Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

Лабораторная работа 9

По дисциплине «Основы программной инженерии»

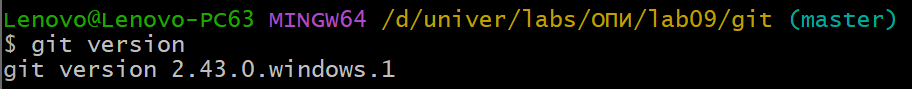
На тему «Системы контроля версий. Основы работы в Git»

Выполнил:

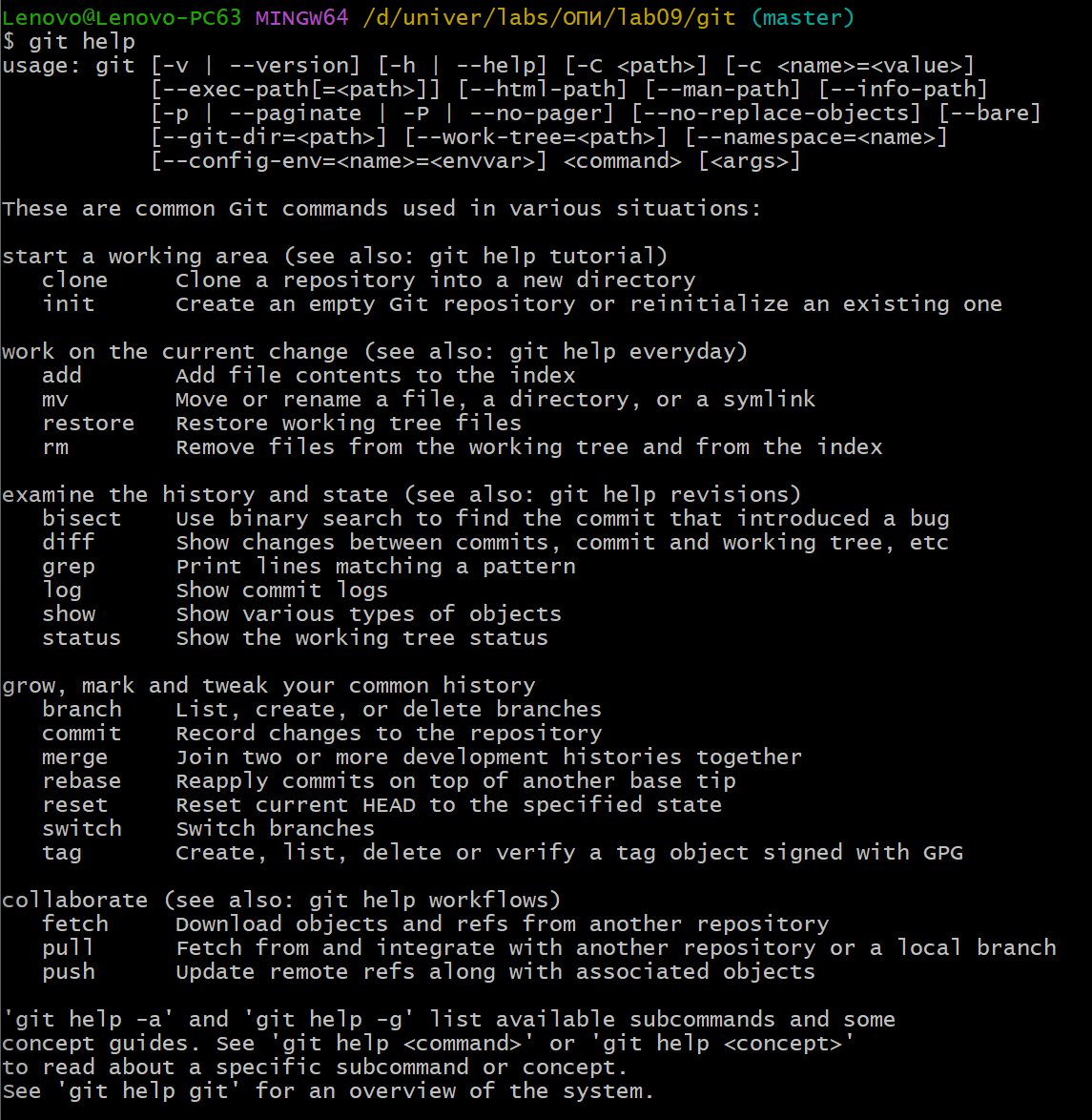
Студент 1 курса, 8 группы, 1 подгруппы

Кулешов Артём Алексеевич

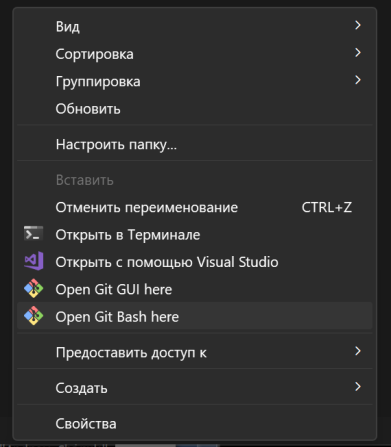
4. Установленная версия Git:



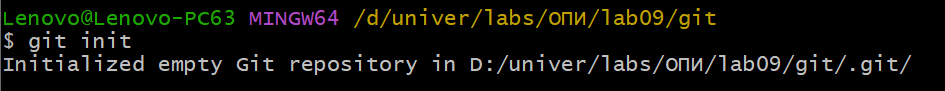
5.Для получения справки по Git можно использовать команду: git help.



