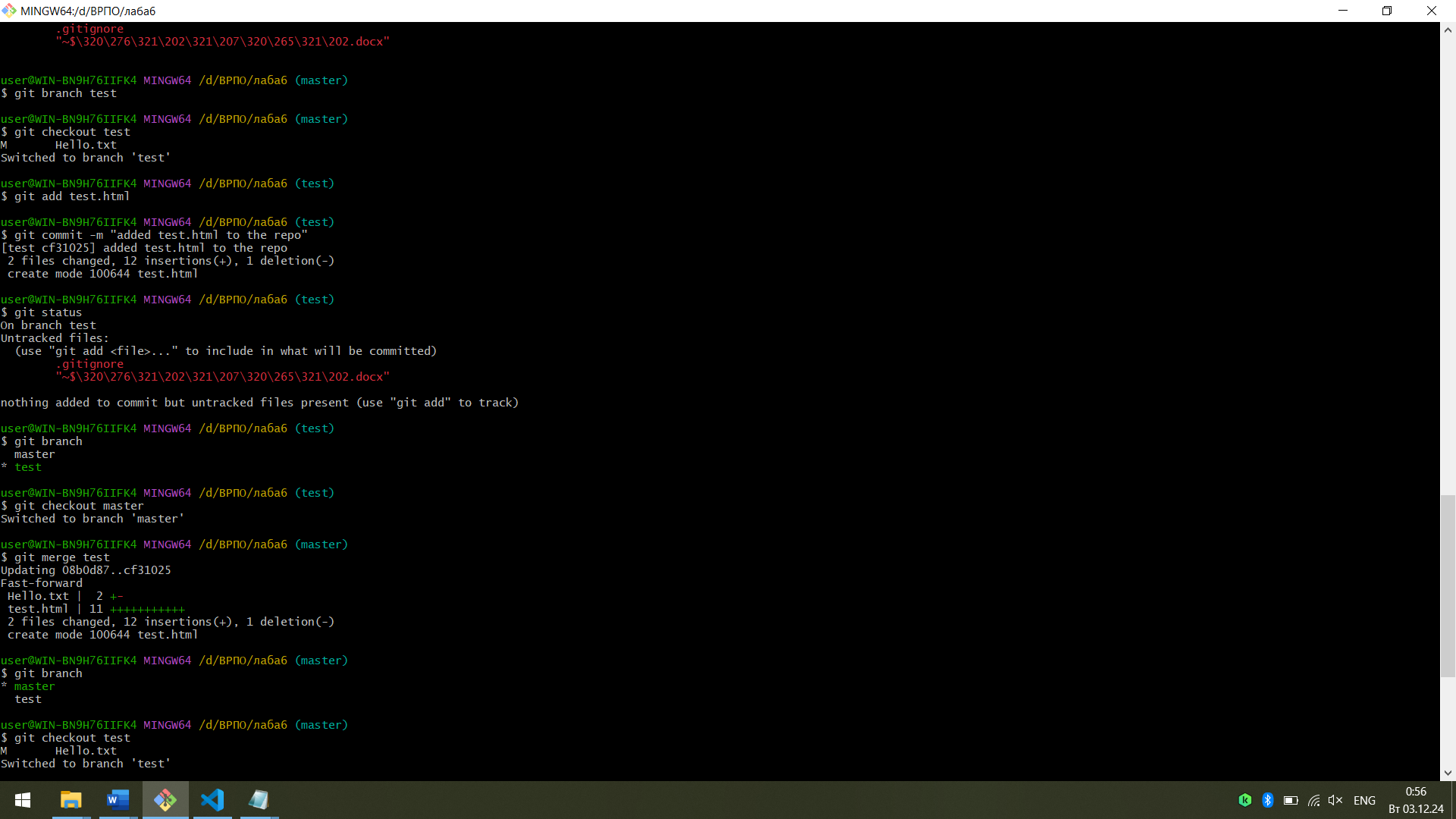


Создание новой ветки

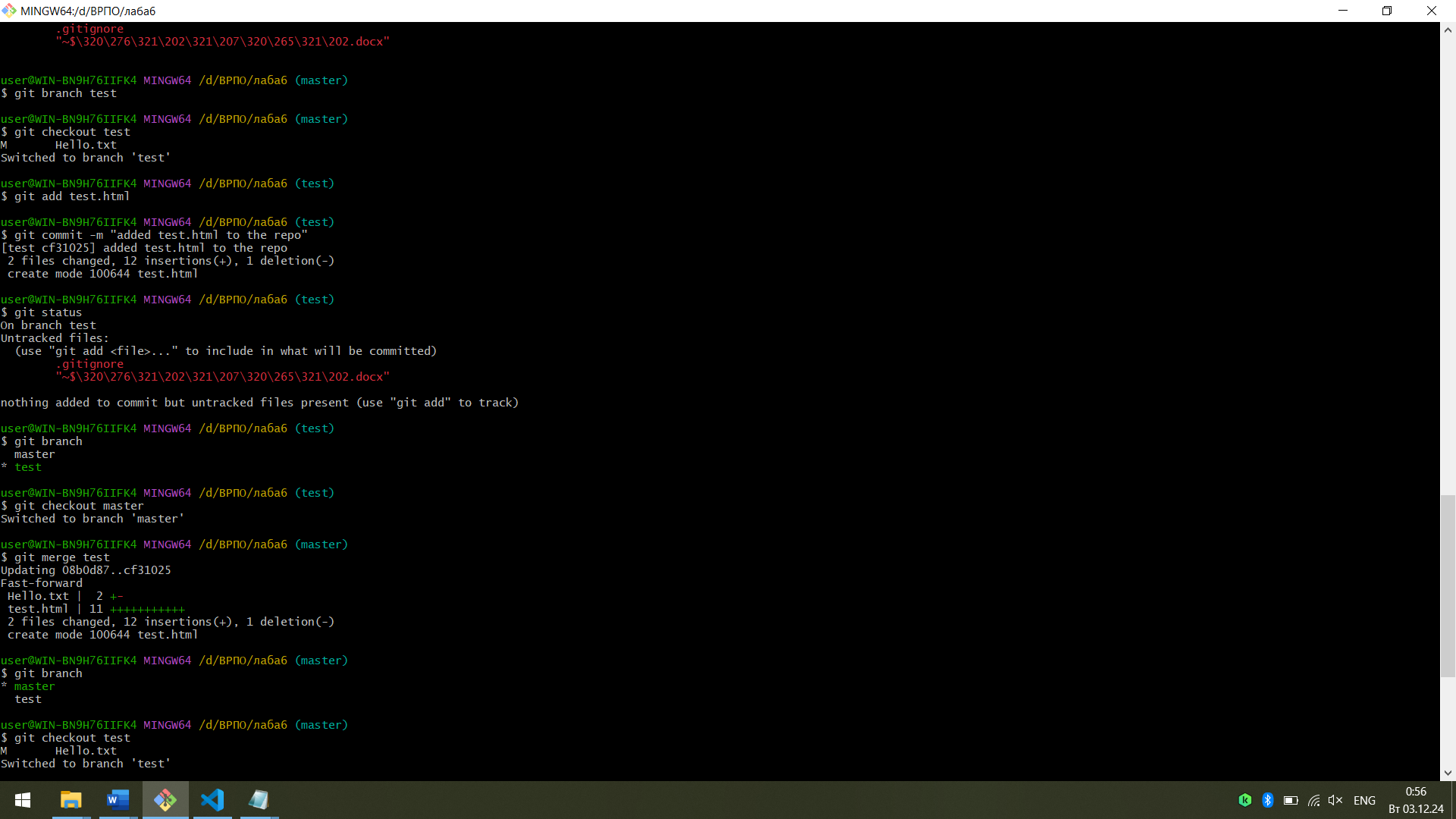
1) С помощью команды **git branch test** создаём новую ветку с именем **test**



