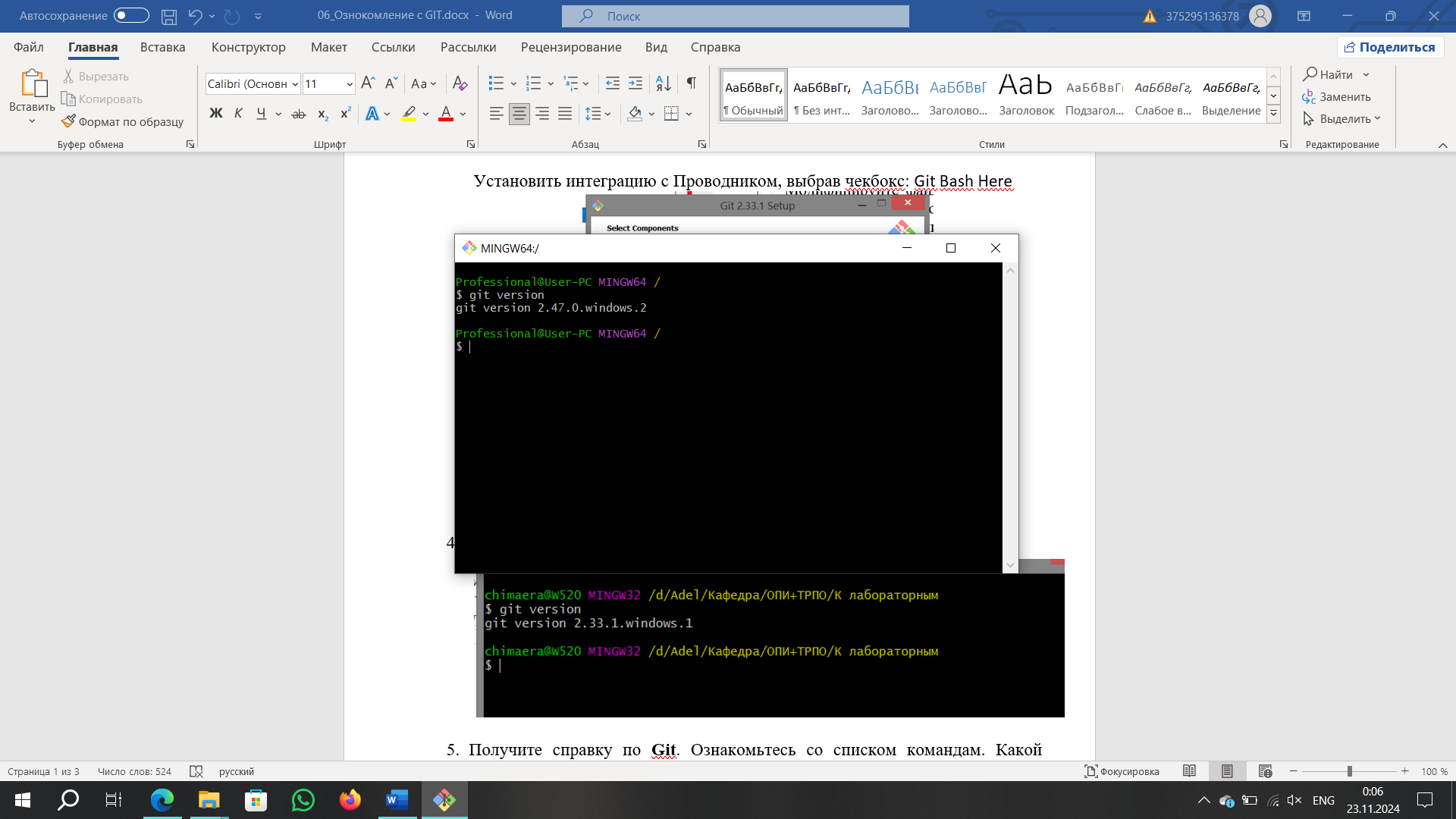
**Лабораторная работа 6**

**Системы контроля версий. Основы работы в Git**

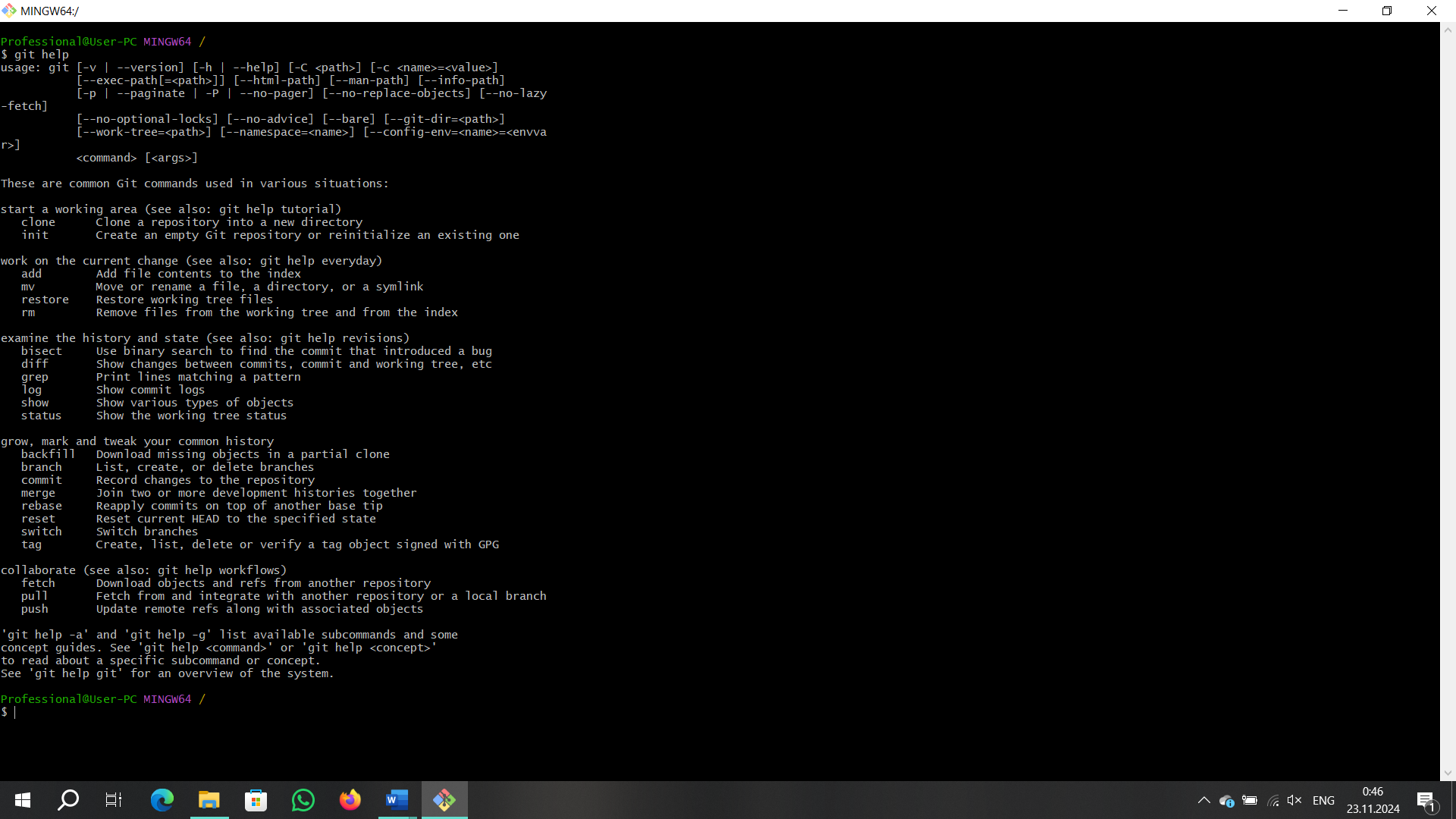
**Цель работы:** ознакомление с основными возможностями системы контроля версий Git.