6. Для создания локального репозитория необходимо перейти в папку, где планируется создать репозиторий и запустить Git Bash с помощью контекстного меню: «Git Bash Here».



Использовать команду: git init.



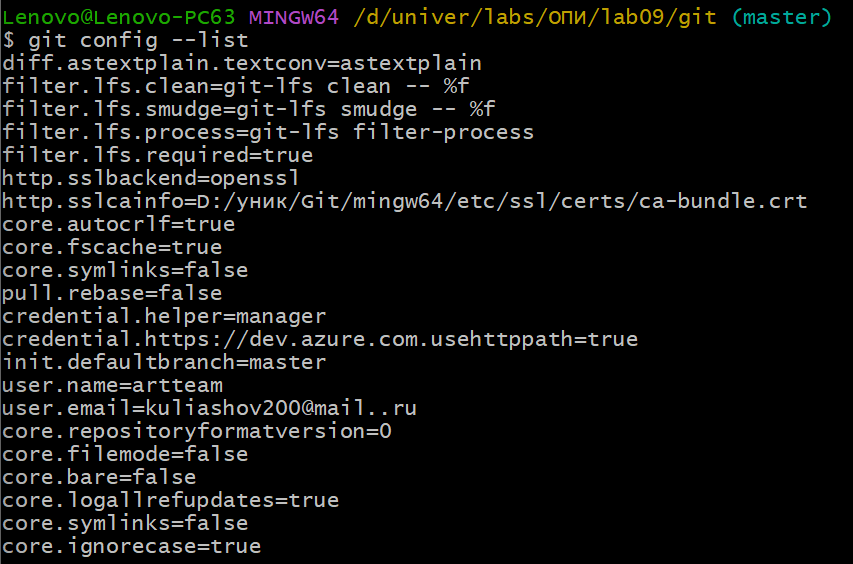
7. Выполнить конфигурацию Git. Для этого необходимо настроить конфигурационный файл, используя: git config --global user.name “artteam”.



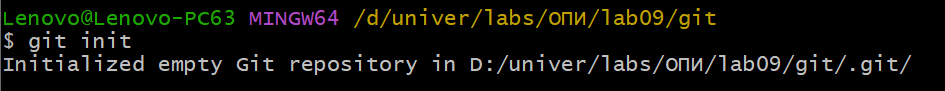
git config --global user.email “kuliashov200@mail.ru”.



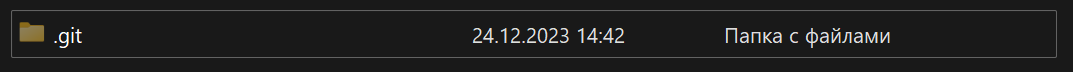
8. Проверить настройки конфигурационного файла командой: git config –list.



