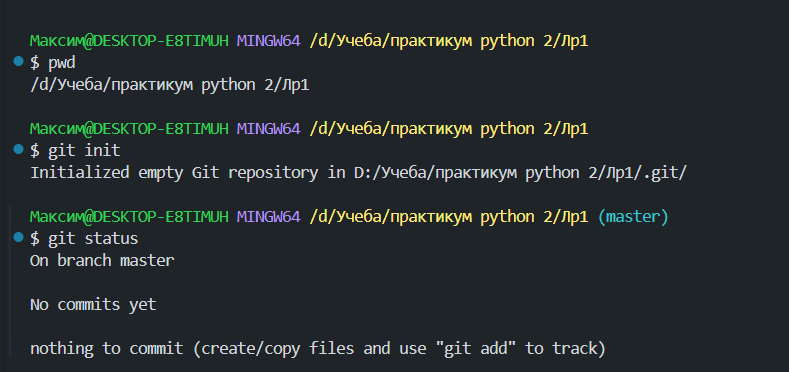
ЛР 1 по практике программирования. Маев Максим ИД23-2

Приёмы работы в современных системах контроля версий.

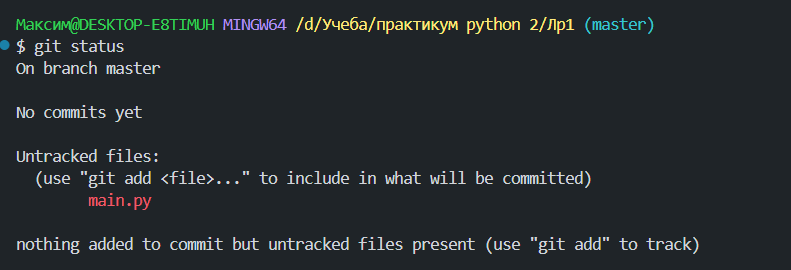
**Цель работы:** познакомиться на практике с основными приемами работы в современных системах контроля версий.

**Задачи для выполнения:**

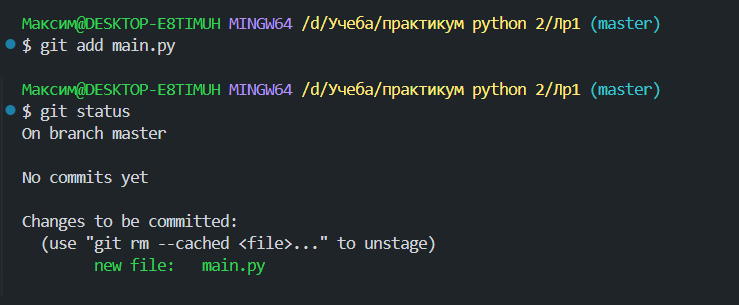
1. Работа с системой контроля версий Git.
2. Работа GUI git клиент GitKraken.
3. Работа с удаленными репозиториями и GitHub.
   1. Установить на компьютер графический клиент Git.
   2. Создайте в своей домашней папке (или в любой другой на ваш выбор) каталог, который будет содержать файлы нового программного проекта.
   3. Выберите тематику программы, которую собираетесь написать.
   4. Инициализируйте в этой директории репозиторий гит при помощи команды git init. Обратите внимание на появление в этой папке скрытой подпапки с названием .git. Если вы ее не видите, то скорее всего, у вас отключено отображение скрытых папок.
   5. Выполните в репозитории команду git status. Проинтерпретируйте полученное сообщение.



* 1. Создайте файл для исходного текста программы. Выполните команду git status.

