|  |  |
| --- | --- |
| Федеральное агентство железнодорожного транспорта  Омский государственный университет путей сообщения  Кафедра «Автоматика и системы управления»  ЗНАКОМСТВА С КОМАНДАМИ GIT  Лабораторная работа № 5  по дисциплине «Информационные технологии» | |
|  | Студент гр. 22м                            Н.И. Яковлев  «    »                  2023 г.  Руководитель –  старший преподаватель кафедры «АиСУ»                            Т. В. Васеева  «    »                  2023 г. |
| Омск 2023 | |

Цель работы

Целью данной лабораторной работы является изучение команд GIT, знакомства с ветками

# Выполнение работы

Следуя заданию лабораторной работы выполним пункты 1-7 (рисунок 1).

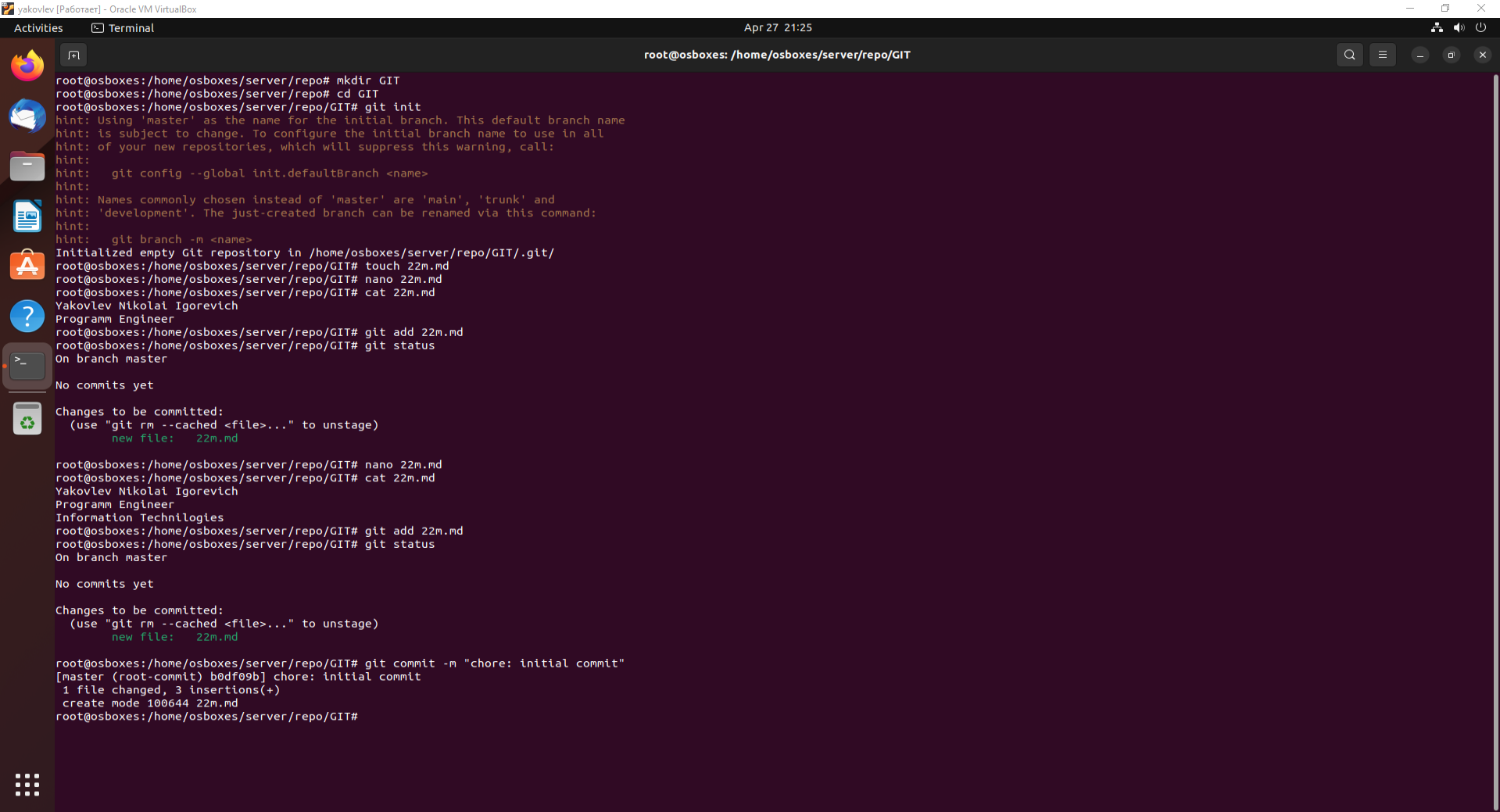


Рисунок  – Пункты 1-7

Затем выполним пункты 8-10 (рисунок 2): внесем некоторые изменения и попробуем восстановить исходный файл.