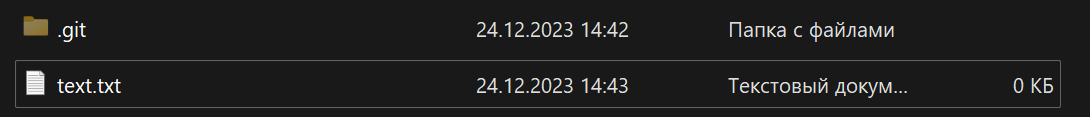
9. Для инициализации локального репозитория в рабочей папке необходимо использовать команду: git init.



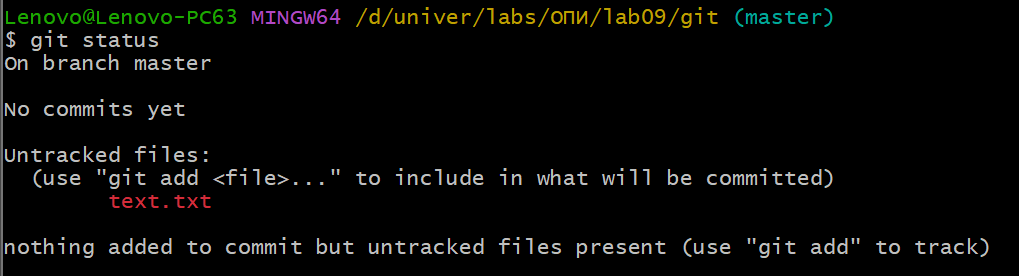
После ввода команды git init, в рабочей папке создаётся скрытый файл: .git



10. В рабочей папке создаём пустой элемент: текстовый файл text.txt.



11.Для получения информации о статусе файла необходимо использовать команду: git status.



После ввода отобразился результат в котором:

On branch master – означает, чтомы получили статус ветки master.

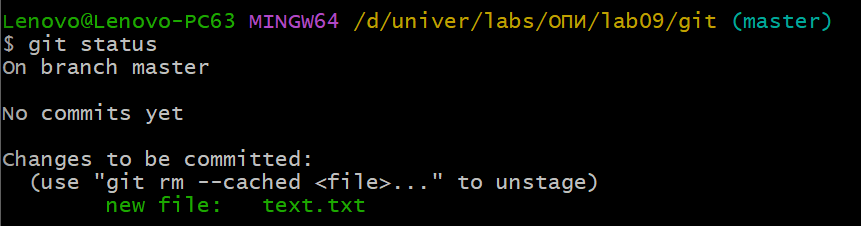
Запись “No commits yet” означает отсутствие коммитов.

Красным цветом отмечаются новые и модифицированные файлы и папки.

12. Для индексикации файла используем команду: git add text.txt.

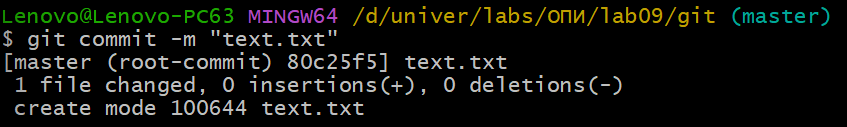


Ещё раз используем команду: git status.



Зелёным цветом отображается индексированный файл.

13. Для фиксации файла вопользуемся командой: git commit -m “text.txt”.

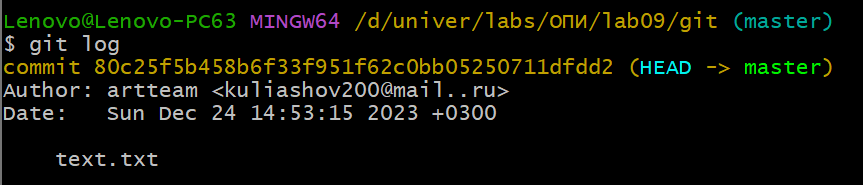


После выполнения команды git информирует об успешном создании нового коммита.

Файл был добавлен в ветку master.

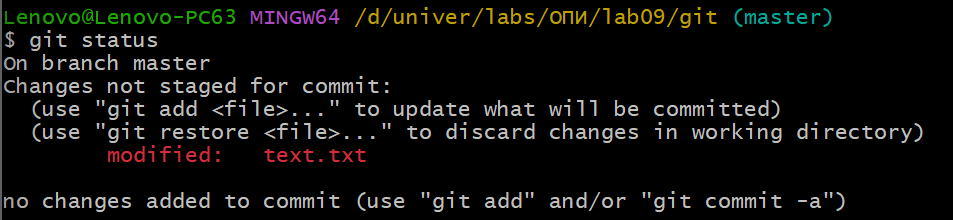
В репозитории в папке .git, изменились: папка logs, папка objects, файл COMMIT\_EDITMSG, файл index.