1. Проверьте установленную версию

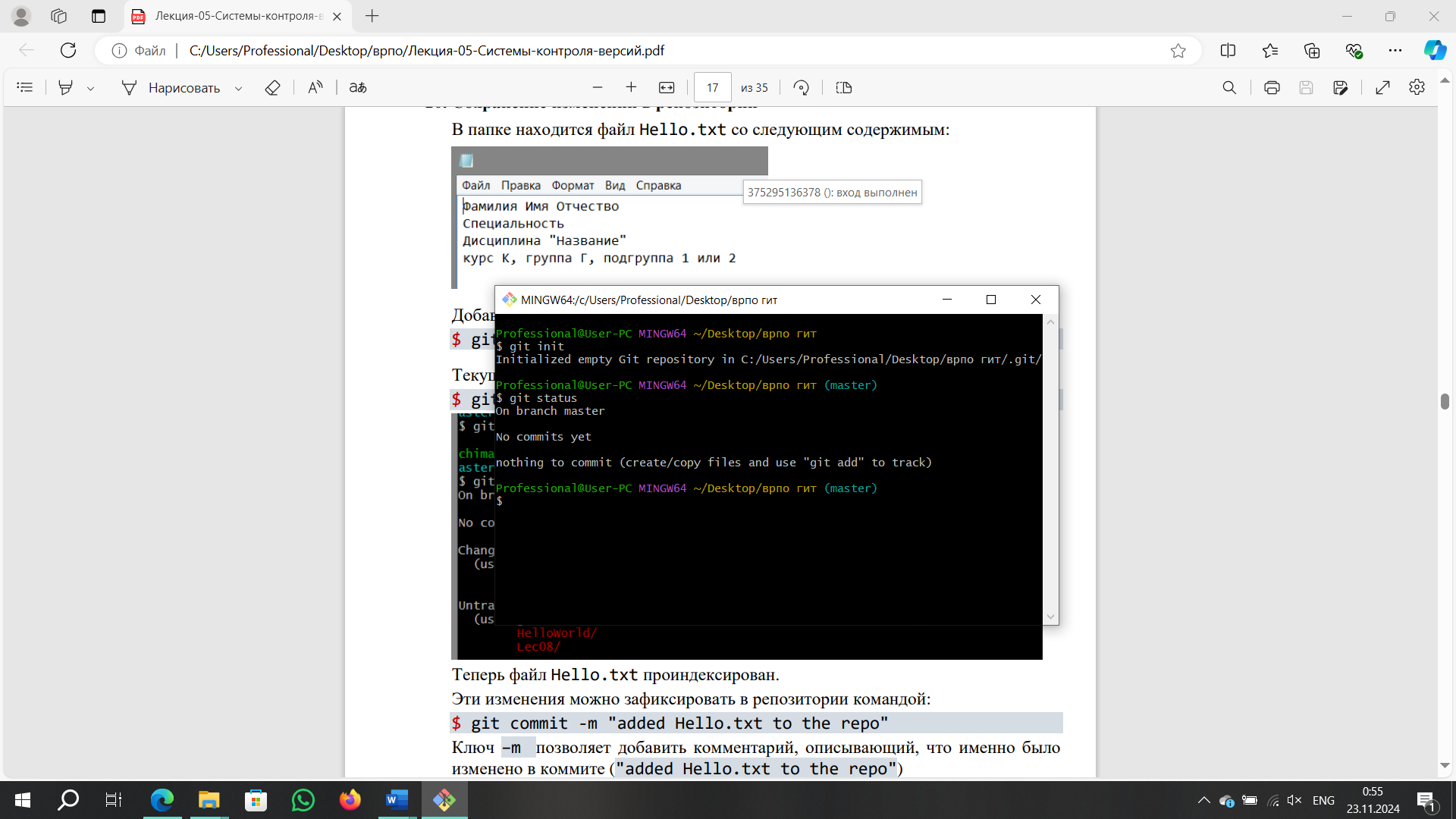


1. Получите справку по **Git**. Ознакомьтесь со списком командам. Какой командой для получения справки вы воспользовались?