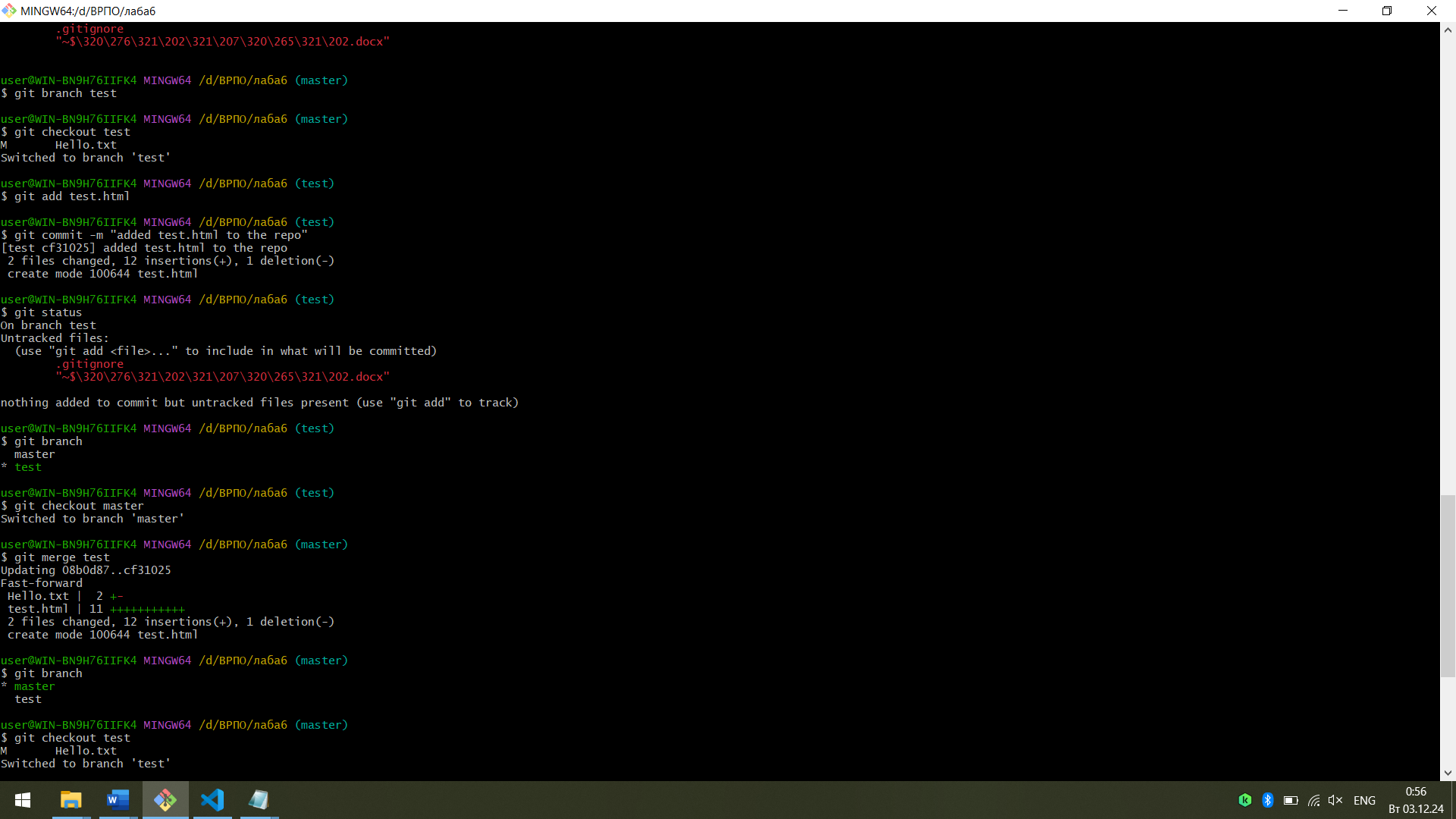
2) Переключим ветки с помощью команды **git checkout test**