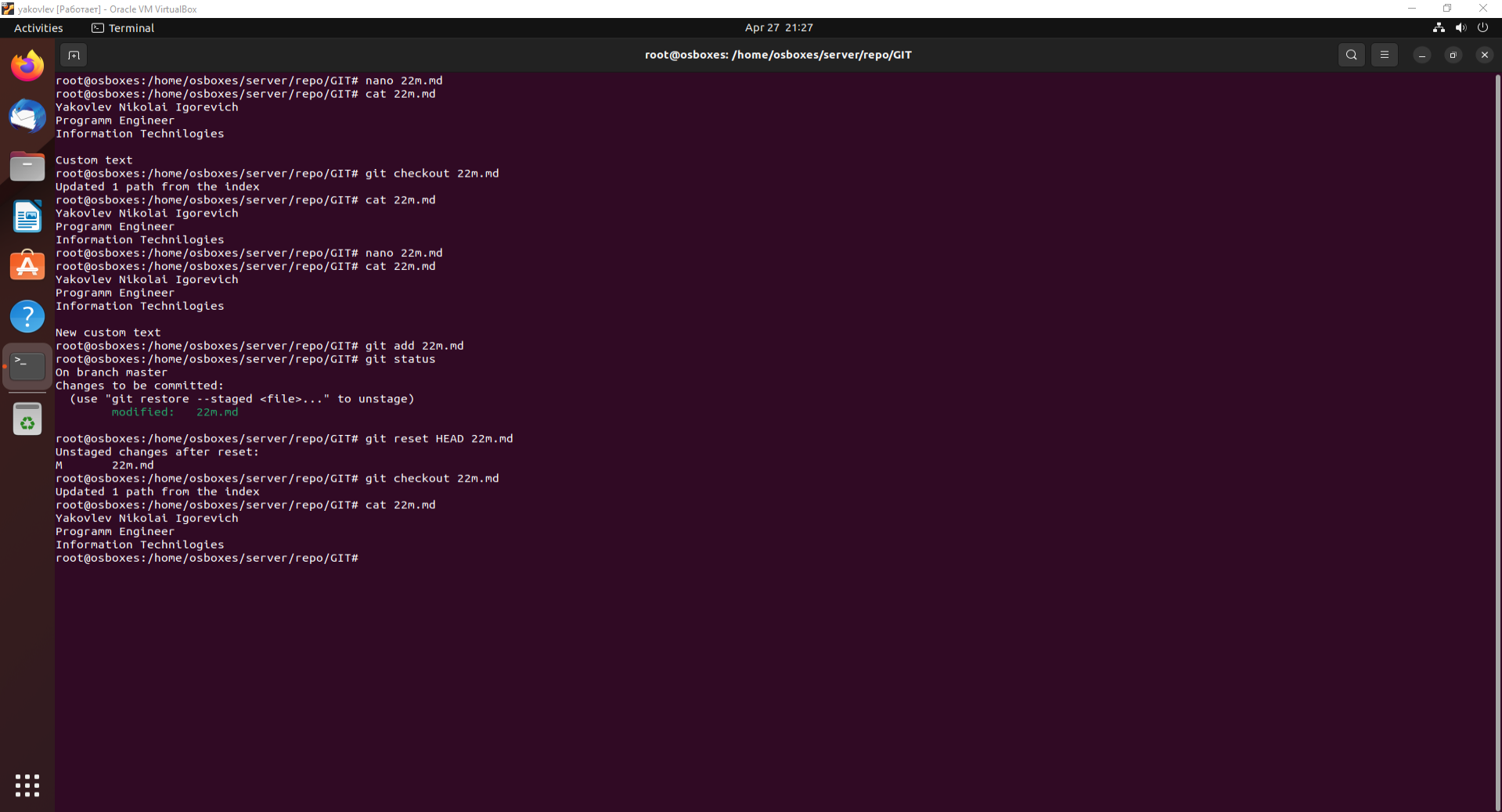


Рисунок  – Пункты 8-10

Выполним пункты 11-13 (рисунок 3): внесем некоторые изменение и закомитим их в прерыдующий коммит.

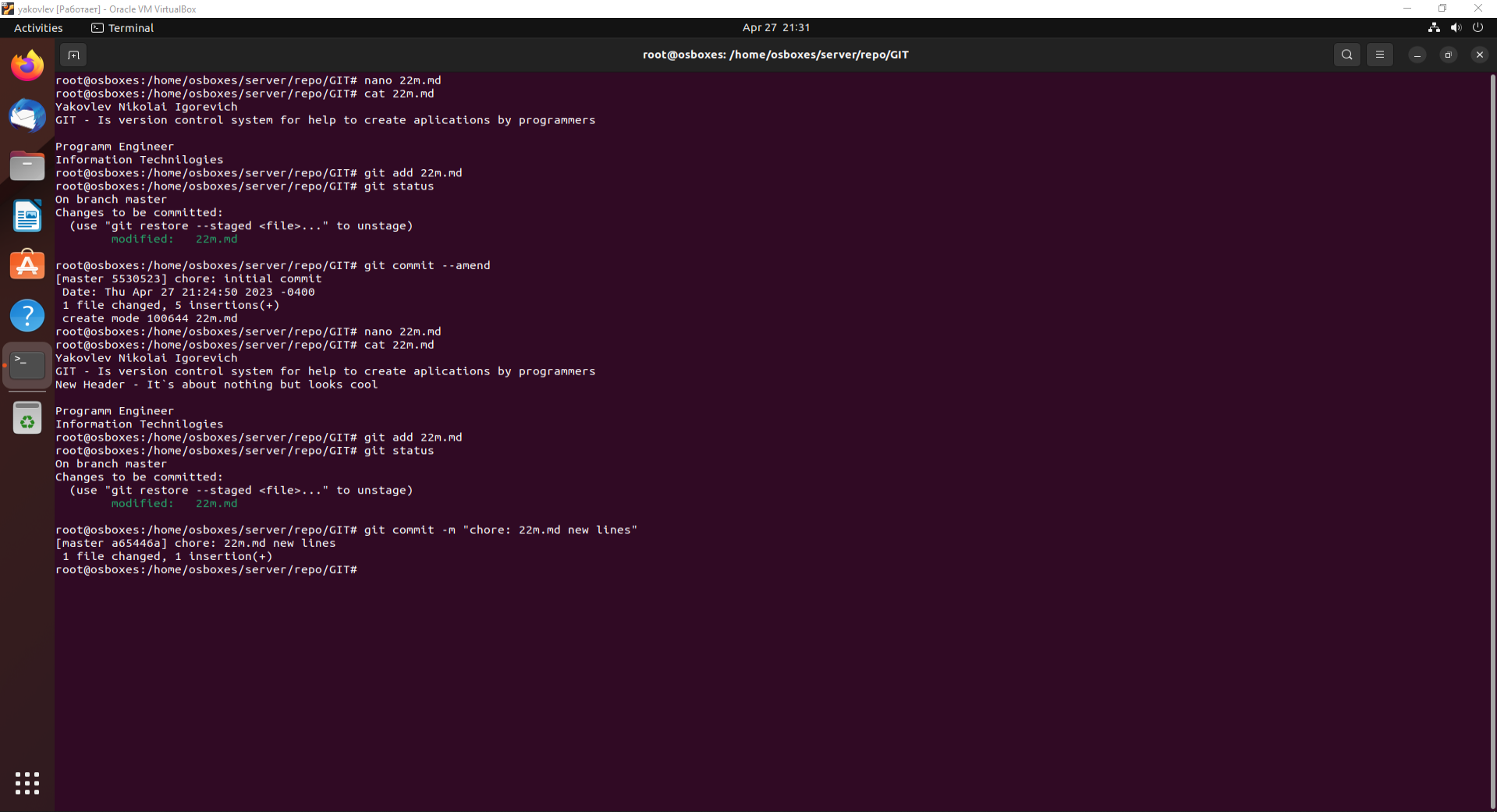


Рисунок  – Пункты 11-13

Выполним пункт 14 (рисунок 4): просмотрим историю коммитов с различными опциями.

