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»

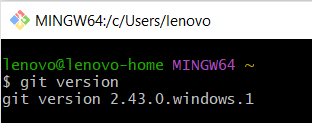
Выполнила:

Студентка 1 курса 10 группы

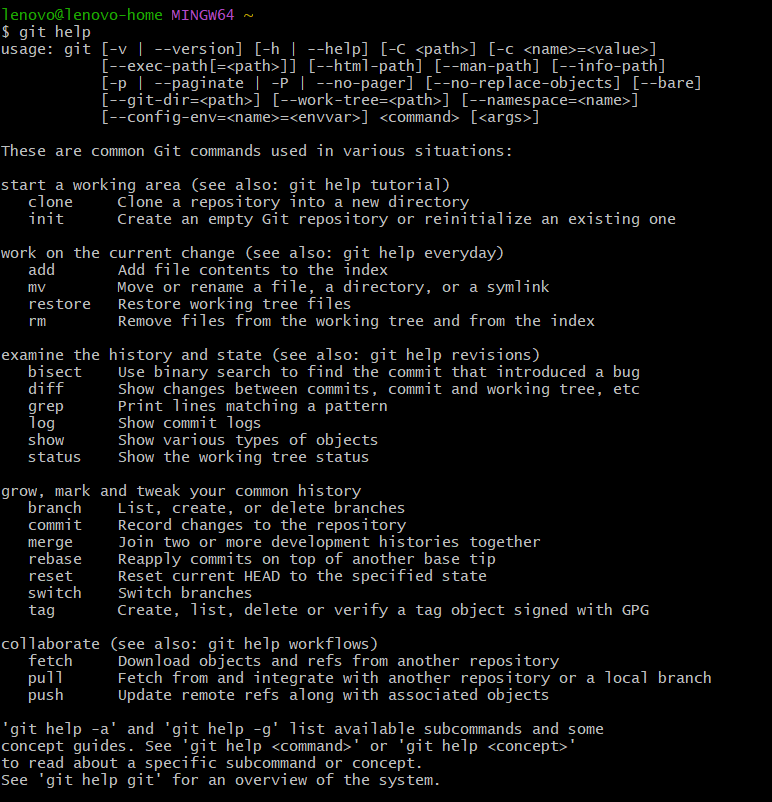
Рублевская Маргарита Владимировна

2023, Минск

4. Установленная версия git:

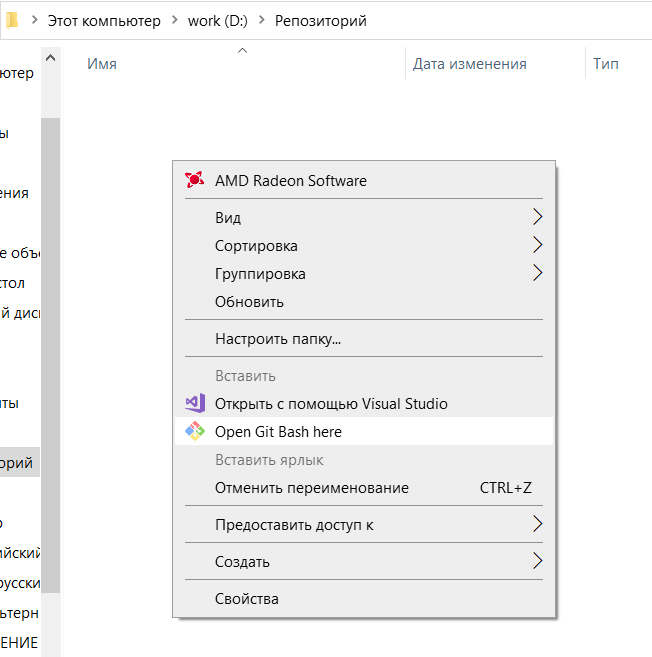


5. Команда $ git help



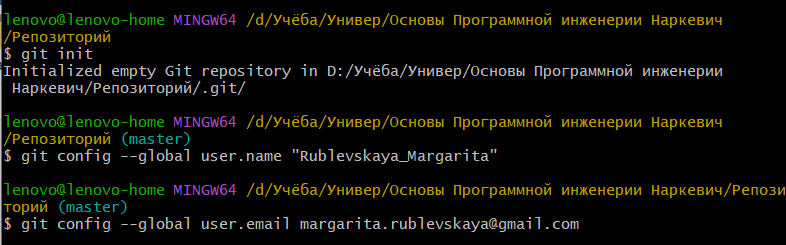
6. 1) Заходим в папку, в которой будет располагаться репозиторий