Создаём новый файл **test.html** и фиксируем его



C помощью команды **git branch** просмотрим список всех веток

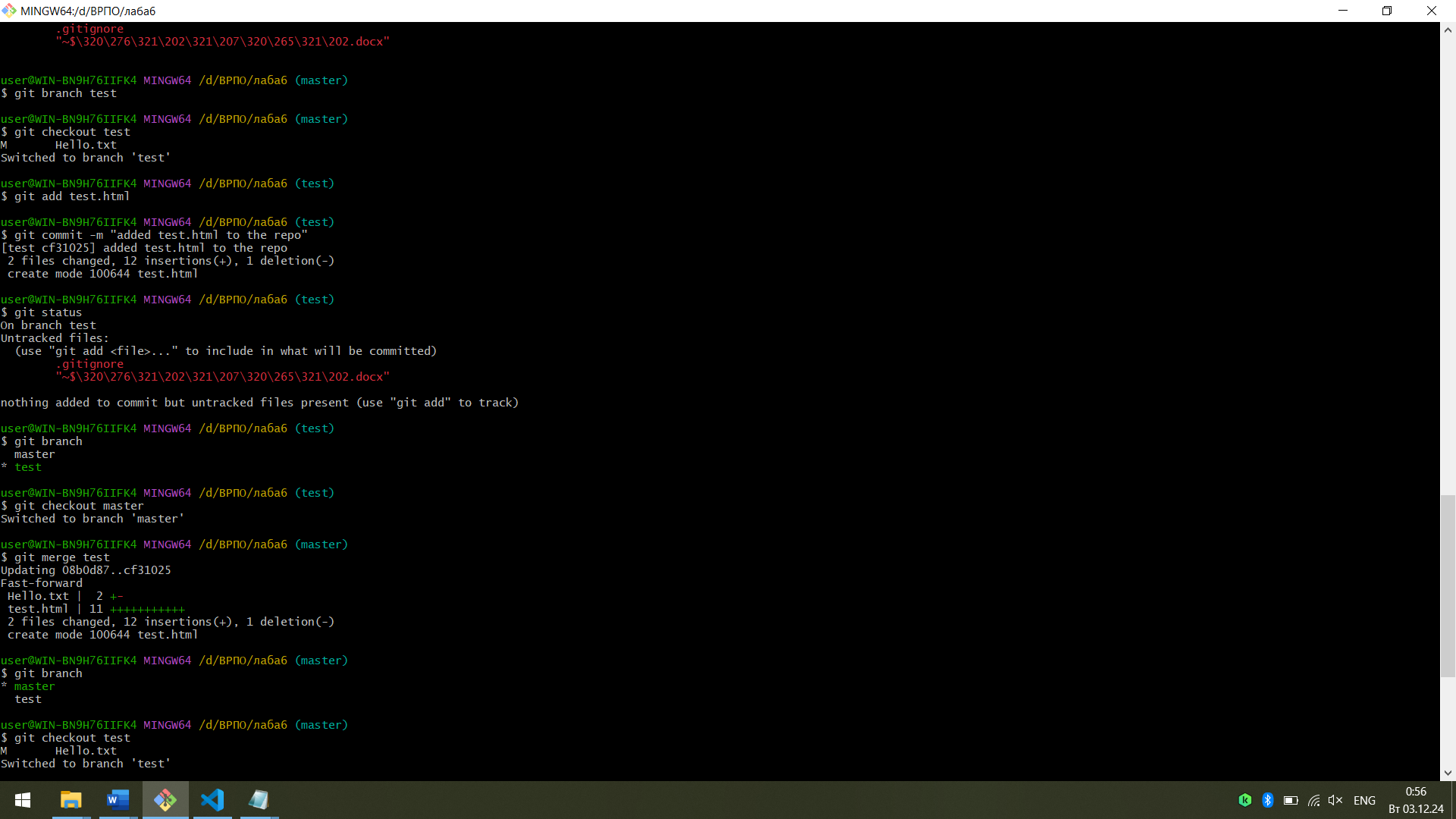


