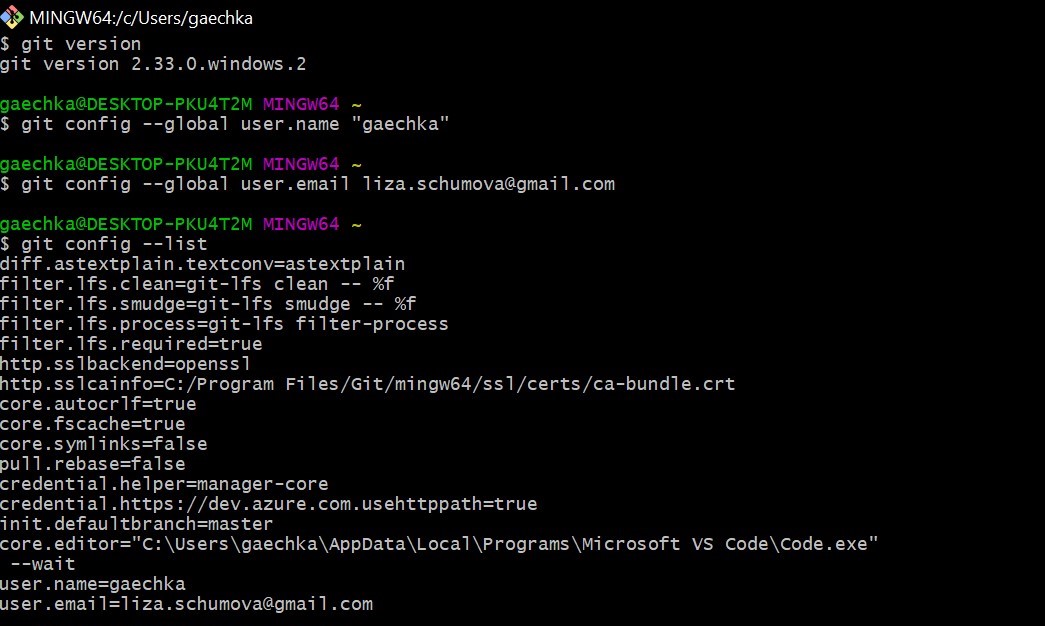
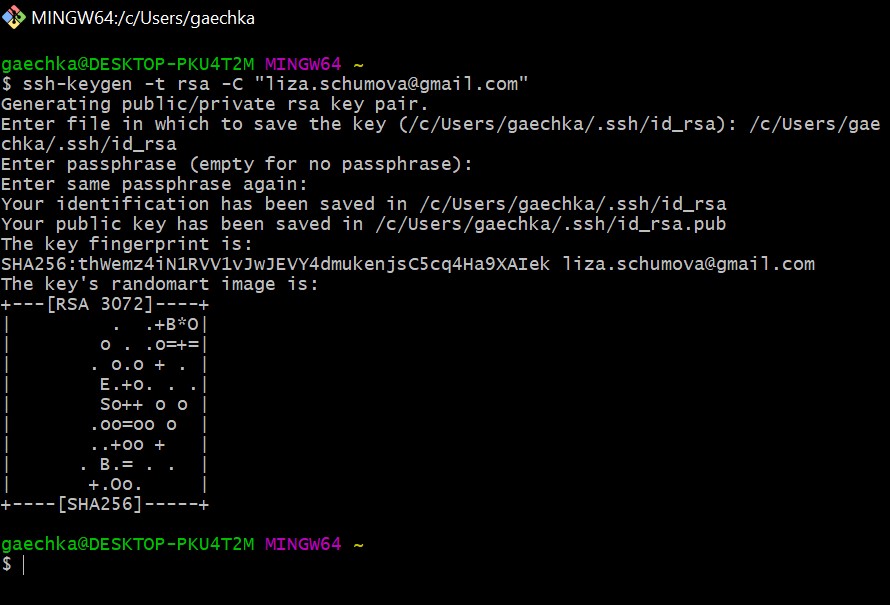
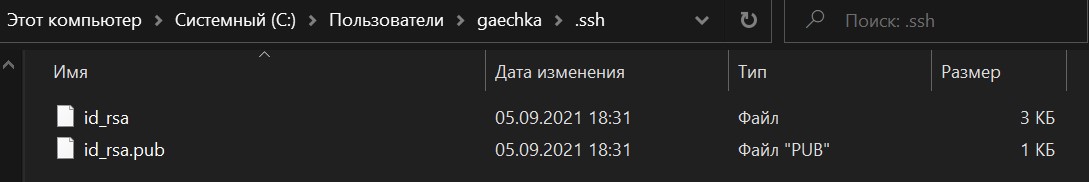
**Настройка конфигурации**