14.Просмотрели журнал с помощью команды: git log.



git информирует об зафиксированном коммите и его названии.

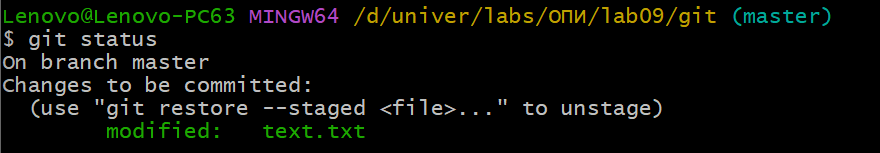
15. В текстовый файл были добавлены изменения, в следствии чего появляется следующее:



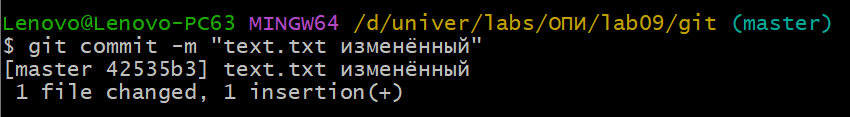
Далее необходимо проиндексировать данный файл.



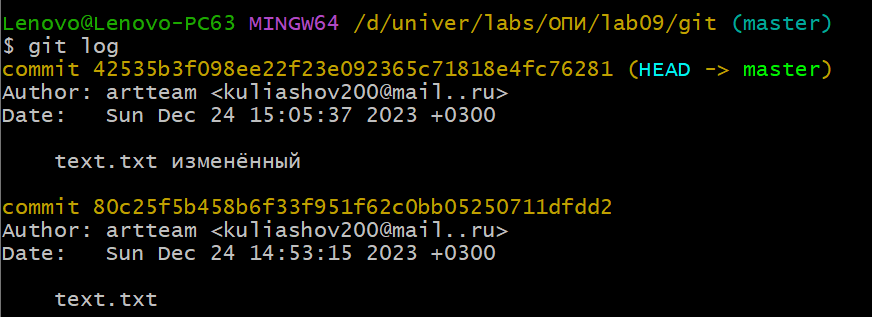
После необходимо воспользоваться командой: git status.



Далее необходимо зафиксировать файл, совершив несколько изменений в его названии для простоты ориентирования между ними.



Вызываем журнал изменений командой: git log. После чего видим изменения файлов и их успешную фиксацию.



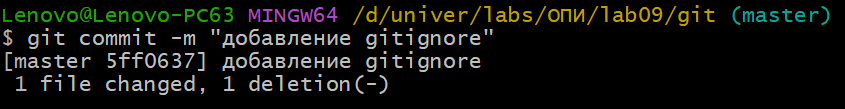
17. Чтобы настроить файл .gitignore, необходимо создать пустой файл в локальном репозитории.



Далее этот модифицированный элемент необходимо проиндексировать.



После зафиксировать.



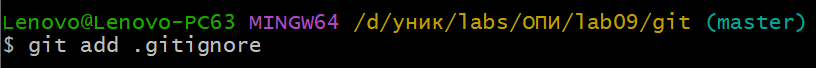
Далее необходимо решить какие файлы должны быть проигнорированы при отслеживании изменений.

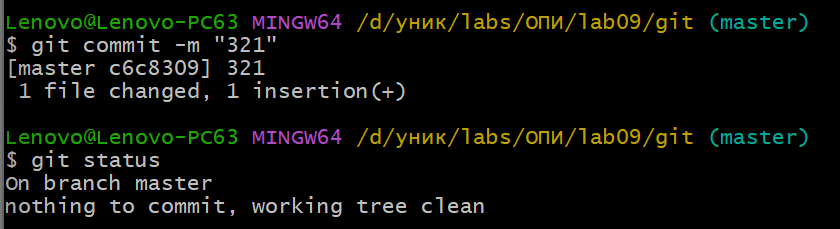
Создаём в репозитории элемент textnone.txt, который должен быть проигнорирован.