Текущей веткой являет **test**

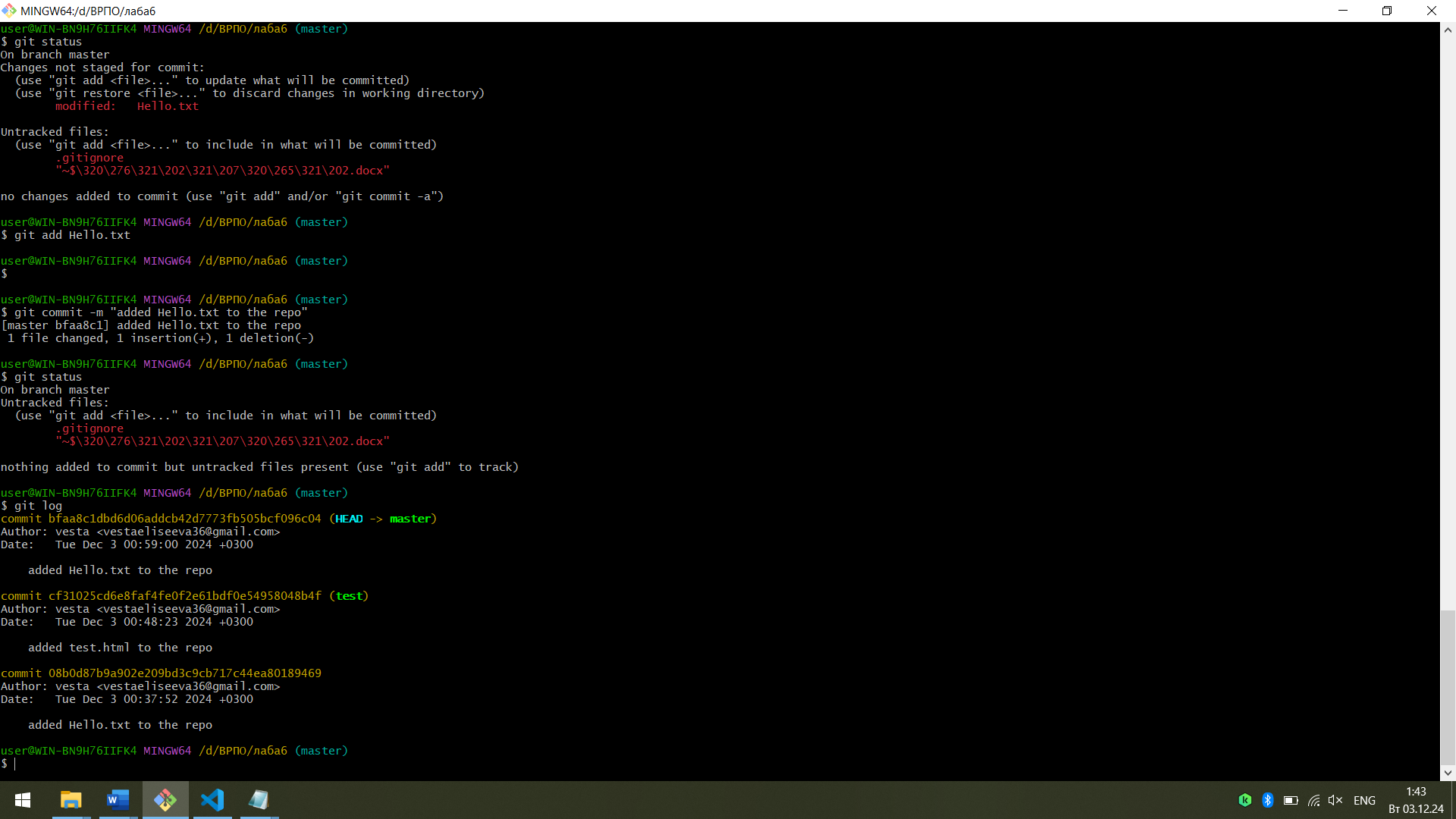
Выполним слияние веток с помощью команды **git merge**