**Создание ключей**

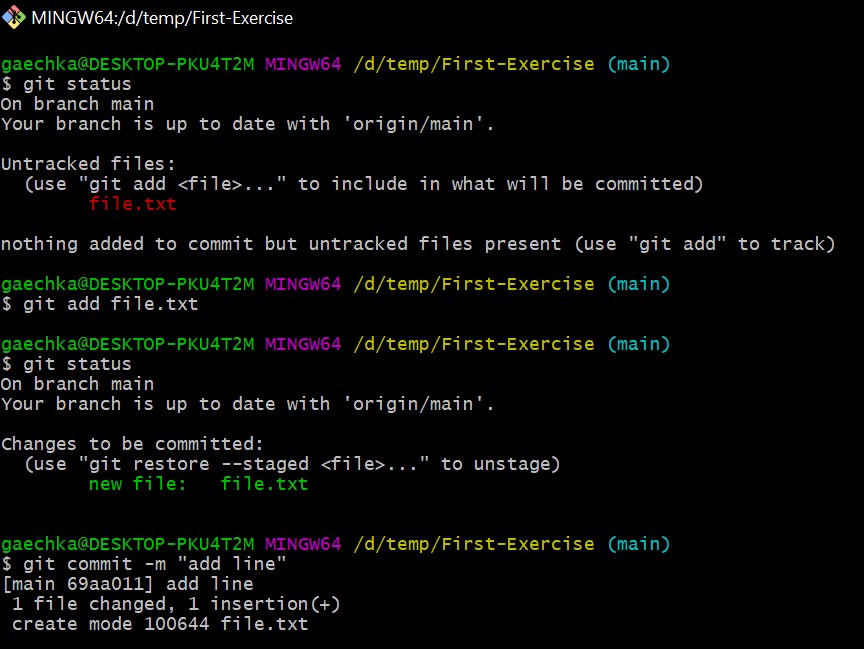




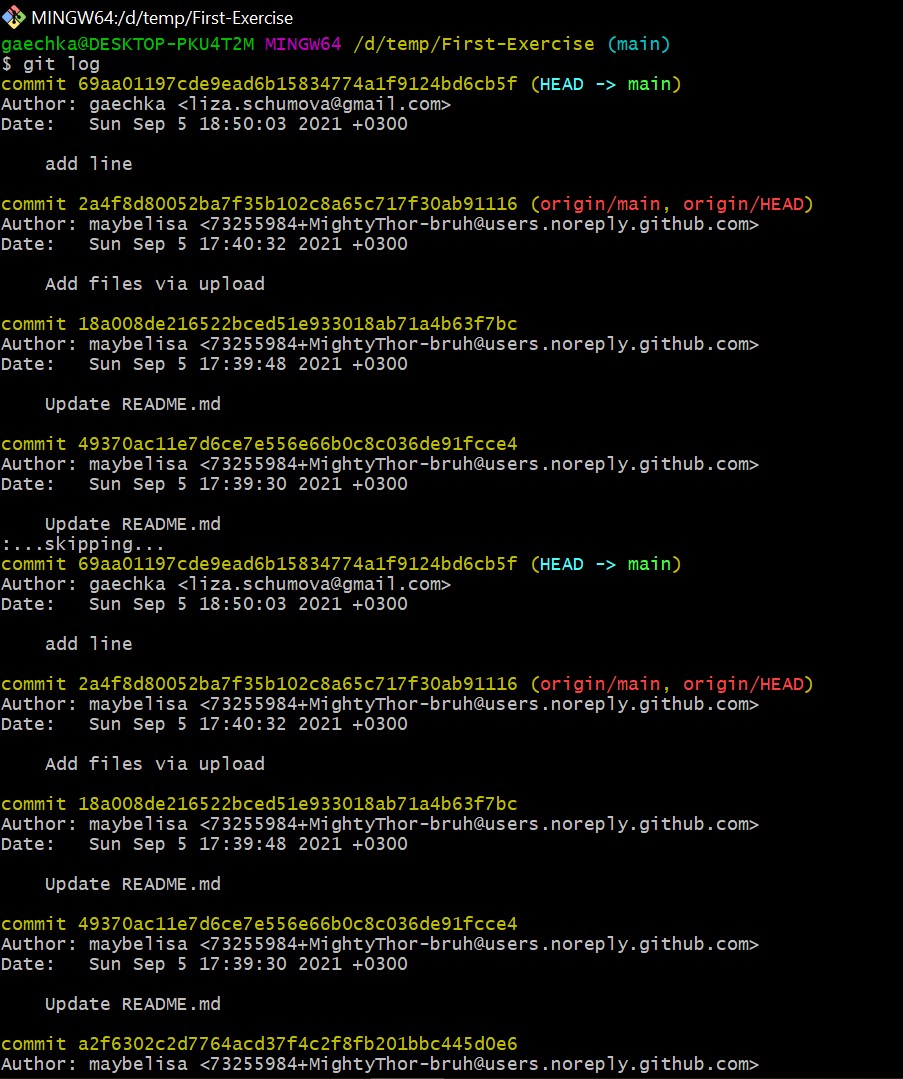
**Клонирование репозитория**