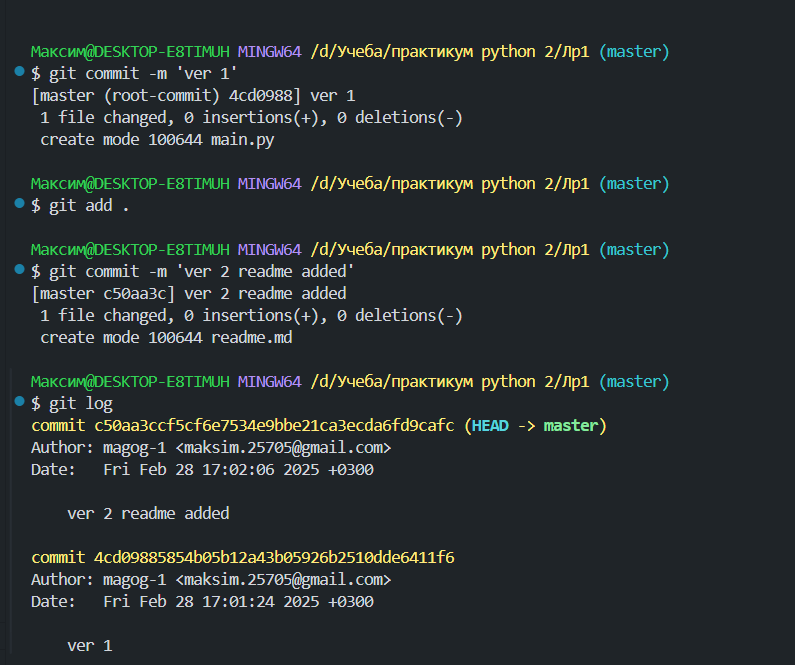


* 1. Добавьте созданный файл под версионный контроль при помощи команды git add. Еще раз выполните git status.

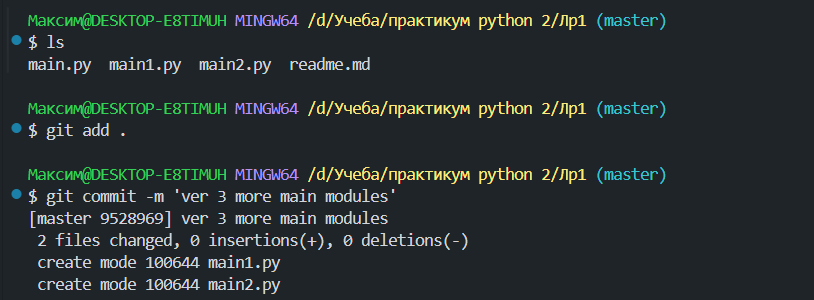


* 1. Когда все файлы готовы к сохранению, сделаем наш первый коммит — зафиксируем все сделанные изменения в «боевой версии». Делается это командой git commit c опцией -m. После -m идёт название коммита в кавычках.
  2. Сделайте еще несколько коммитов. Выполните команду git log для

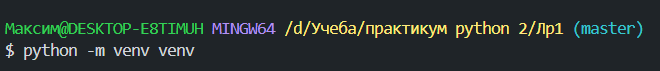
просмотра истории коммитов.



* 1. Сделайте так, чтобы при коммите измененные файлы автоматически добавлялись в коммит.
  2. Добавьте еще несколько файлов с исходным текстом программы.
  3. Добавьте все новые файлы под версионный контроль одной командой.



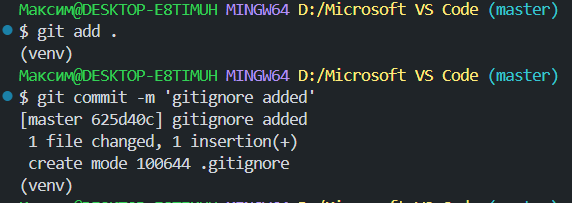
* 1. Инициализируйте в рабочей директории виртуальное окружение



* 1. Добавьте созданную служебную папку в файл .gitignore. Проверьте,

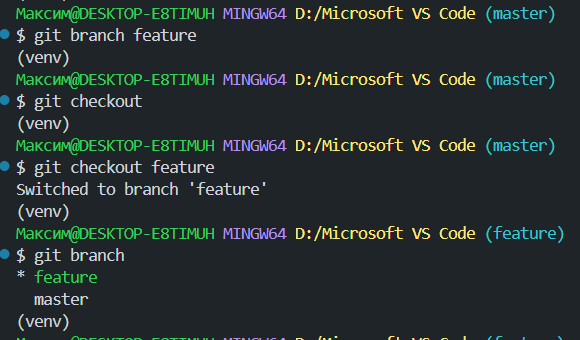
что они не добавляются в репозитории при добавлении новых файлов с исходным кодом.





* 1. Создайте новую тематическую ветку git branch. Перейдите в нее с

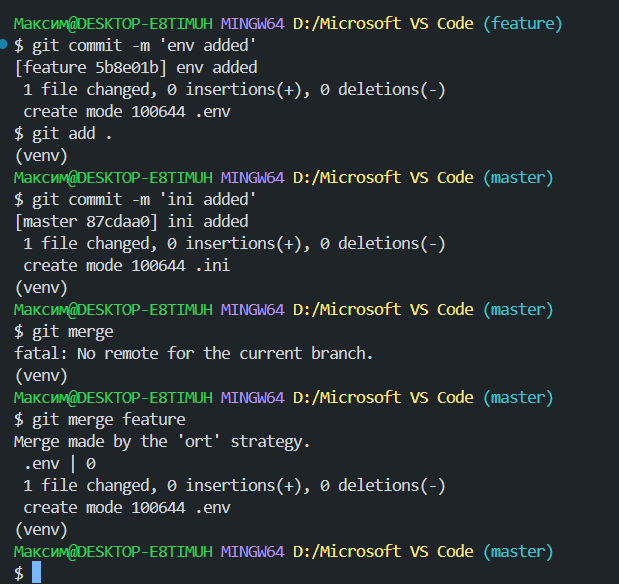
помощью git checkout. Выведите на экран список всех веток.