1) Переходим в ветку в которой будем выполнять слияние, а именно **master**

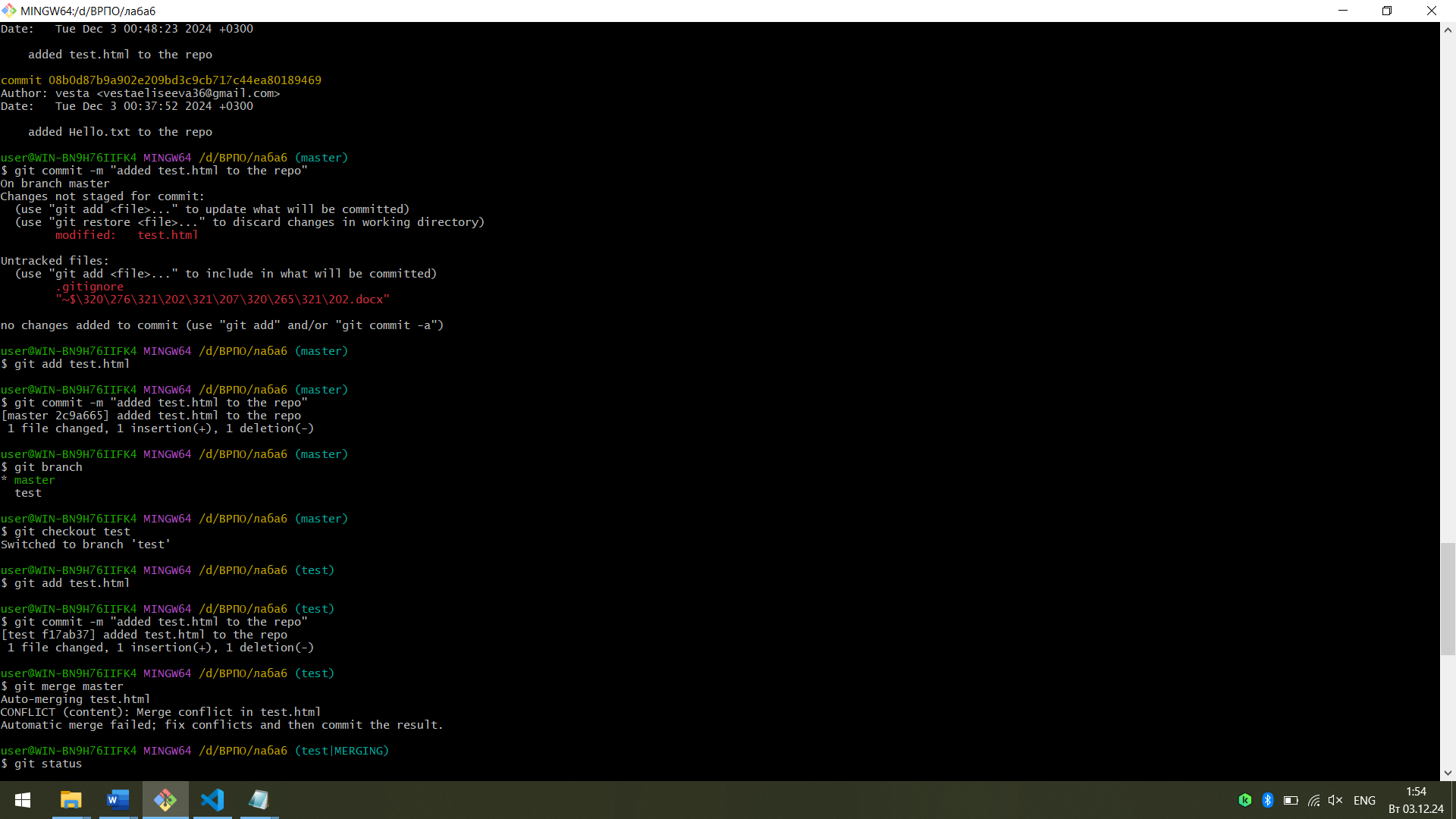
1.1) Для этого используем команду **git checkout master**