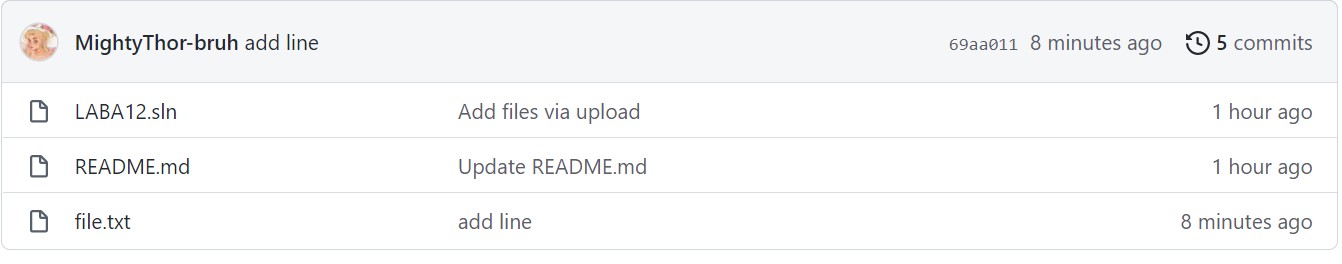
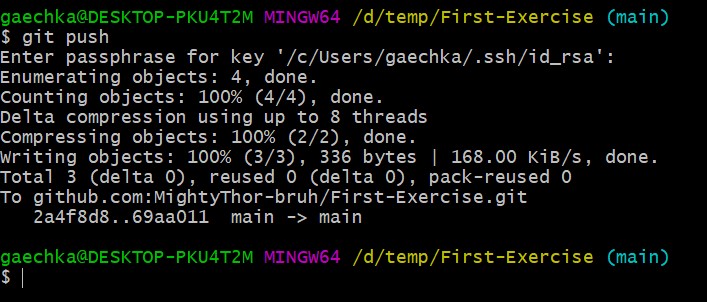
**Определение статуса. Для определения статуса был добавлен файл file.txt**