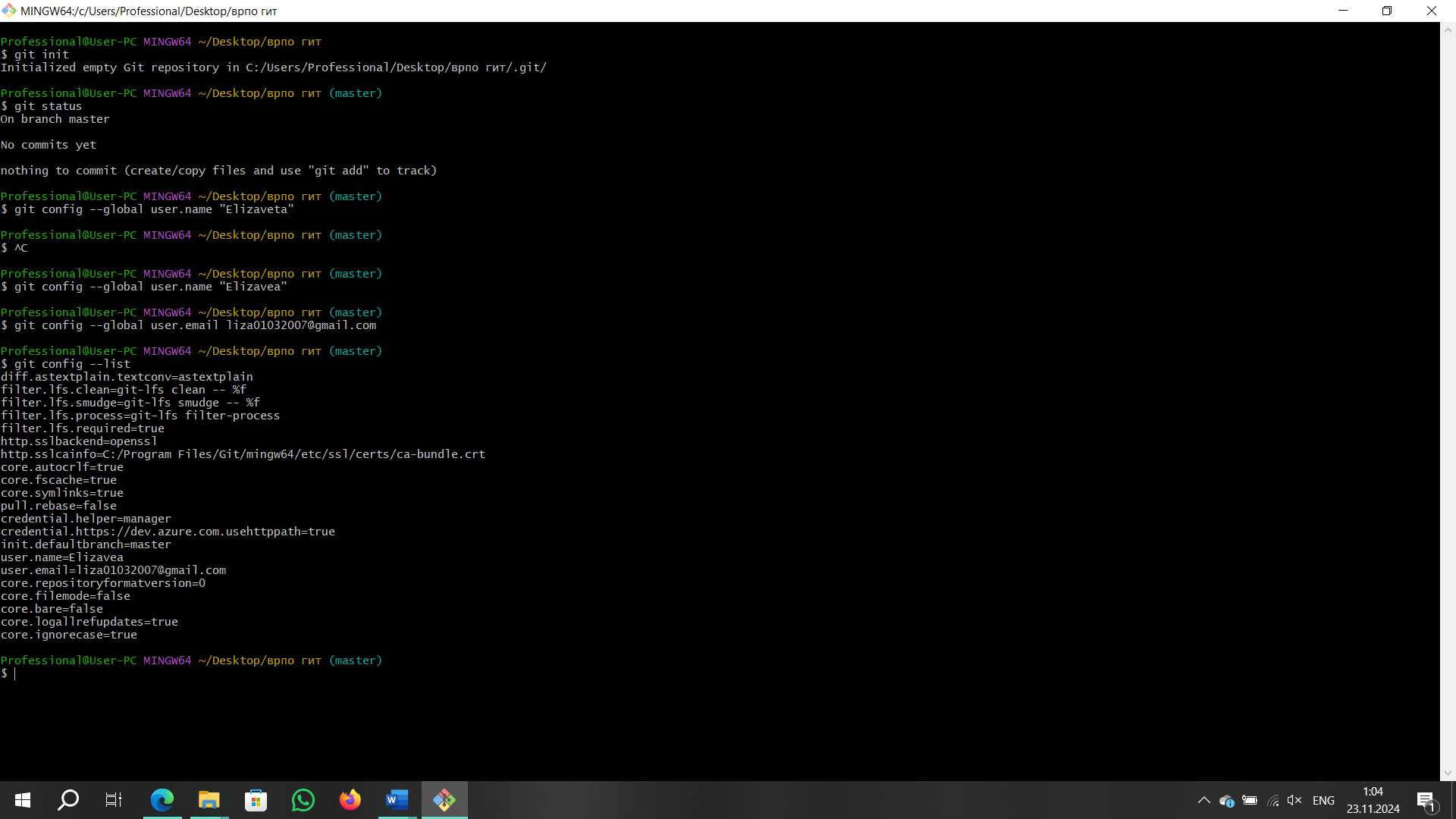
Командой git help



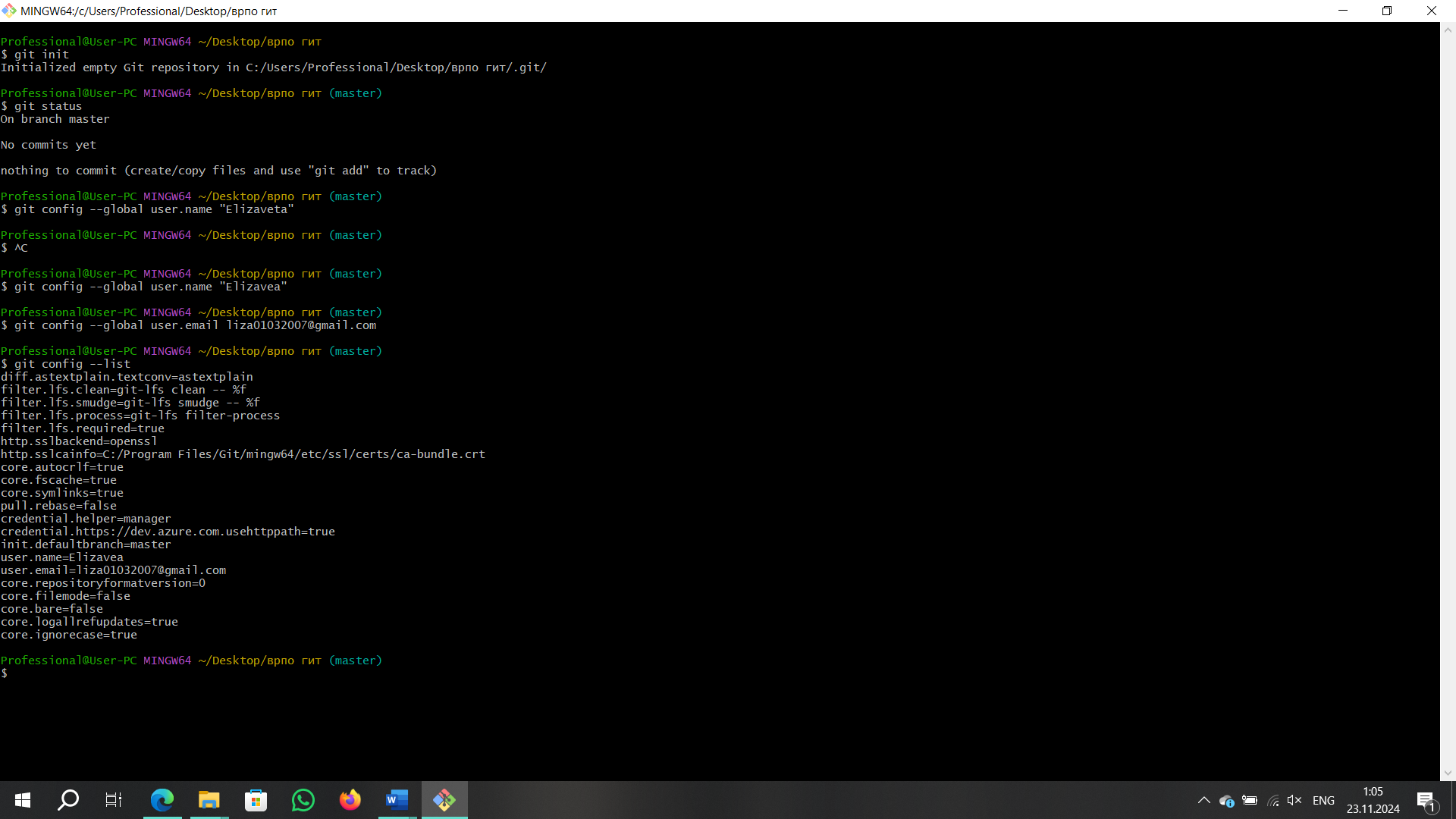
1. Создайте локальный репозиторий. Опишите последовательность действий в отчете.
2. Создайте локальный репозиторий
3. Создаем папку
4. Запускаем Git Bash с помощью контекстного меню: «Git Bash Here»
5. Инициализируем репозитоий с помощью команды git init