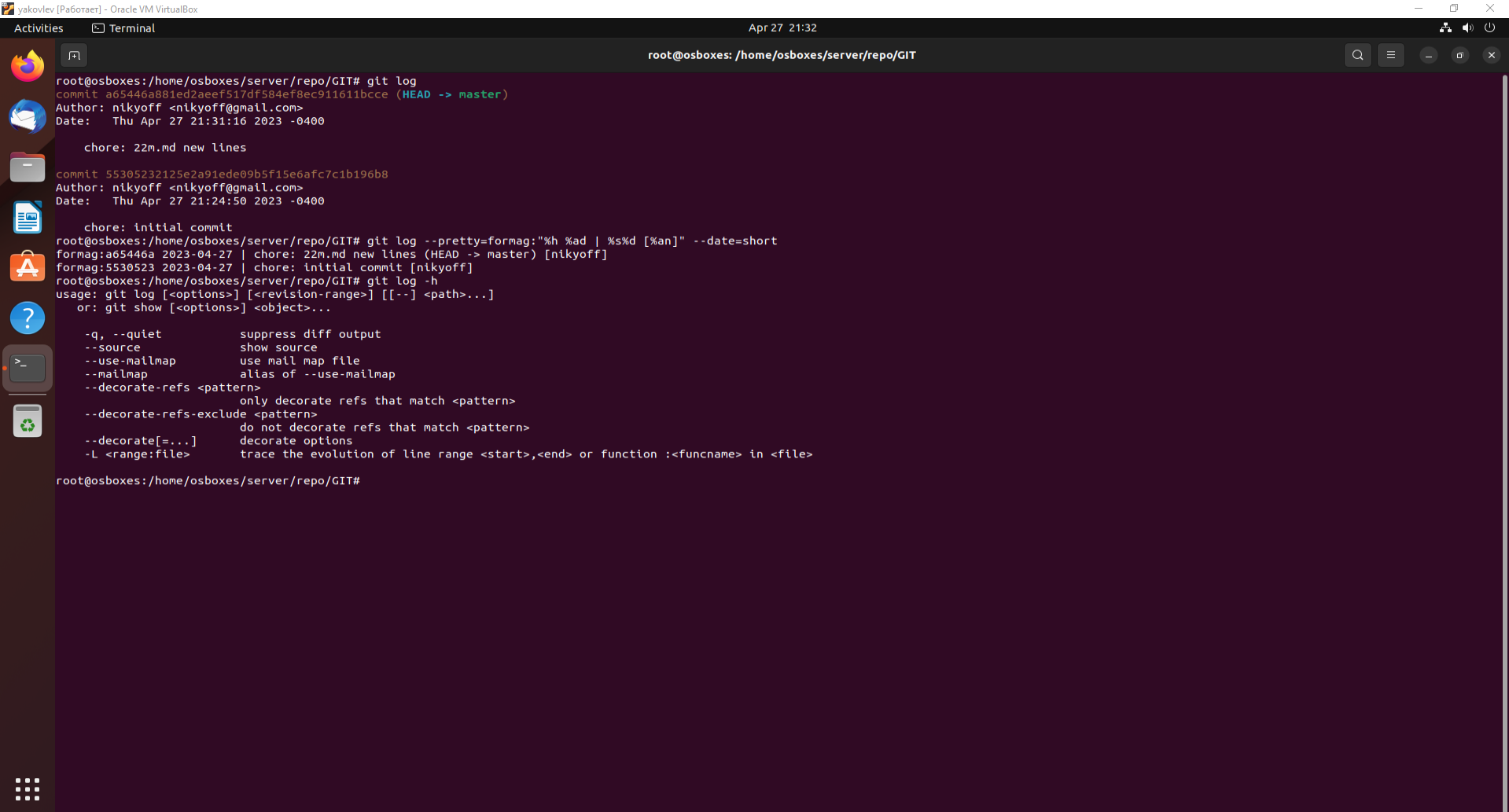


Рисунок  – Пункт 14

Выполним пункты 15-18 (рисунок 5): создадим новую ветку и внесем по комиту в каждую из веток.

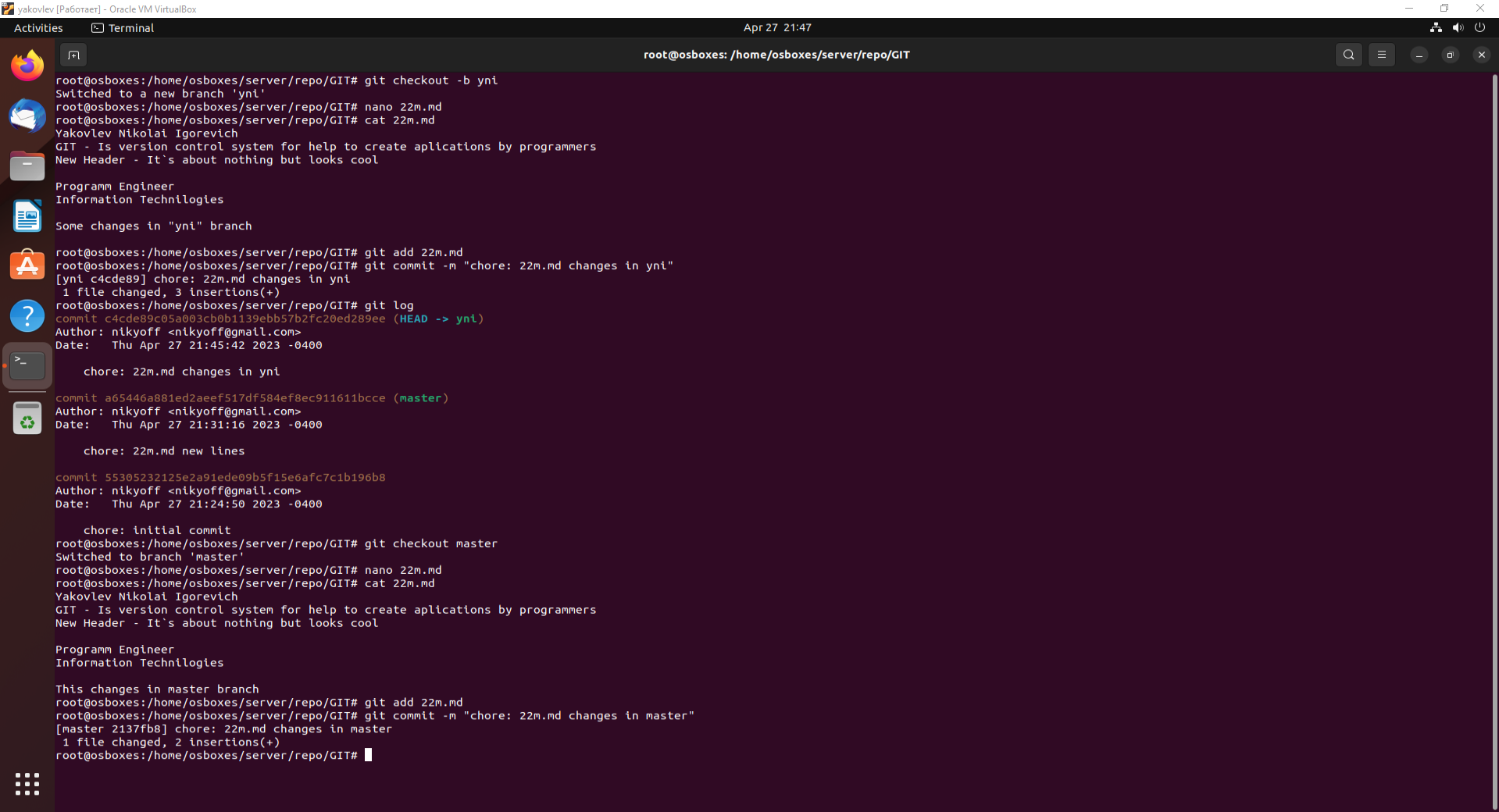


Рисунок  – Пункты 15-18

Выполним пункт 19 (рисунок 6): С помощью reset установим указатель на коммит с первыми изменениями файла на ветке yni.