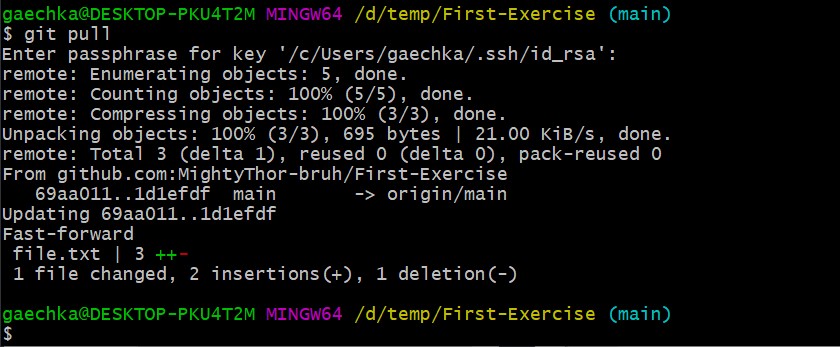
**Log**



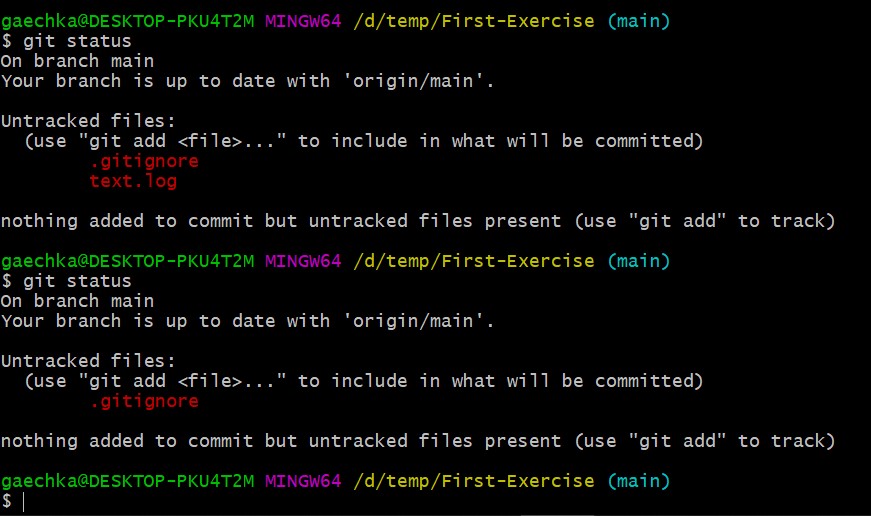
**Push**



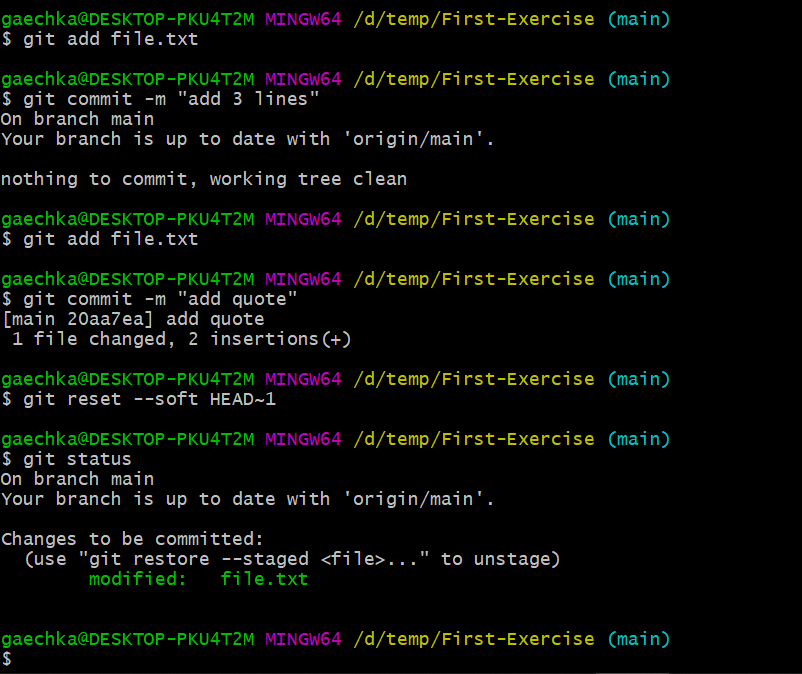
**Pull**



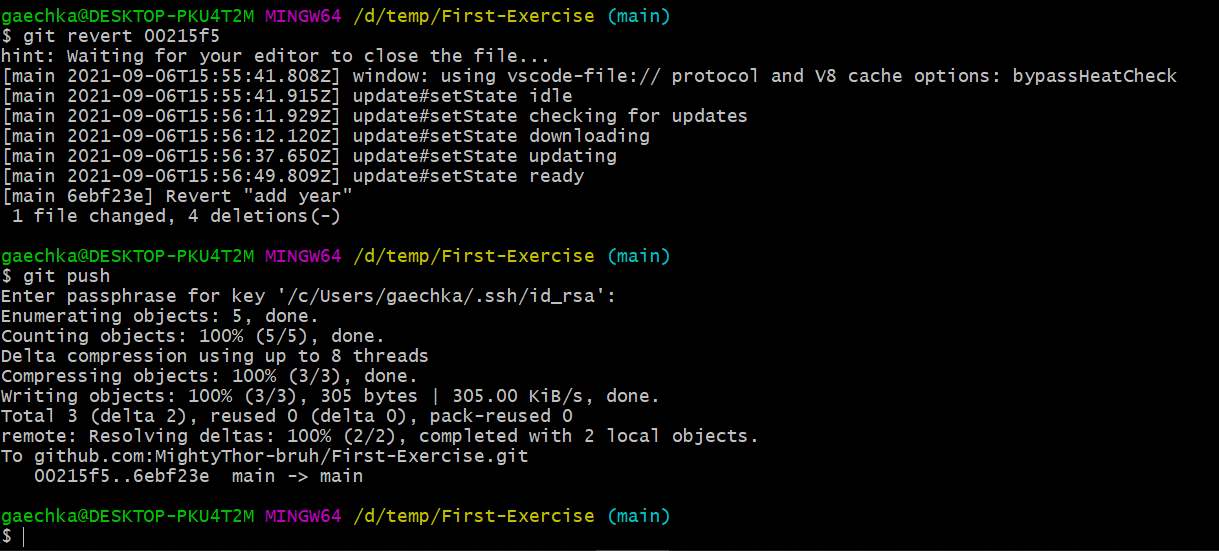
**Ignore**

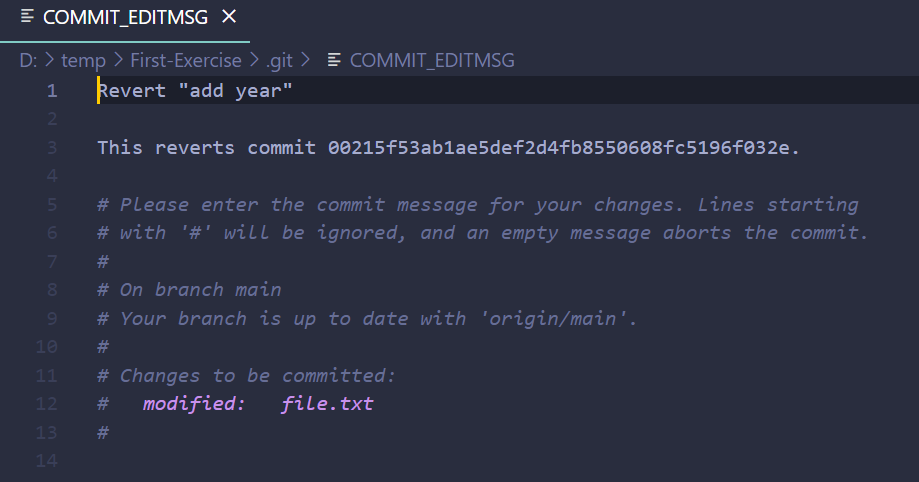


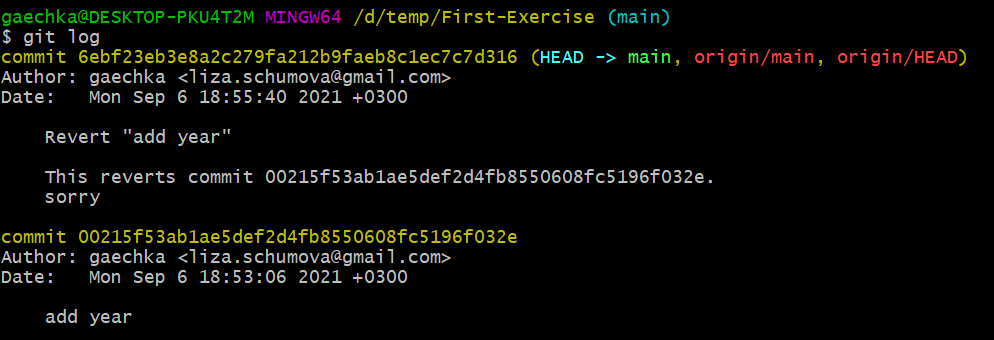
**Reset**



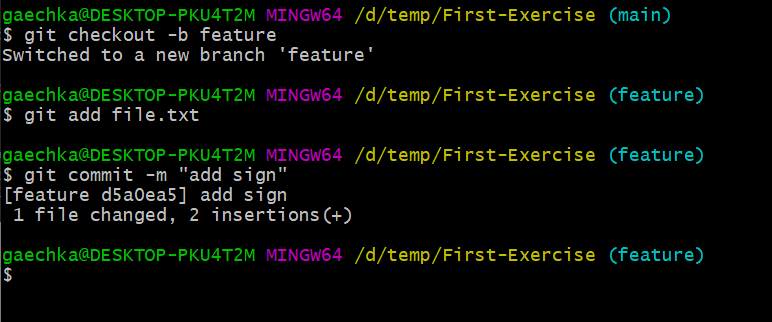
**Revert**

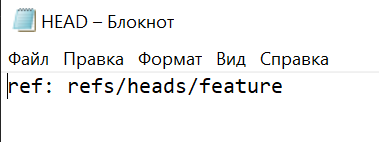


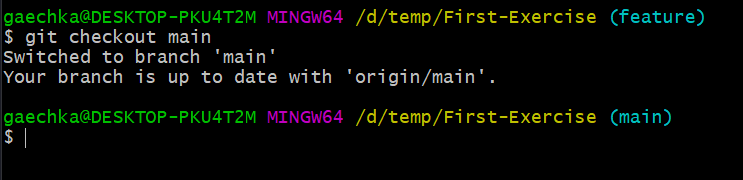




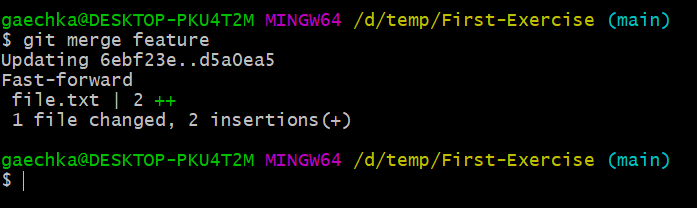
**Branch**

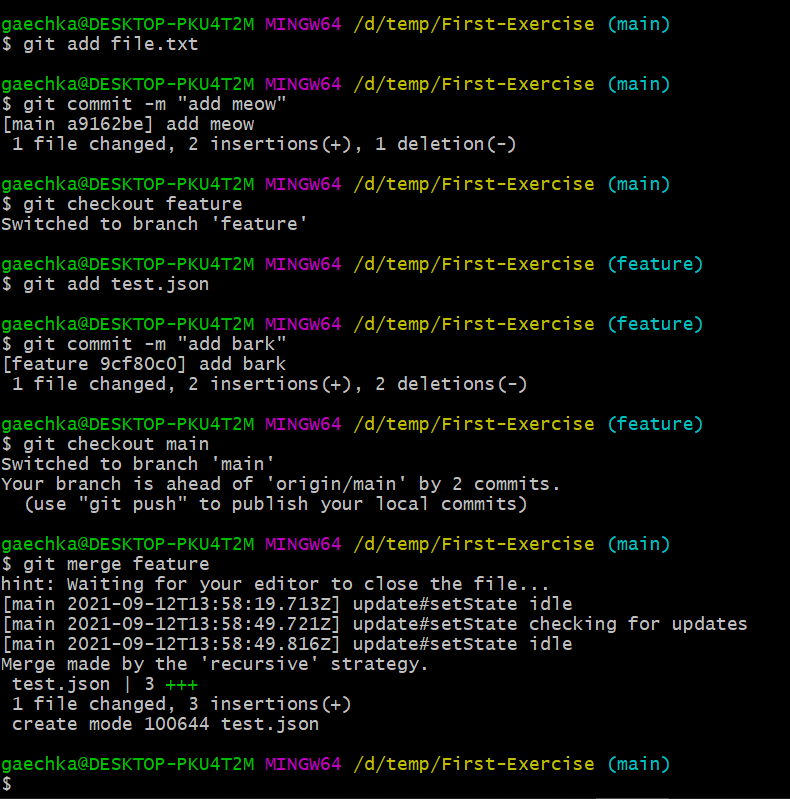


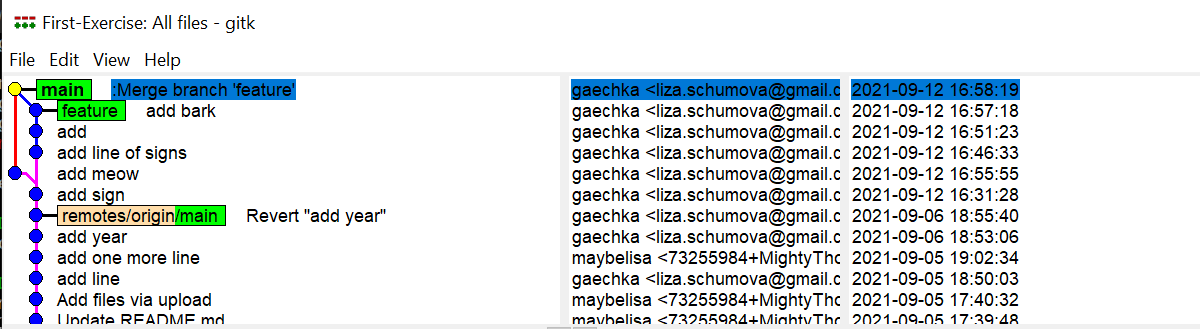




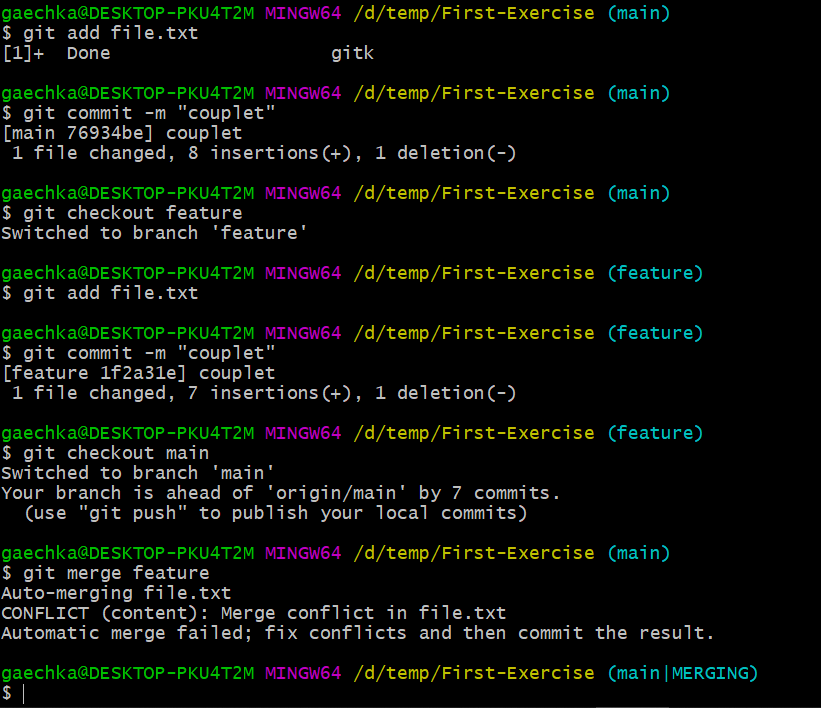
**Merge**

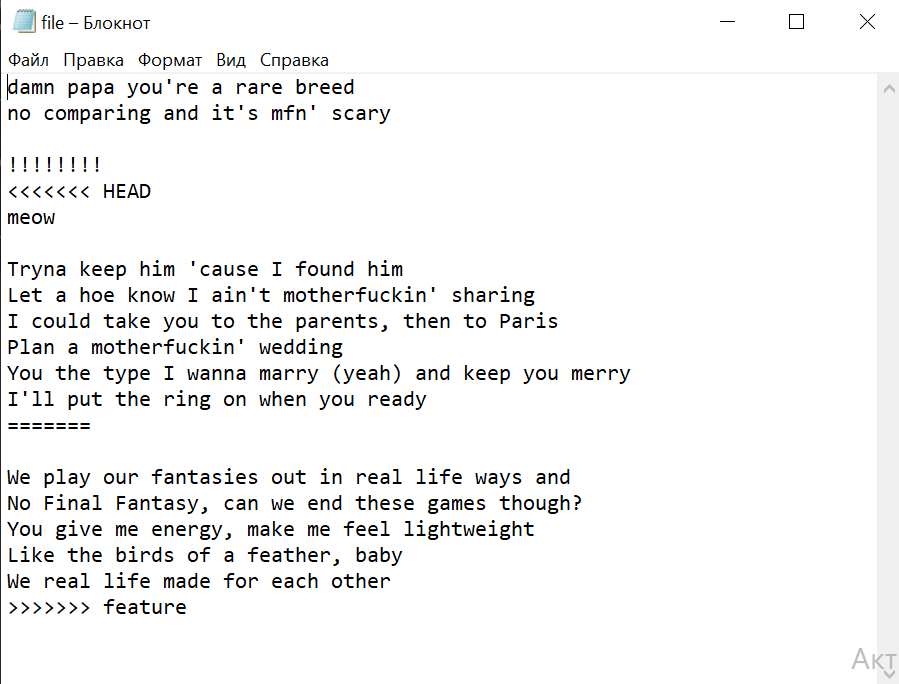


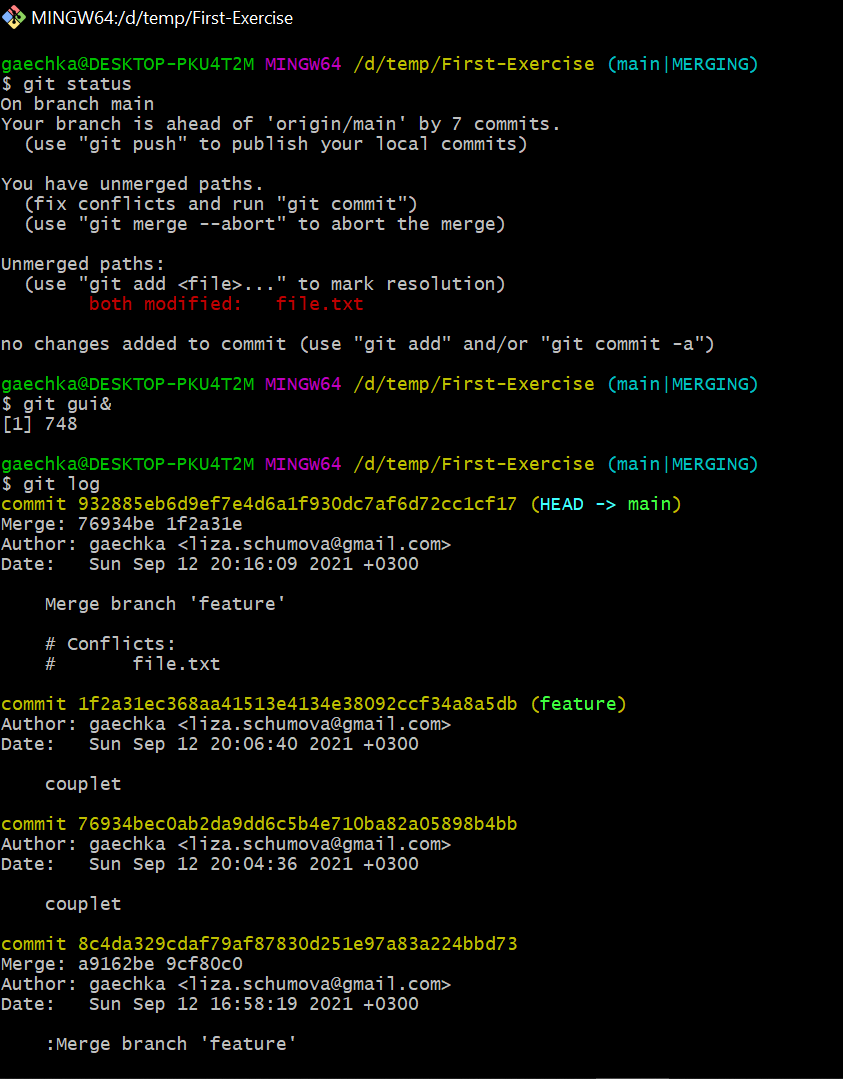