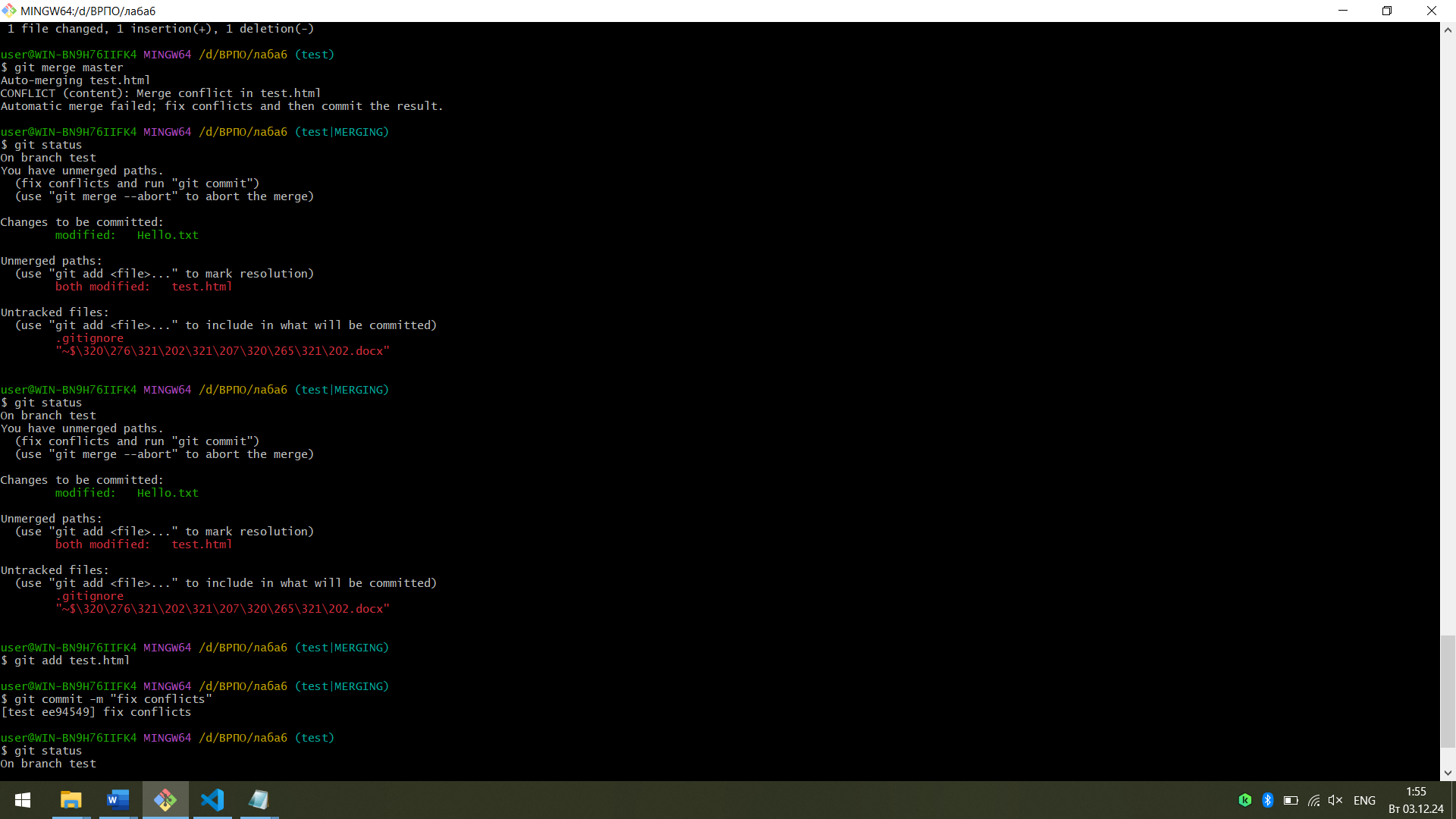
2) Просмотрим статус репозитории



1) По очереди отредактируем файл test.html в ветках master и test. Выполним commit.



Выполним слияние в одной и второй ветках.



Разрешим ситуацию конфликта слияний.

Изменим наш текстовый документ и оставим тот вариант, который нас устраивает, сохраним