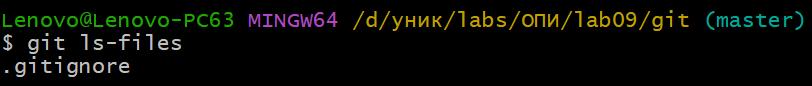


Добавляем исключение в .gitignore. 

После индексируем элемент.



Далее фиксируем элемент. 



Вопросы

1. Что такое система контроля версий, для чего ее используют?

Система контроля версий - программное обеспечение для облегчения

работы с изменяющейся информацией и разработки проекта совместно с коллегами.

1. Какой принцип хранения файлов использует Git?

Git является распределенным, то есть не зависит от одного центрального сервера, на котором находятся файлы. Вместо этого он работает полностью локально, сохраняя данные на жестком диске, которые называются репозиторием.

**Git** — распределённая система контроля версий, позволяющая сохранять изменения, внесённые в файлы, которые хранятся в репозитории. Сами изменения сохраняются в виде снимков, называемых коммитами. Они могут размещаться на разных серверах, поэтому вы всегда восстановите код в случае сбоя, а также без проблем откатитесь до любого предыдущего состояния

1. В чем отличие Git от других систем контроля версий?

Отличие Git от других систем контроля версий заключается в его распределенной структуре. Каждый пользователь имеет локальную копию репозитория, что обеспечивает быструю работу и возможность работы без подключения к сети.

Основное отличие Git от любой другой системы контроля версий (включая Subversion и её собратьев) — это подход к работе со своими данными. Концептуально, большинство других систем хранят информацию в виде списка изменений в файлах. Эти системы (CVS, Subversion, Perforce, Bazaar и т. д.) представляют хранимую информацию в виде набора файлов и изменений, сделанных в каждом файле, по времени (обычно это называют контролем версий, основанным на различиях).

Git не хранит и не обрабатывает данные таким способом. Вместо этого, подход Git к хранению данных больше похож на набор снимков миниатюрной файловой системы. Каждый раз, когда вы делаете коммит, то есть сохраняете состояние своего проекта в Git, система запоминает, как выглядит каждый файл в этот момент, и сохраняет ссылку на этот снимок. Для увеличения эффективности, если файлы не были изменены, Git не запоминает эти файлы вновь, а только создаёт ссылку на предыдущую версию идентичного файла, который уже сохранён. Git представляетсвоиданныекак, скажем, **потокснимков**.

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

Модифицированное (modified) cостояние - изменения уже внесены в файл, но пока не зафиксированы в базе данных

Индексированное (staged)состояние - текущая версия модифицированного файлапомечена как требующая последующейфиксации

Зафиксированное (committed)состояние - данные надежно сохранены в локальнойбазе

1. Что такое индексация файла? Какой командой она выполняется?

Индексация файла в Git означает добавление изменений в файлы в индекс, таким образом подготавливая их к следующему коммиту. Этовыполняется с использованиемкомандыgit add

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

Фиксация (или коммит) в Git означает сохранение текущего состояния индексированных файлов в репозитории. Это выполняется с использованием команды gitcommit

1. Продемонстрируйте команду проверки выбранных настроек.

Для просмотра настроек Gitможно использовать команду: gitconfig–list

1. Как инициализировать репозиторий в существующей папке?

Если у вас уже есть существующая папка с проектом, и вы превратить ее в Git-репозиторий, используйте команду: gitinit

1. Как указать файлы, за которыми должна следить система?

Чтобы начать отслеживание файлов, используется команда: gitadd. Например, чтобы начать отслеживание всех файлов, можно выполнить: gitadd .

1. Как выполнить фиксацию изменений?

Чтобы зафиксировать изменения, нужно использовать команду gitcommit

1. Какой командой определить состояния файлов?

Чтобы определить состояния файлов, используется команда gitstatus. Она покажет измененные, индексированные и зафиксированные файлы.

1. Для чего создается файл.gitignore? Поясните его структуру.

Файл .gitignore создается для игнорирования определенных файлов и папок при работе с Git. Его структура состоит из шаблонов для игнорируемых файлов и папок.