**Разрешение конфликтов**







**Вопросы:**

**1. Что такое система контроля версий и для чего её используют?**

Контроль версий — это практика отслеживания изменений программного кода и управления ими. Системы контроля версий — это программные инструменты, помогающие командам разработчиков управлять изменениями в исходном коде с течением времени.

**2. Какой принцип хранения файлов использует Git?**

В Git единицей хранения данных является объект.

**3. В чем отличие git от других систем контроля версий?**

**4. В каких трех основных состояниях файлы могут находиться в Git?**

В Git файлы могут находиться в одном из трёх состояний: зафиксированном, изменённом и подготовленном. "Зафиксированный" значит, что файл уже сохранён в локальной базе. К изменённым относятся файлы, которые поменялись, но ещё не были зафиксированы. Подготовленные файлы — это изменённые файлы, отмеченные для включения в следующий коммит.

**5. Что такое индексация файла? Какой командой она выполняется?**

Git add

**6. Что такое фиксация файла? Какой командой она выполняется?**

**Коммит, commit**

**7. Продемонстрируйте команду проверки выбранных настроек.**

Git config

**8. Как инициализировать репозиторий в существующей папке?**

**9. Как указать файлы, за которыми должна следить система?**

Проиндексировать их

**10. Как выполнить фиксацию изменений?**

**11. Какой командой определить состояния файлов?**

Git status

**12. Для чего создается файл .gitignore? Поясните его структуру.**

**13. Как используется команда git diff?**

Для сравнения изменений

**14. Как используется команда git commit?**

**15. Как используется команда git log? Какие у нее есть параметры?