2) Запускаем GIT Bash here:



3) Используем в открывшемся окне команду $ git init

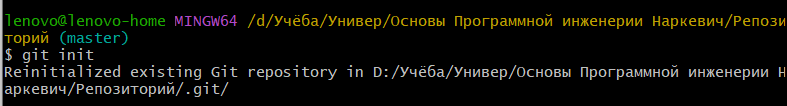
7. Выполним конфигурацию гит.

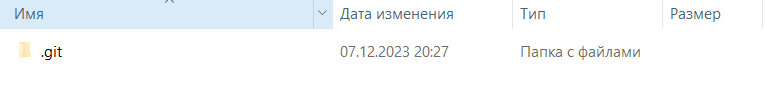


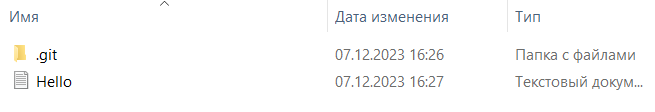
8. Проверим настройки конфигурационного файла командой  
git config --list



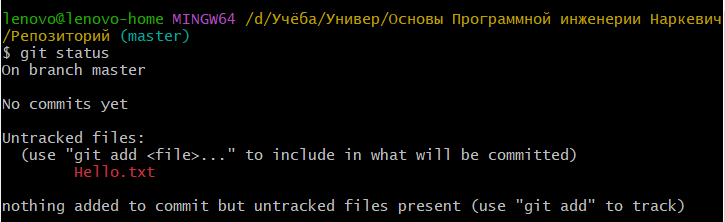
9. Инициализируем локальный репозиторий в рабочей папке



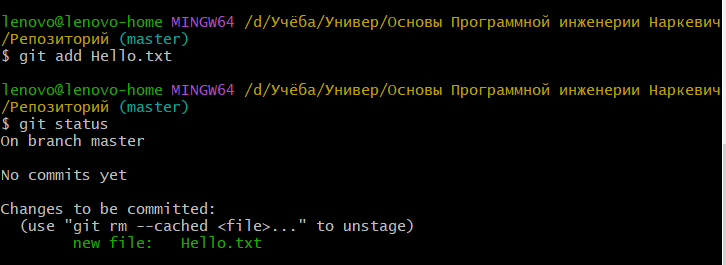




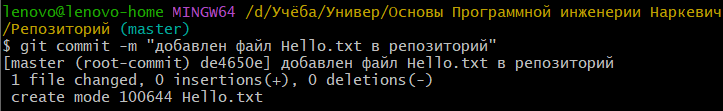
11. Красным цветом отмечаются новые и модифицированные файлы и папки



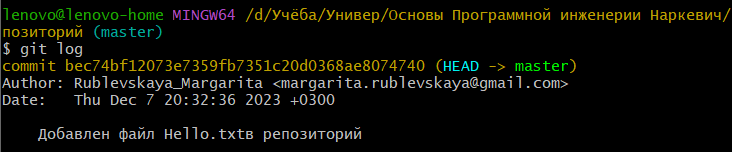
12. Команда отображает обновление состояния репозитория.



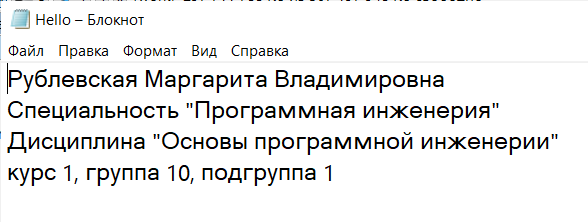
13. Элемент был зафиксирован в репозитории и добавлен к скрытым файлам. Об успешном коммите сообщают две последние строки сообщения. Файл добавлен в ветку master.