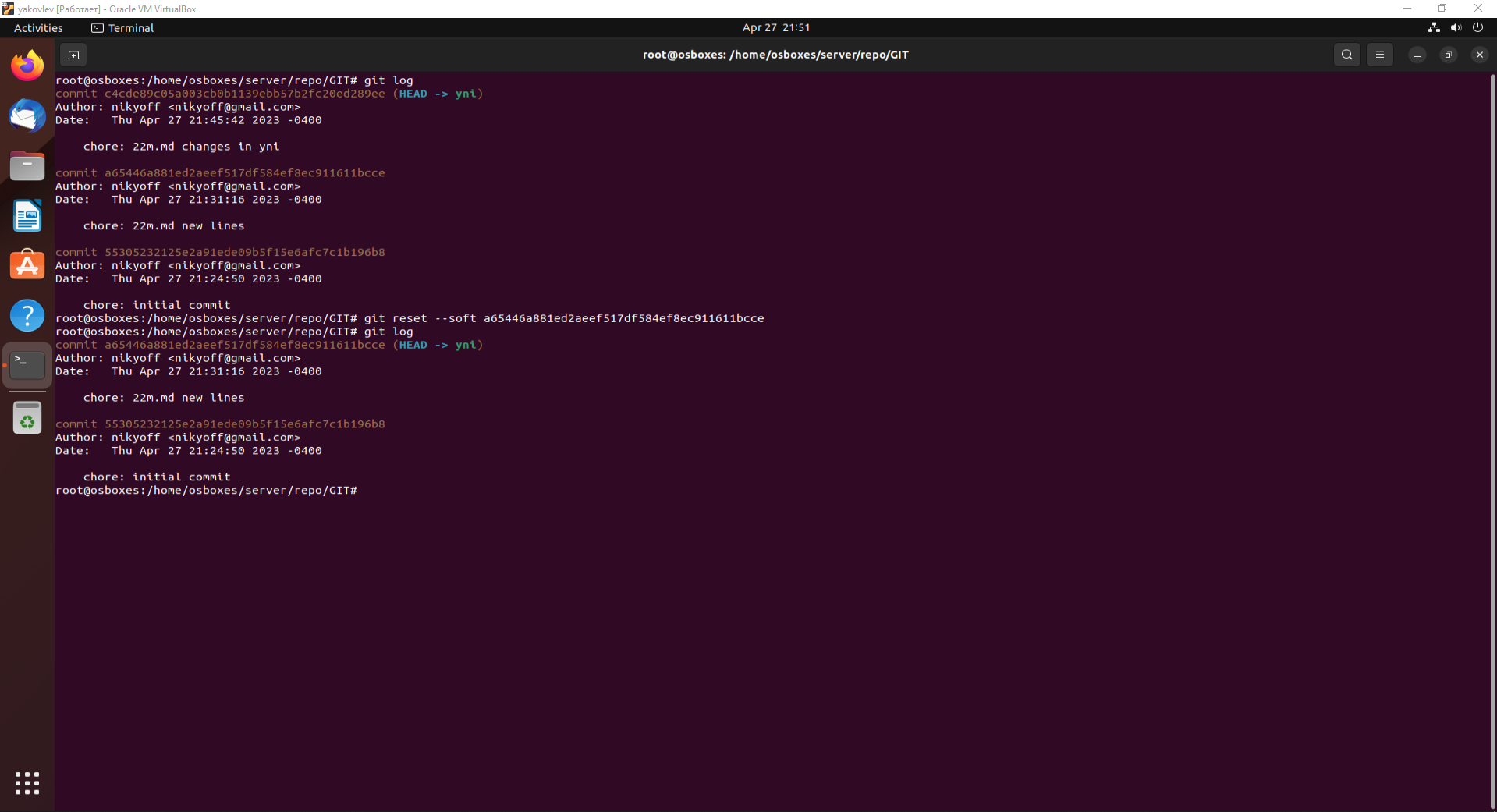


Рисунок  – Пункт 19

Выполним пункт 20 (рисунок 7): Установим указатель ветки master на коммит с первм изменением файла.