5. Выполните конфигурацию гит



6. Проверить настройки конфигурационного файла командой git config --list



7. Инициализируйте локальный репозиторий в рабочей папке (см. лекцию), где будет он будет размещен.

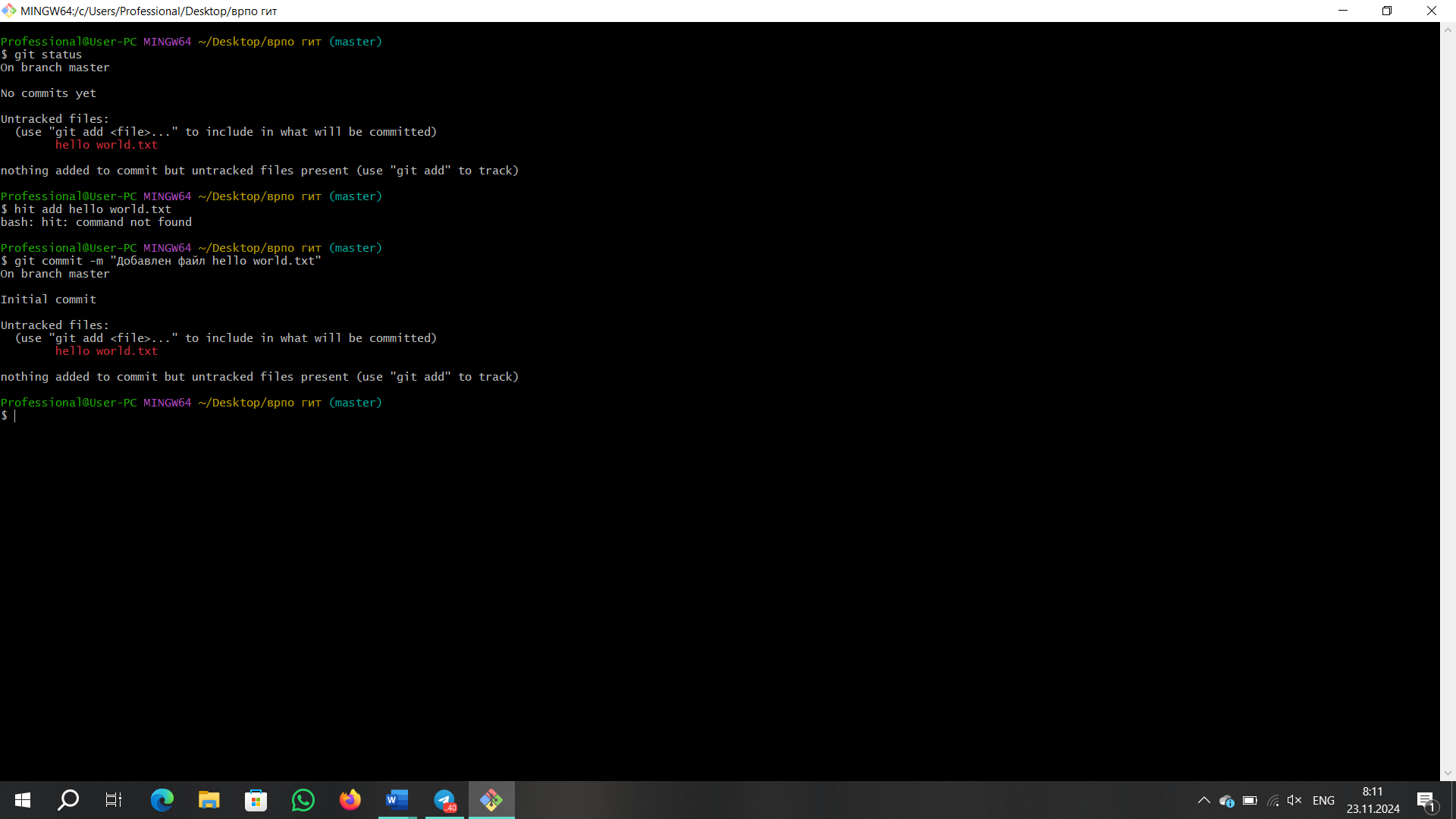
1. Инициализация репозитория: Выполнена команда git init, создавшая новый репозиторий в указанной папке.

2. Создание файла: Создан файл HelloWorld.txt

3. Проверка статуса: Выполнена команда git status, которая показала неотслеживаемый файл.

4. Добавление файла в индекс: Выполнена команда git add HelloWorld.txt

5. Фиксация изменений: Выполнена команда git commit -m "Добавлен файл HelloWorld.txt".



8. Научитесь получать информацию о статусе (status) репозитория. Какая информация отображается в терминальном окне GitBash? Прокомментируйте вывод команды в отчете

Вывод команды git status

1.Текущая ветка:

2. Неотслеживаемые файлы:

3.Измененные файлы

4.Подготовленные файлы:

5. Указания по действиям

• Текущая ветка: Мы находимся на ветке branch, но еще не сделали ни одного коммита.

• Неотслеживаемые файлы: В репозитории есть файл hellow world.txt, который не отслеживается Git. Это значит, что он был создан, но еще не добавлен в индекс.

• Указания по действиям: Git предлагает использовать команду git add, чтобы начать отслеживать файл. Это полезно для новых пользователей, так как подсказывает, что делать дальше.