**

**16. Как используется команда git commit –amend?**

При передаче этого параметра будет изменен последний коммит. Вместо создания нового коммита проиндексированные изменения будут добавлены в предыдущий коммит.

**17. Как отобразить удаленные репозитории?**

**18. Как извлечь данные из удаленного репозитория?**

Git Pull

**19. Как отправить данные в удаленный репозиторий?**

Git Push

**20. Какая команда позволяет отобразить удаленные репозитории, связанные с текущим локальным?**

**Git remote**

**21. Каким образом можно получить изменения из удаленного репозитория в локальный?**

Git Pull

**22. Для чего используется команда fetch? В чем отличие команды fetch от pull?**

После использования fetch необходимо делать merge. Pull этого не требует.

**23. Для чего используется команда merge? В чем отличие merge от rebase?**

Для слияния веток. Rebase перемещает дочерний коммит к новому родительскому, т.е. «меняет базу».

**24. Что такое pull request?**

Pull Request — это запрос на вливание изменений из моей ветки в основную ветку исходного репозитория.

**25. Какая команда позволяет отобразить историю репозитория?**

Git log

**26. Продемонстрируйте создание новых веток в Git? Что такое ветвление? Что такое указатель HEAD?**

Git checkout -b – создание ветки. Head – головной коммит.

**27. Как используется команда git checkout?**

Перемещение по ветвям

**28. Как выполнить включение изменений из одной ветки в другую?**

Git merge

**29. Какие проблемы могут быть при слиянии и как они разрешаются?**

**30. Что такое GitLab?**

**GitLab** — веб-приложение и система управления репозиториями программного кода для Git.