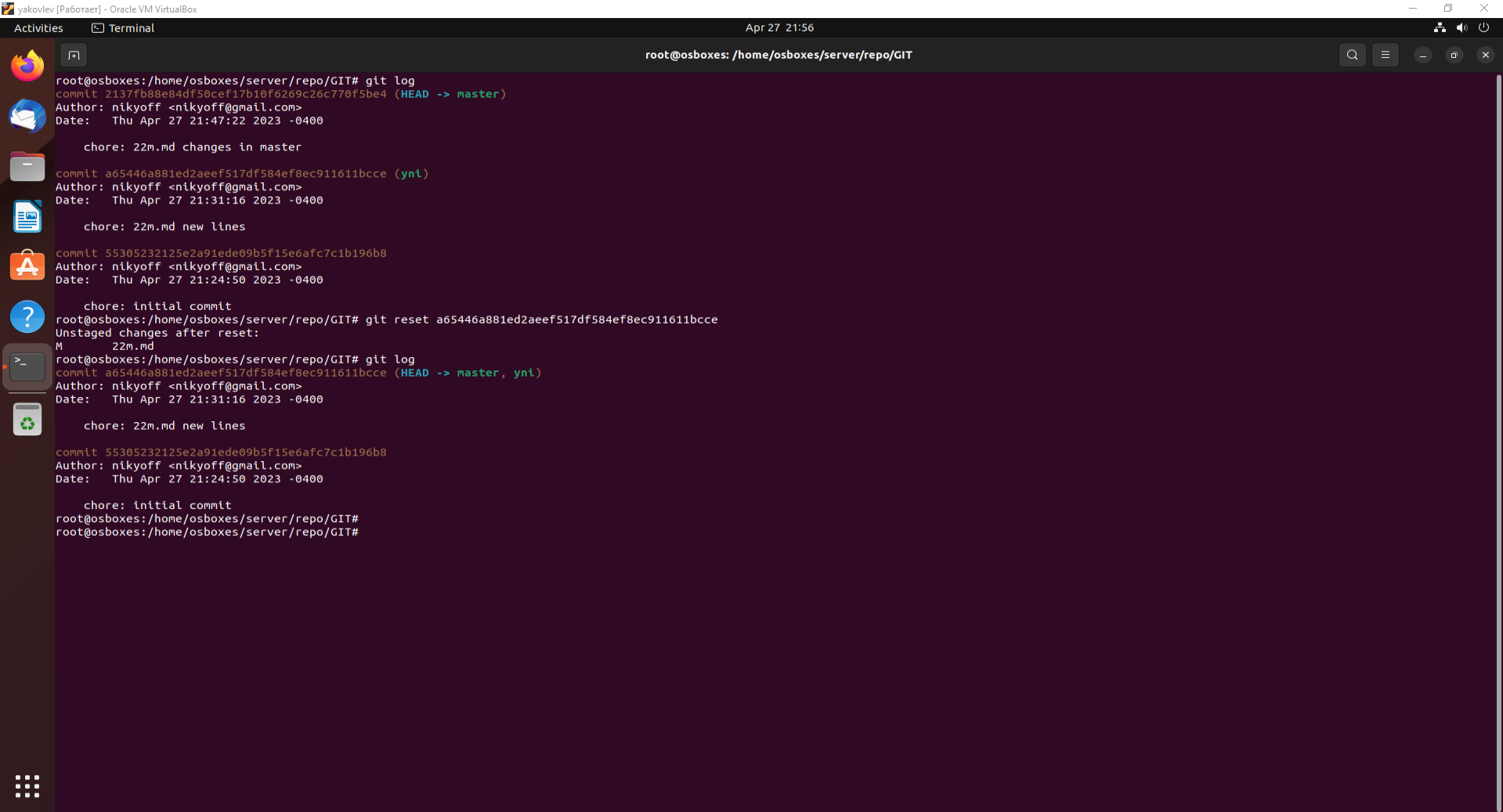


Рисунок  – Пункт 20

Выполним пункт 21 (рисунок 8): На новой ветке переместим указатель на первоначальный коммит в этой работе.

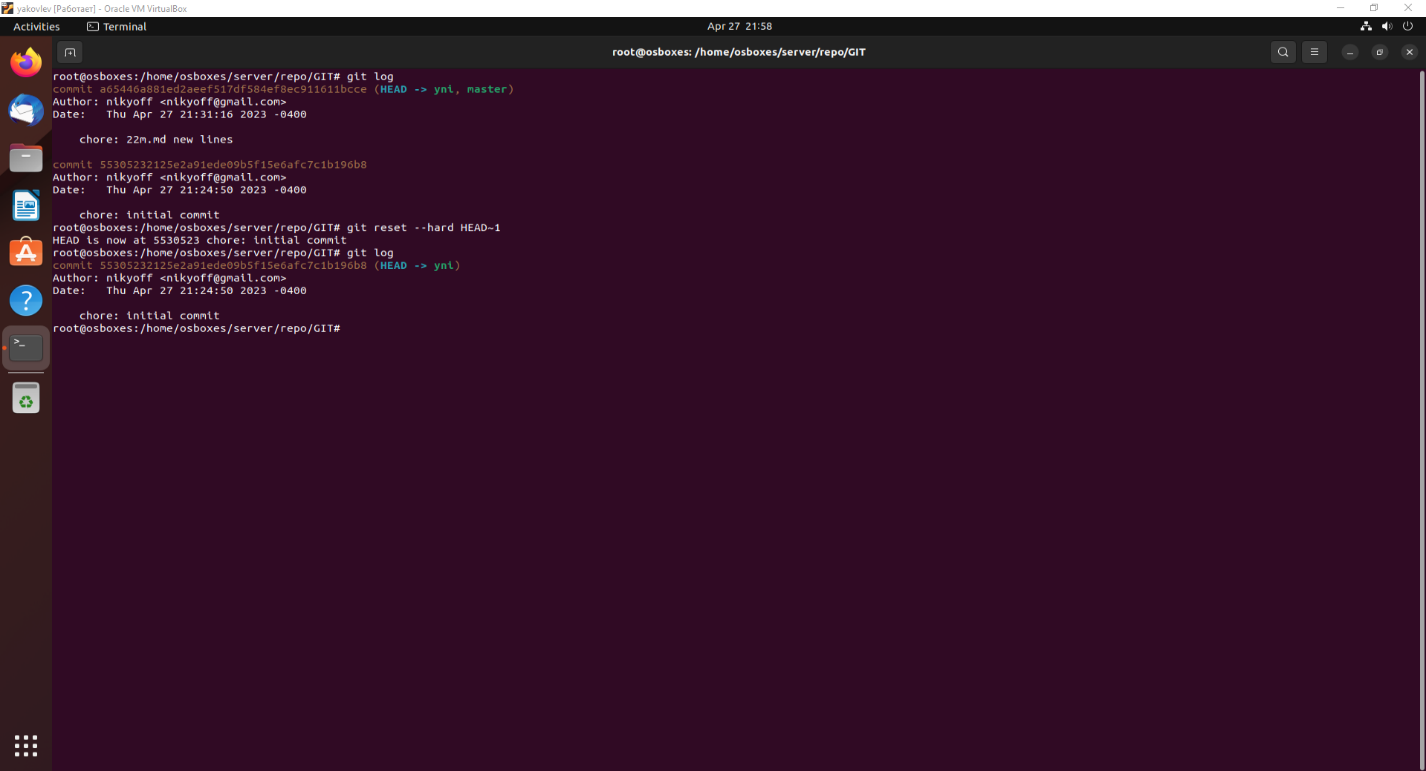


Рисунок  – Пункт 21

Таблица 1 – Команды

|  |  |  |
| --- | --- | --- |
| Команда | Значение | Расшифровка ключей |
| git status | Стаус репозитория |  |
| git checkout {branch|file} | Переключает ветки или восстанавливает файлы | “-b”: создает и переключается на новую ветку |

Продолжение таблицы 1

|  |  |  |
| --- | --- | --- |
| git reset {commit} | Изменяет состояние указателя HEAD | “--soft”: Обновляются только указатели “--mixed”: Указатели ссылок обновлдяются, раздел проиндексированных файлов сбрасывается до состояния указанного коммита, любые отменныеные изменения перемещаются в рабочий католог  “--hard”: указатели в истории коммитов обновляются на указанный коммит, происходит сброс раздела проиндексированных файлов и рабочего католога до указанного коммита |
| git commit | Фиксирует изменения | “--amend": добавить изменения в предыдущий коммит “--no-edit”: не изменять сообщение коммита |
| git log | Выводит информацию о коммитах | “--pretty”: устанавливает формат вывода |

# Контрольные вопросы

## Что такое: --soft, --mixed, --hard? Примеры в терминале

Три дерева Git: рабочий каталог, раздел проиндексированных файлов и история коммитов. Эти три ключа отличаются работой с этими деревьями.

Hard обновляет все три дерева, иными словами полностью откатывает до указанного коммита (рисунок 9). Важно сказать, что возможности восстановить изменения нет.