9. Проиндексируйте файл Hello.txt Какие изменения произошли в репозитории

Файл будет добавлен в индекс:

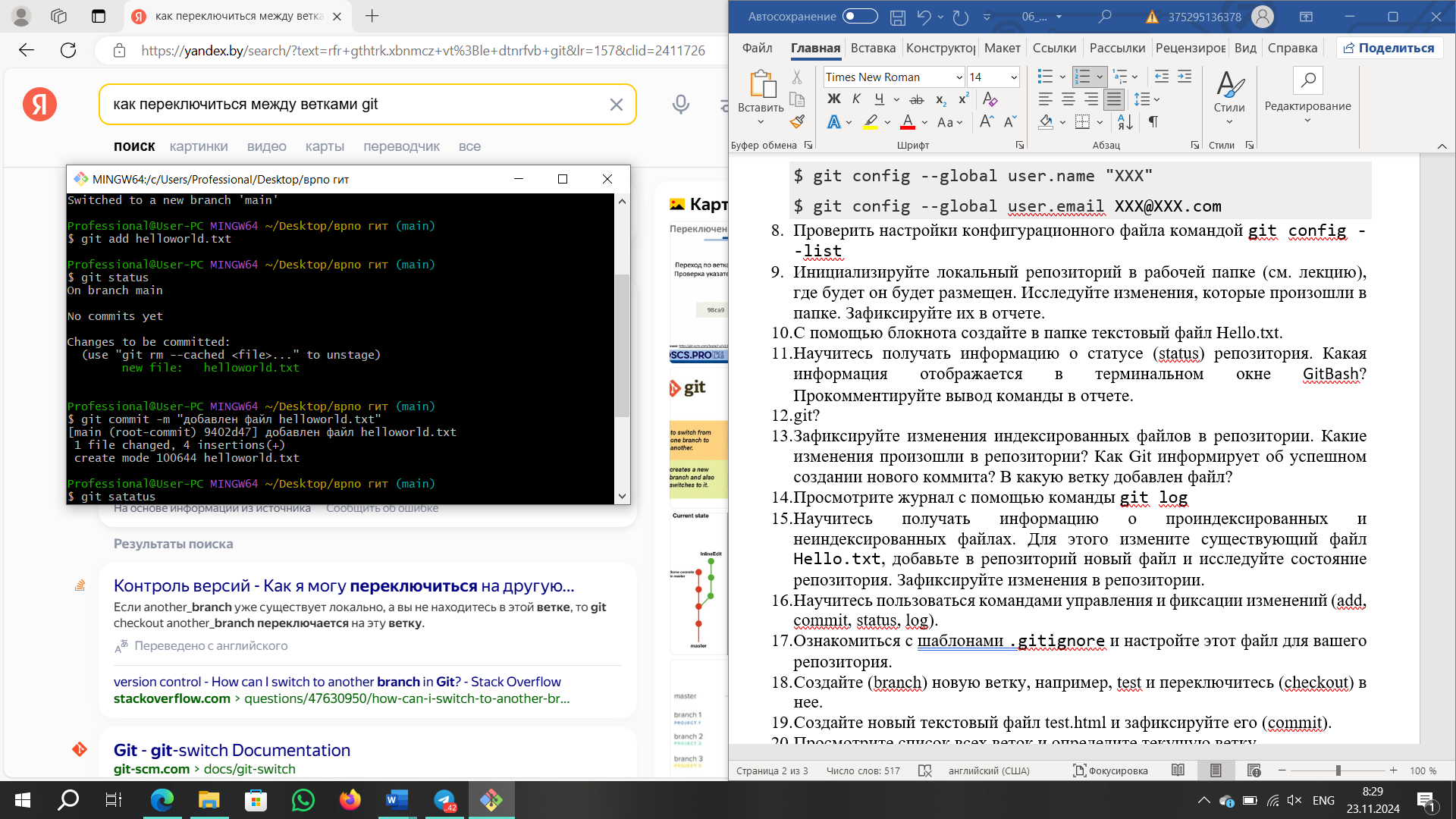
• Это означает, что изменения в файле Hello.txt теперь отслеживаются Git и готовы к коммиту. Файл будет перемещен из состояния "неотслеживаемый" или "измененный" в состояние "подготовленный".

2. Изменение статуса репозитория:

• Если вы снова выполните команду git status, вы увидите, что Hello.txt теперь находится в разделе "Changes to be committed".

3. Готовность к коммиту:

• Теперь файл готов к коммиту. Вы можете зафиксировать изменения с помощью команды:



10. Зафиксируйте изменения индексированных файлов в репозитории. Какие изменения произошли в репозитории? Как Git информирует об успешном создании нового коммита? В какую ветку добавлен файл