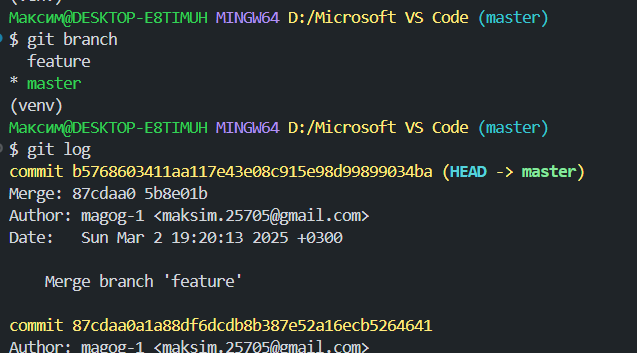
* 1. Сделайте несколько коммитов в основную и тематическую ветки
  2. Слейте изменения в основную ветку с помощью git merge. Если

произошел конфликт слияния, разрешите его и завершите слияние с

помощью git commit.

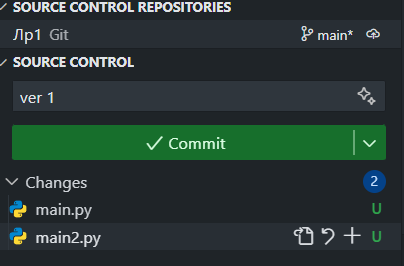


* 1. При получении в процессе разработки программы в стабильно работающем состоянии, слейте это состояние в основную ветку и добавьте к коммиту слияния пометку с номером релиза.

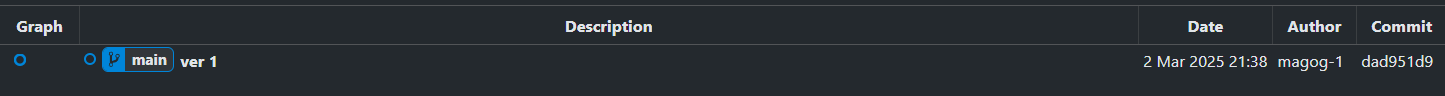


**Упражнение 2.** Работа GUI git клиент GitKraken.

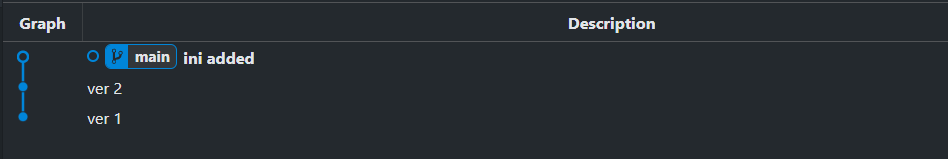
1. Выберите тематику программы, которую собираетесь написать. Создайте для нее рабочую директорию
2. Инициализируйте в рабочей директории репозиторий.
3. Создайте файл для исходного текста программы.
4. Добавьте созданный файл под версионный контроль.



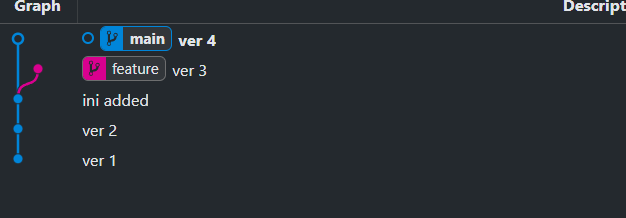
1. Сделайте начальный коммит.



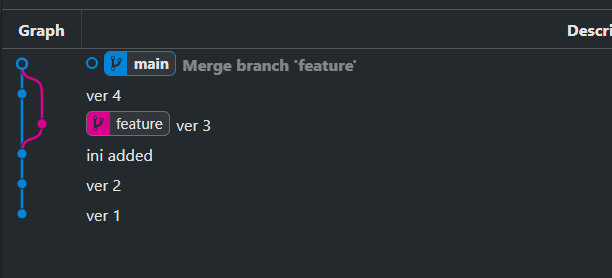
1. Сделайте еще несколько коммитов.
2. Добавьте еще несколько файлов с исходным текстом программы.