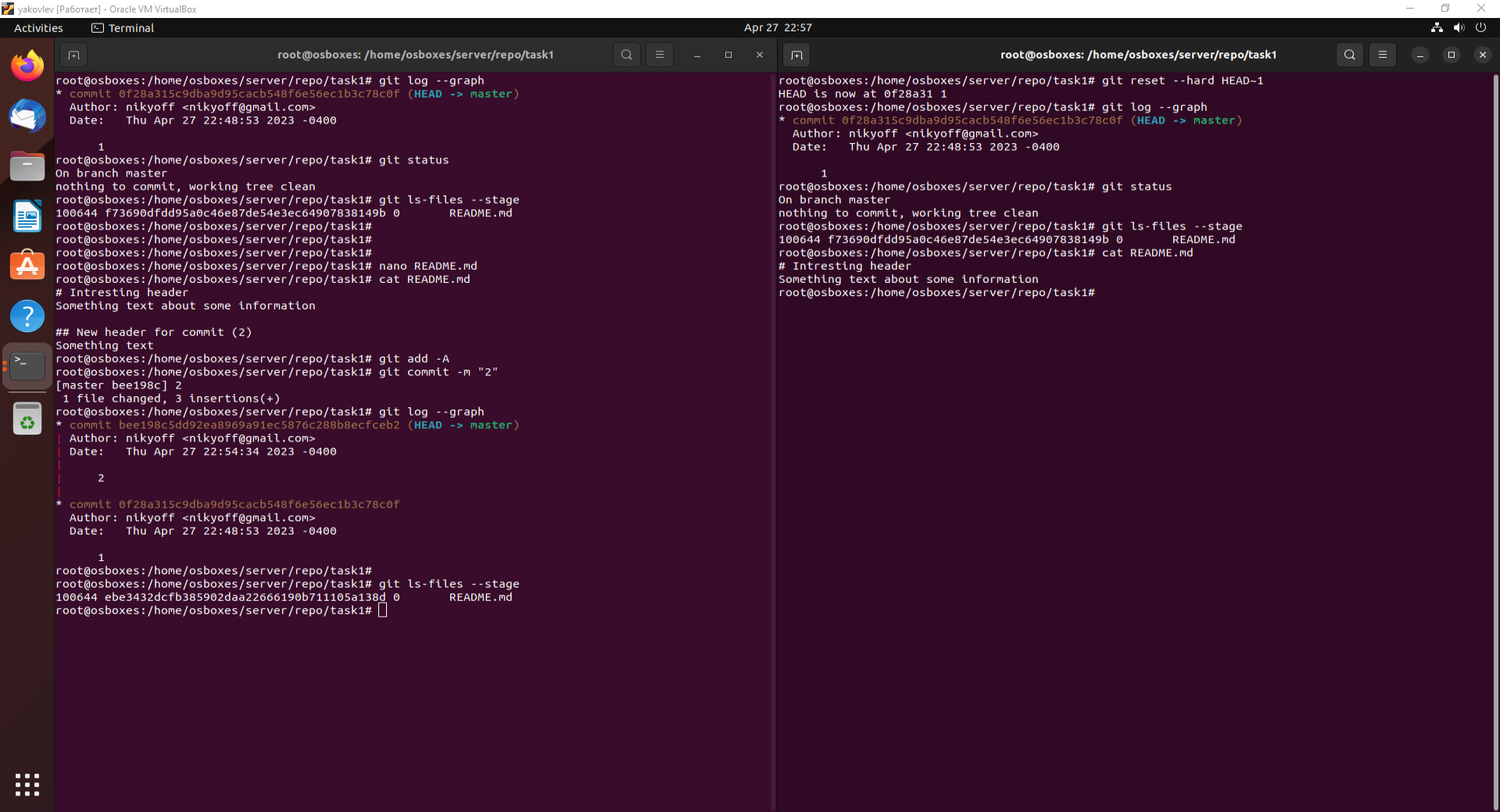


Рисунок  – Ключ hard

Mixed делает тоже самое, за одним исключением: все изменения попадают в рабочий католог. Иными словами сбрасывается индексация, откатывается коммит, но при этом все изменения остаются как непроиндексированные изменения (рисунок 10).