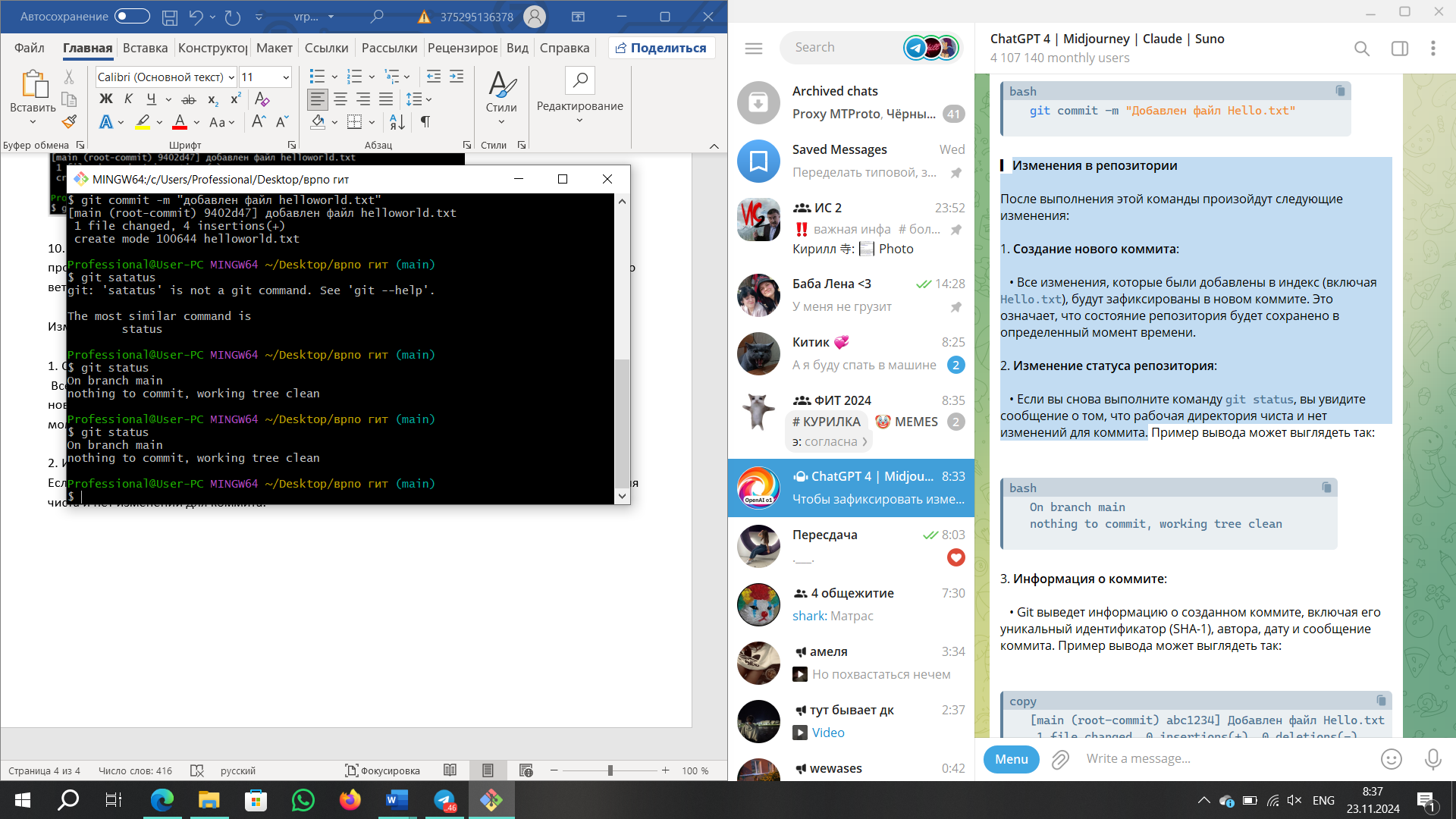
Изменения в репозитории:

1. Создание нового коммита:

Все изменения, которые были добавлены в индекс (включая Hello.txt), будут зафиксированы в новом коммите. Это означает, что состояние репозитория будет сохранено в определенный момент времени.

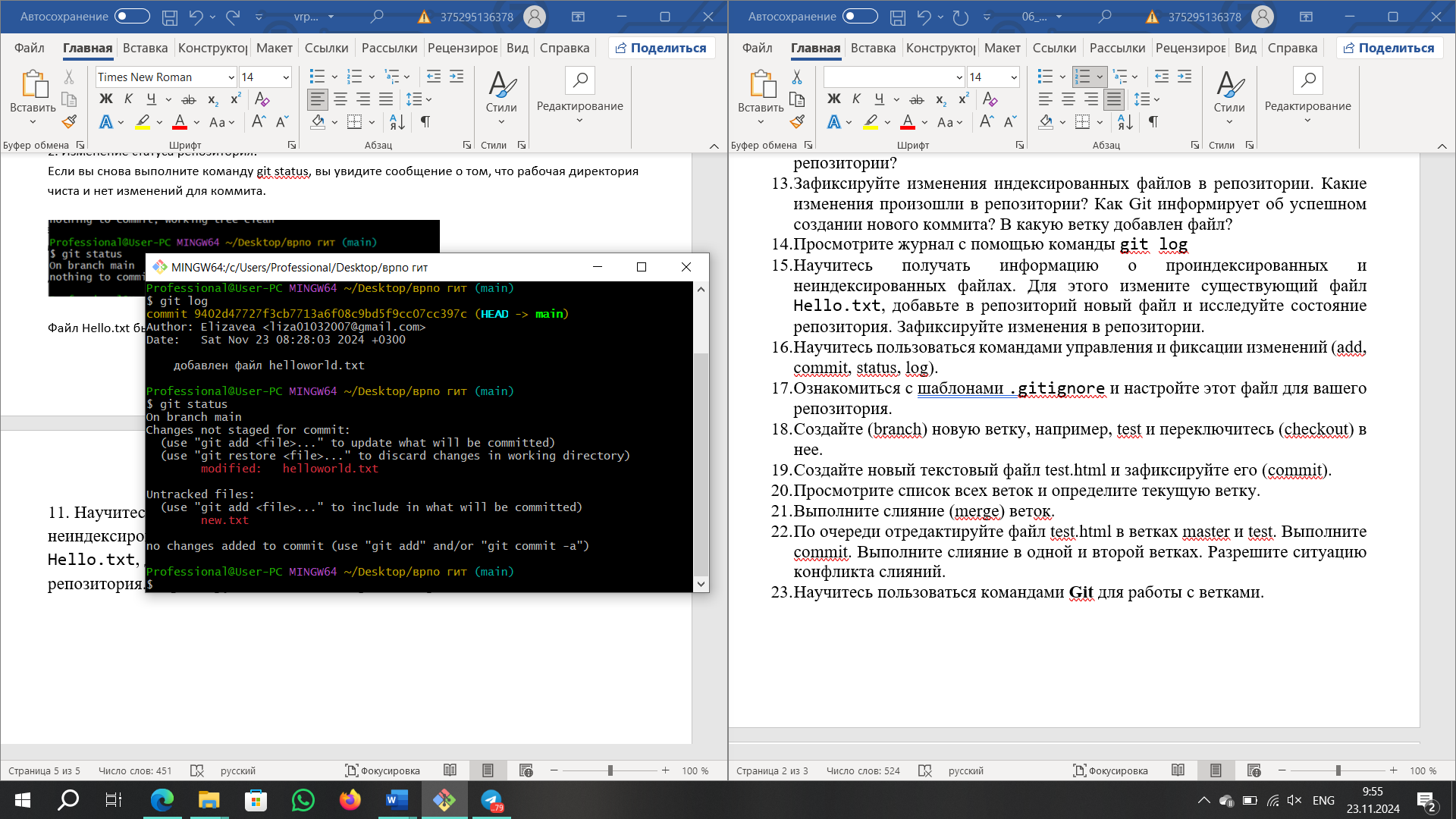
2. Изменение статуса репозитория:

Если вы снова выполните команду git status, вы увидите сообщение о том, что рабочая директория чиста и нет изменений для коммита.