1. Создайте новую тематическую ветку.
2. Сделайте несколько коммитов в основную и тематическую ветки.

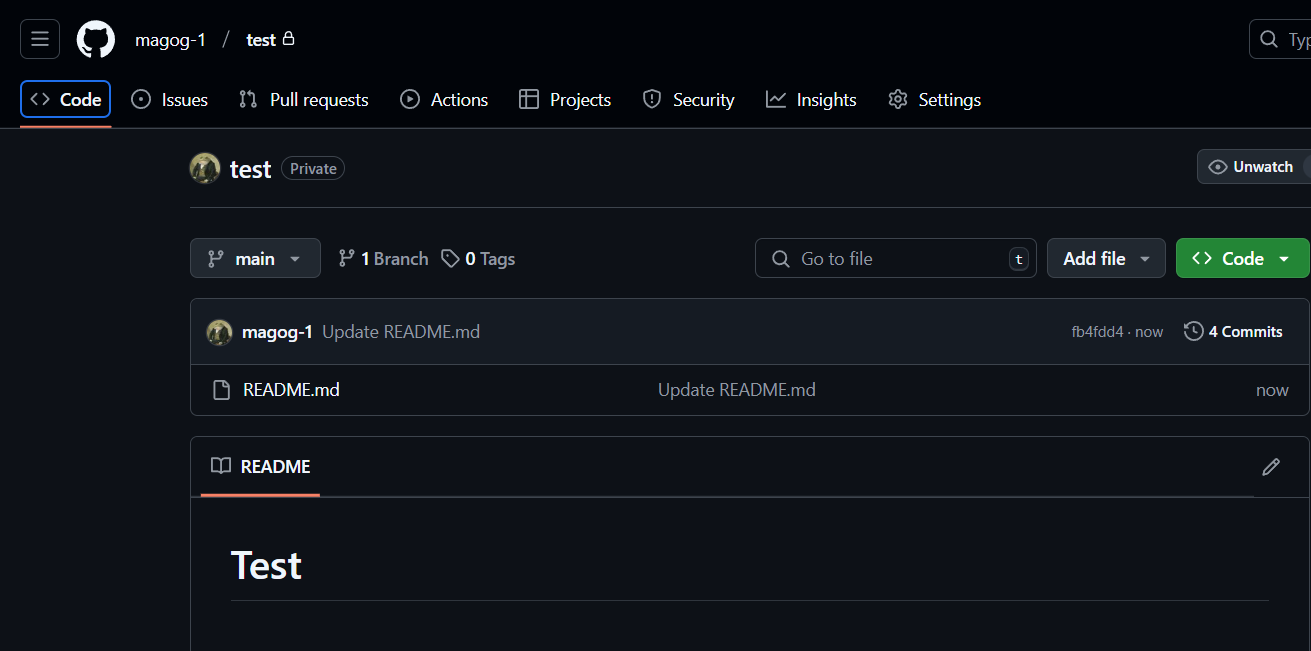


1. Слейте изменения в основную ветку. Если произошел конфликт слияния, разрешите его и завершите слияние.
2. При получении в процессе разработки программы в стабильно работающем состоянии, слейте это состояние в основную ветку и добавьте к коммиту слияния пометку с номером релиза.

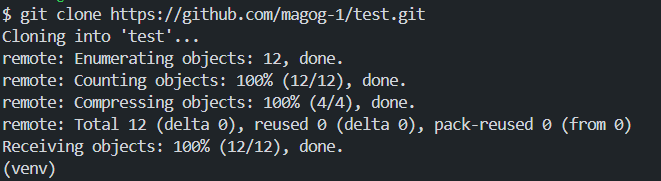


**Упражнение 3.** Работа с удаленными репозиториями и GitHub.

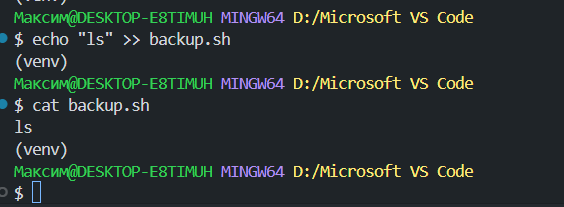
1. Зарегистрироваться на сайте github.com и создать новый пустой репозиторий
2. Создать в этом репозитории файл Readme



1. Если вы работаете с git:
2. – Склонировать созданный удаленный репозиторий в директорию ~/git/test



1. – На локальной машине пишем скрипт ~/git/test/backup.sh, с произвольным содержанием



1. – Фиксируем скрипт в репозитории (делаем коммит)
2. – Обновляем удаленный репозиторий репозиторий (делаем пуш)