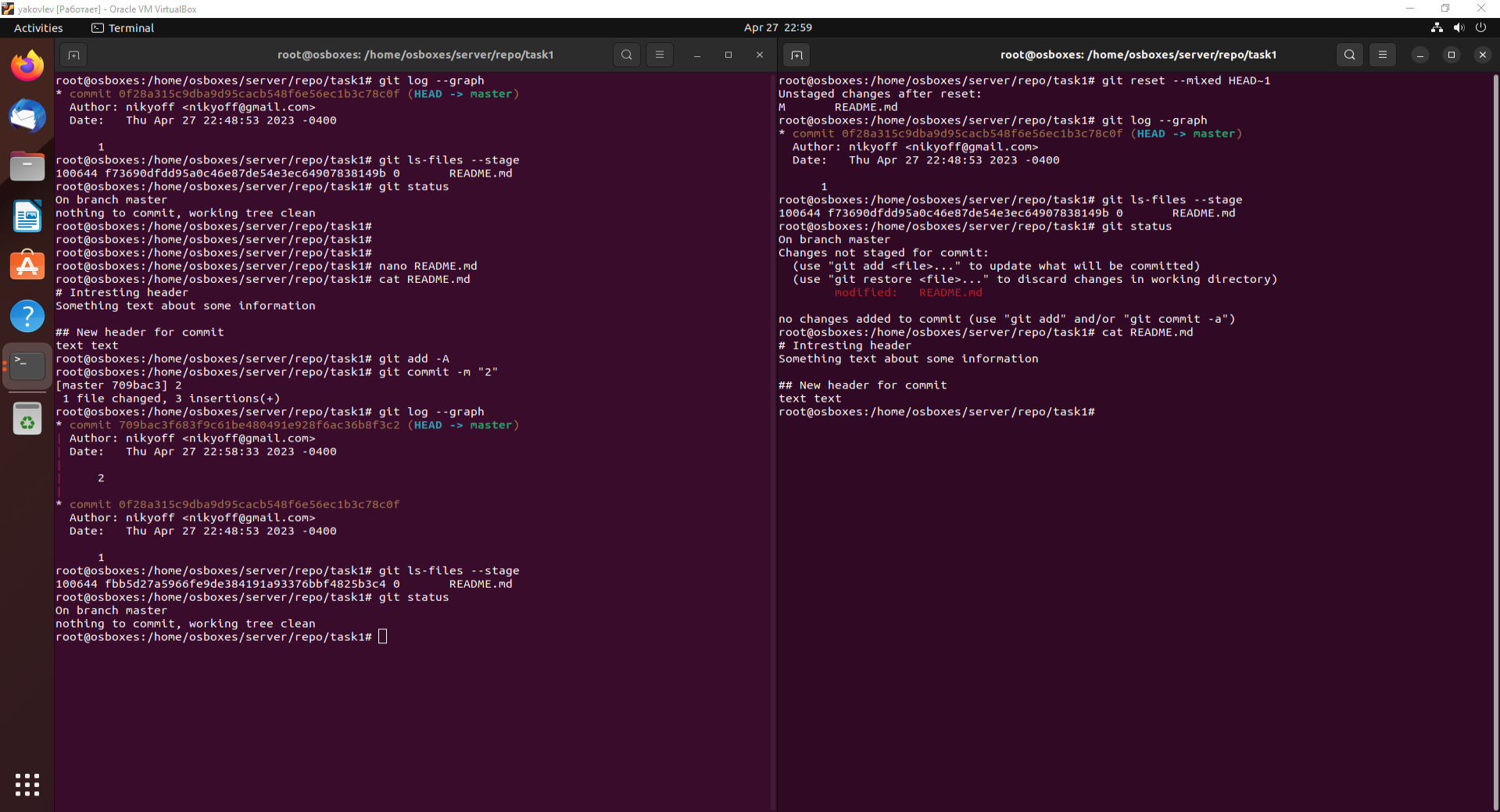


Рисунок  – Ключ mixed

В свою очередь soft изменяет лишь указатель ветки на указанный коммит, т.е. в конечном итоге мы не ожидаем увидеть каких-либо изменений, за исключением перестановки указателя. При этом важно сказать, если мы сдалем целевой коммит, а после сделаем git reset --soft, то коммит будет отменен и его нужно будет повторно закоммитить (рисунок 11).

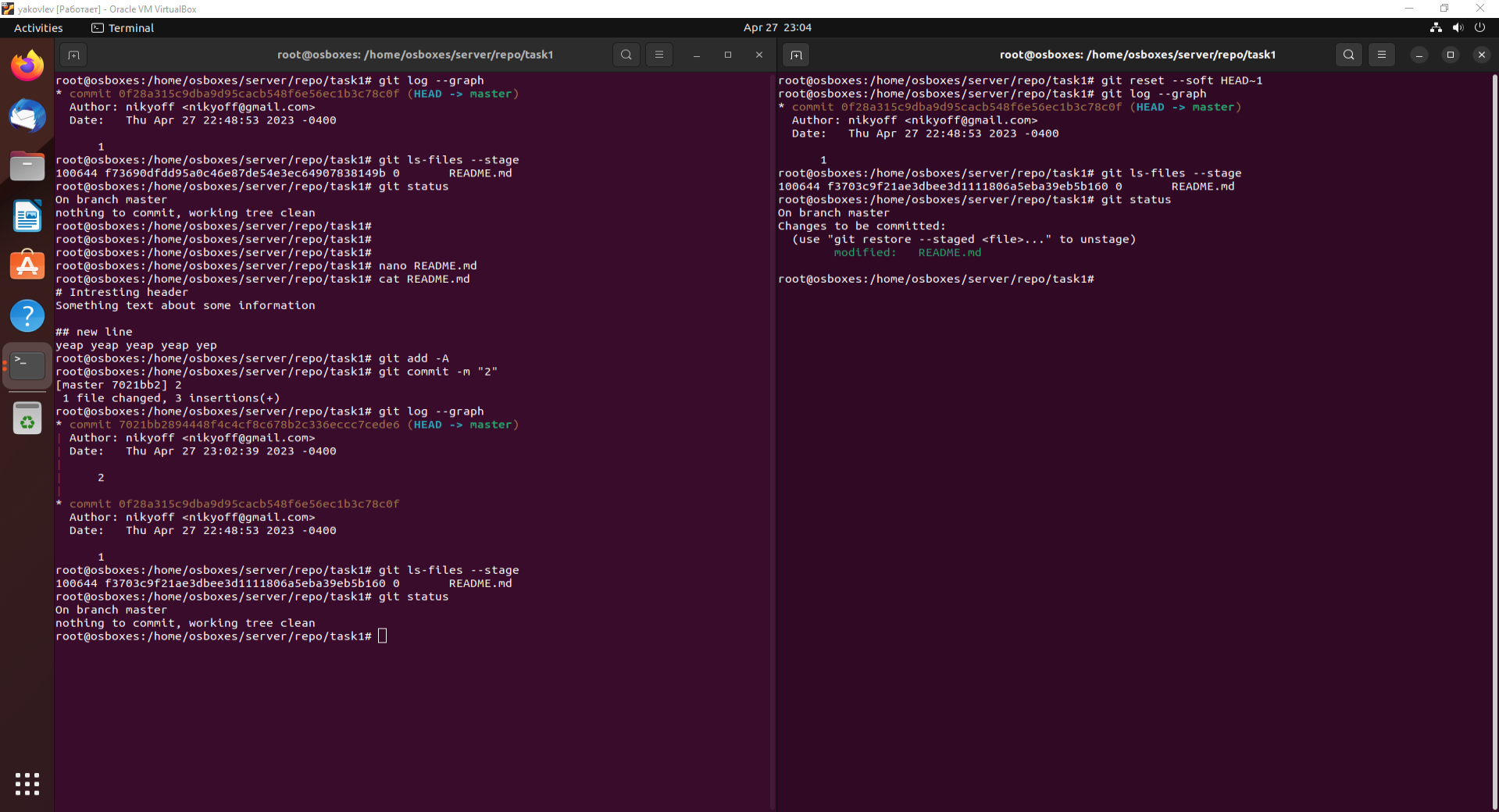


Рисунок  – Ключ soft

## Просмотр истории коммитов и веток (5 вариантов)

Пример просмотра истории коммитов и веток (рисунок 12).