Файл Hello.txt был добавлен в ветку main

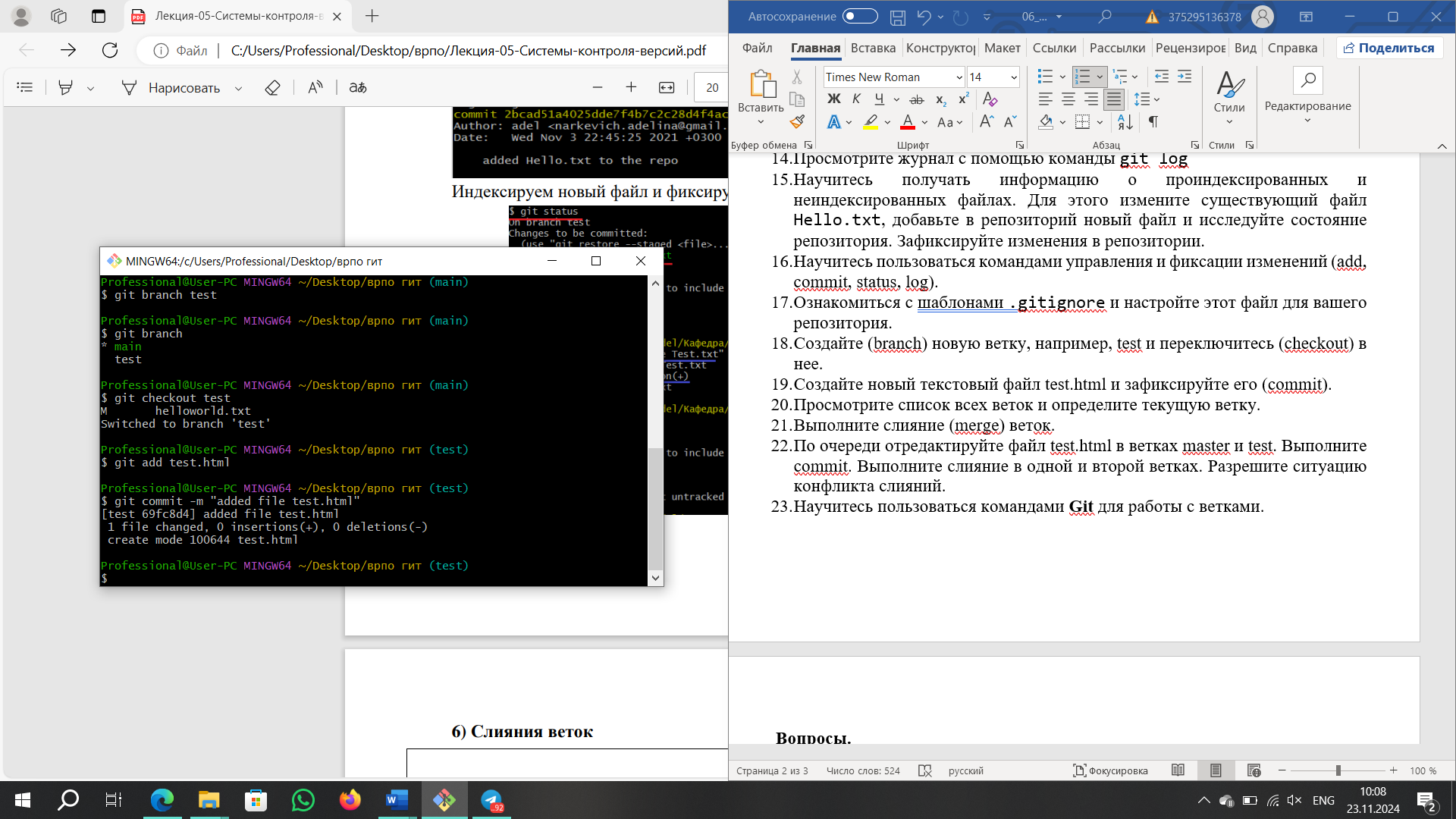
11. Научитесь получать информацию о проиндексированных и неиндексированных файлах. Для этого измените существующий файл Hello.txt, добавьте в репозиторий новый файл и исследуйте состояние репозитория. Зафиксируйте изменения в репозитории



В репозитории стало отображаться изменение файла hello.txt и добавление файла new.txt и то что они не были еще индексированны

12. Создайте (branch) новую ветку, например, test и переключитесь (checkout) в нее.

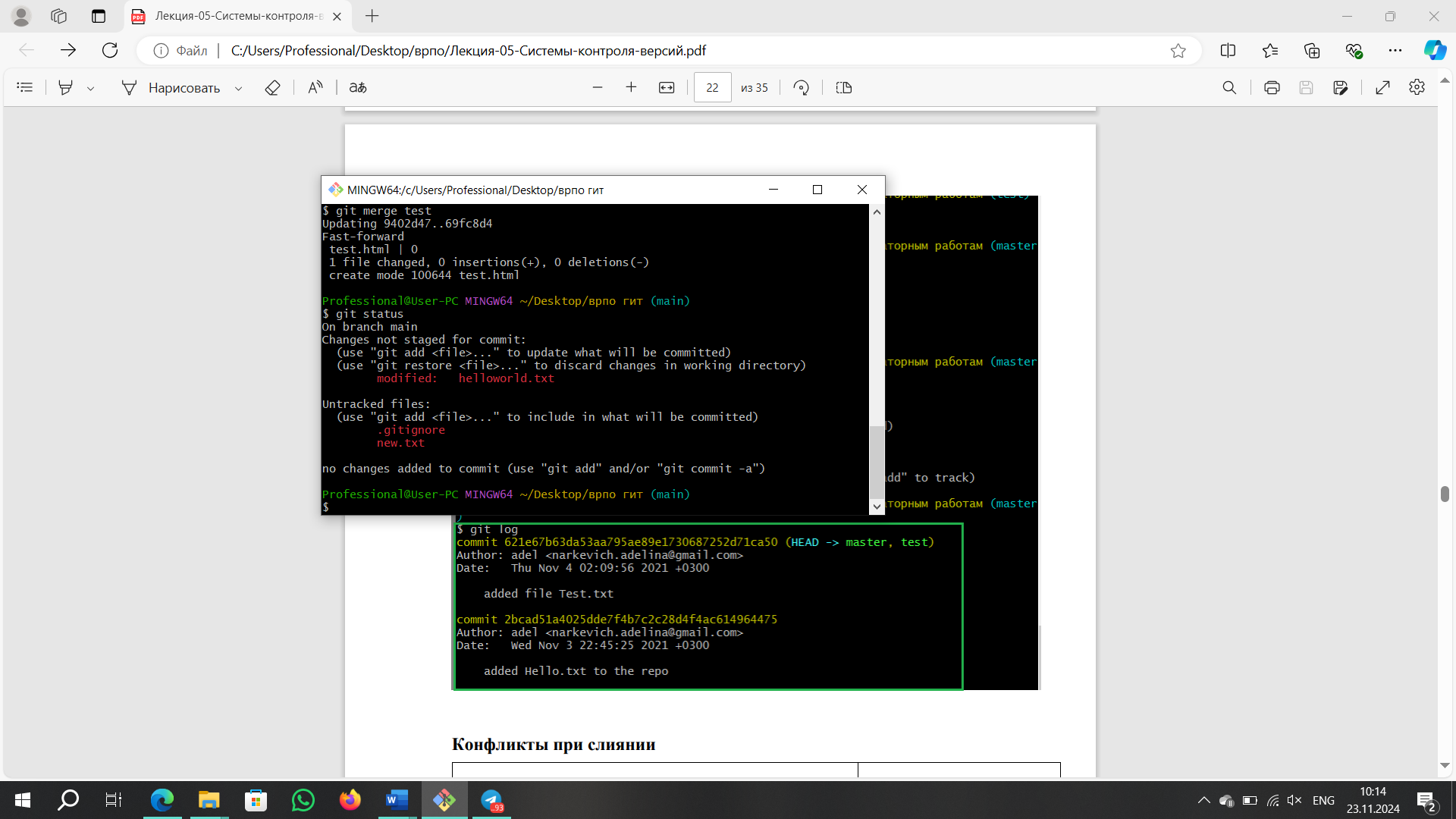
13. Создайте новый текстовый файл test.html и зафиксируйте его (commit).



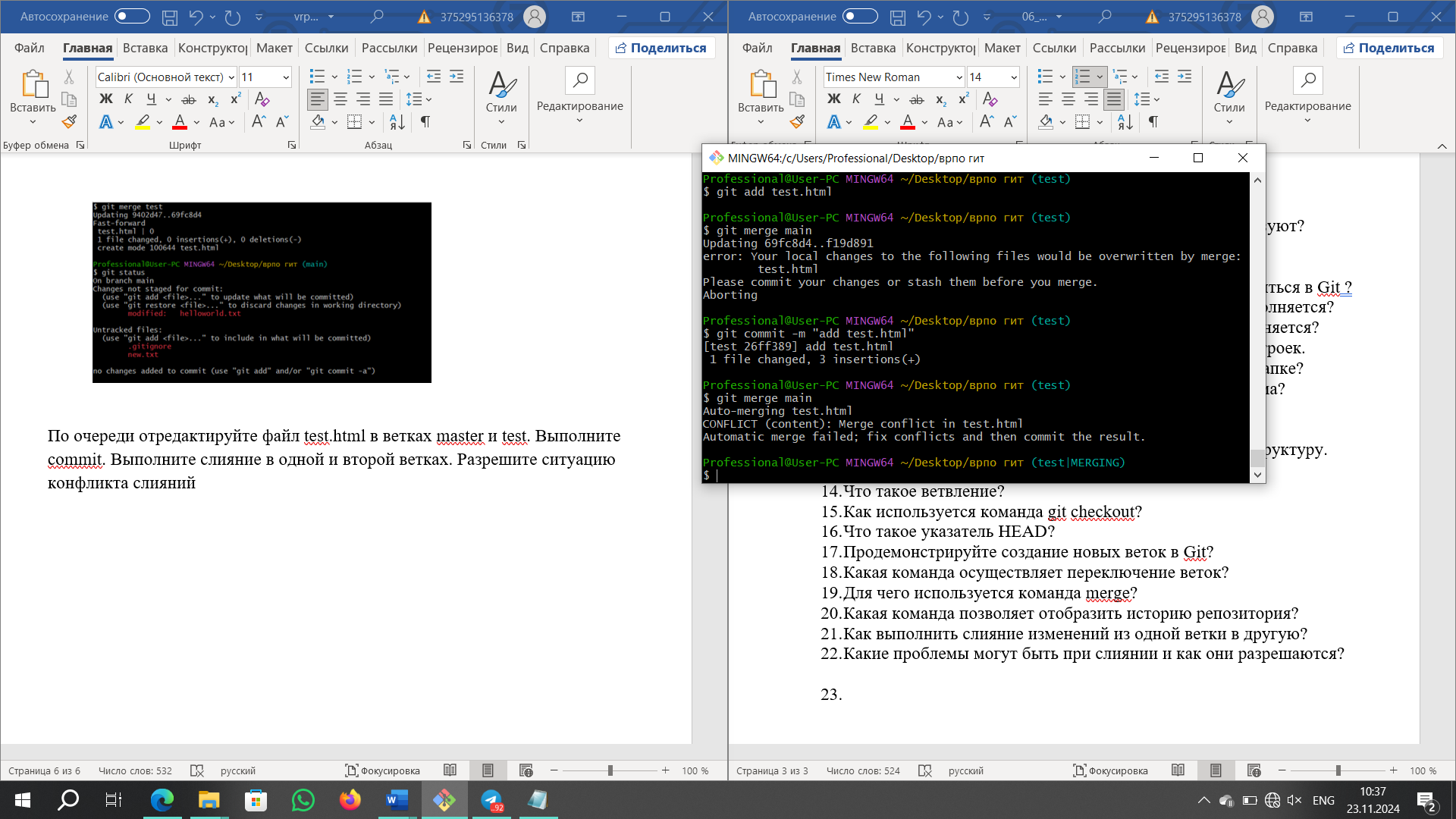
14. Просмотрите список всех веток и определите текущую ветку

Текущая ветка test

15. Выполните слияние (merge) веток



По очереди отредактируйте файл test.html в ветках master и test. Выполните commit. Выполните слияние в одной и второй ветках. Разрешите ситуацию конфликта слияний



Решение конфликта