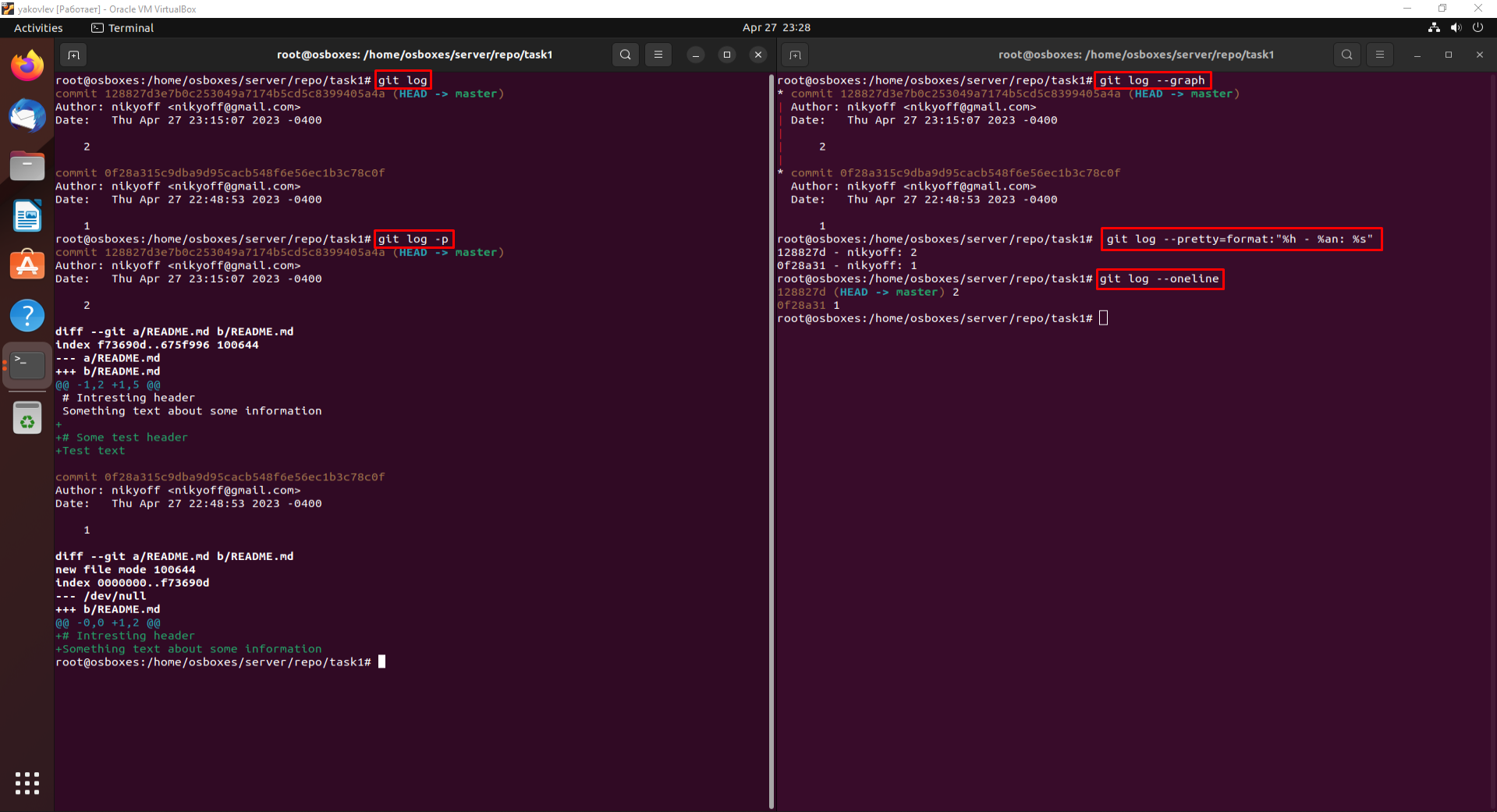


Рисунок  – Просмотр истории коммитов и веток

## Выполнить работу с ветками

Выполним задание (рисунок 13).

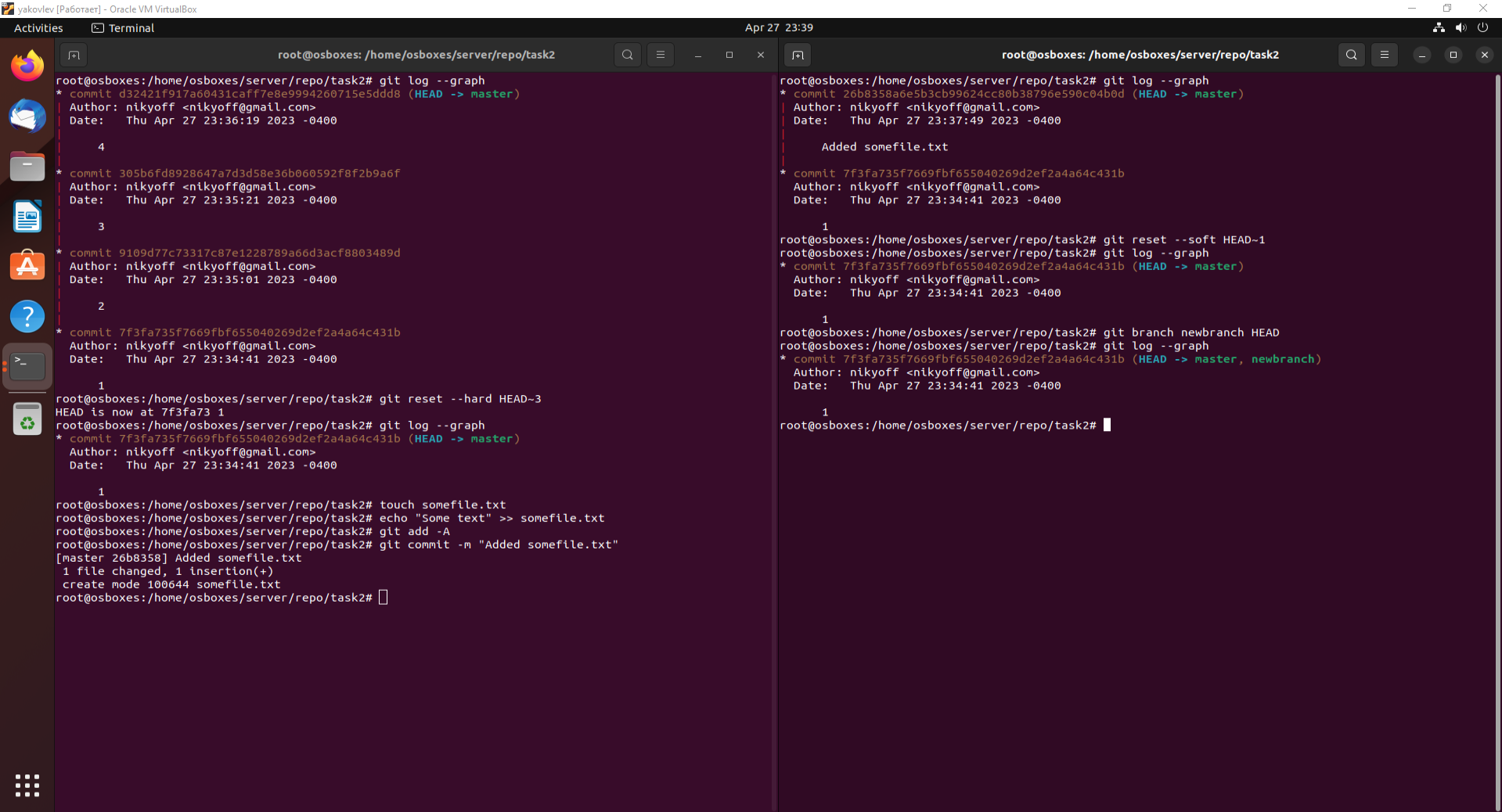


Рисунок  – Выполненное задание

Вывод

В данной лабораторной работе, мы познакомились с командами GIT, научились работать с индексацией, фиксацией и комитами, а также посмотрели на базовые возможности при работе с ветвлениями.