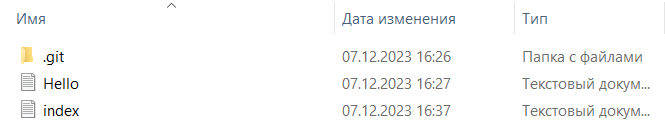


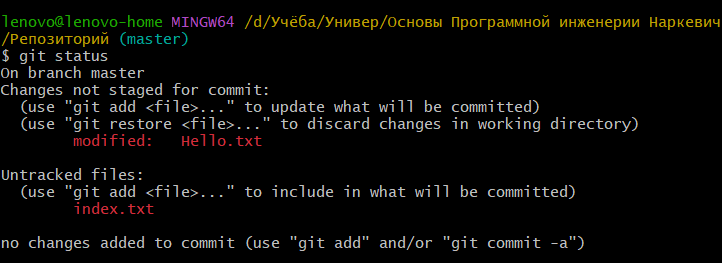
14. Просмотрим журнал с помощью команды git log

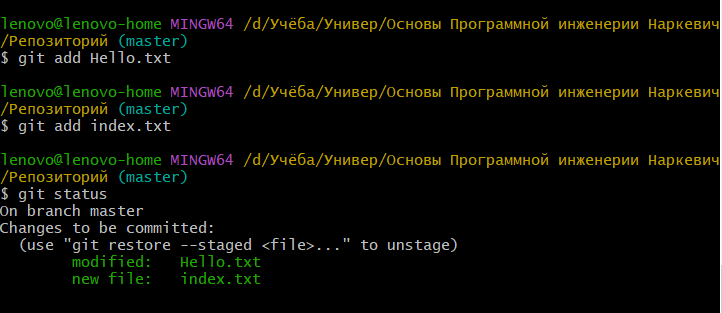


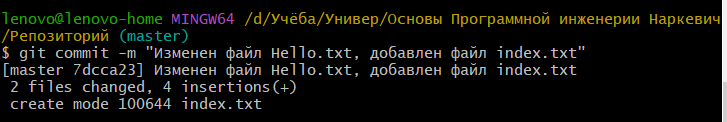
15. Изменим существующий файл Hello.txt, добавим в репозиторий новый файл и исследуем состояние репозитория,



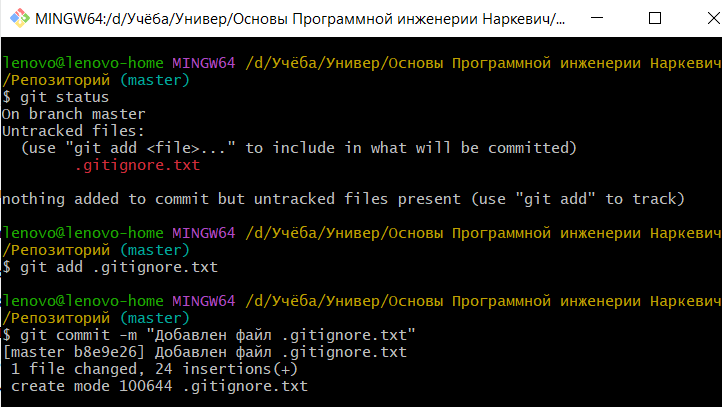








17. Ознакомимся с шаблонами .gitignore и настроим этот файл для нашего репозитория.



Вопросы:

1. Система контроля версий (СКВ) - это инструмент, который позволяет отслеживать изменения в файле или наборе файлов со временем. Она используется для управления и контроля версий файлов, отслеживания изменений, восстановления предыдущих версий и совместной работы над проектами.

2. Git использует принцип хранения файлов в виде снимков состояния проекта. Вместо сохранения только разницы между версиями, Git сохраняет полные снимки, которые отражают состояние проекта на определенный момент времени. Это позволяет быстро переключаться между версиями, восстанавливать предыдущие версии файлов и проводить операции такие как слияние (merge) и ветвление (branching).

3. Отличие Git от других систем контроля версий заключается в его распределенной архитектуре. Каждый пользователь Git имеет полную копию репозитория, включая историю изменений. Это позволяет работать над проектом offline, делать локальные изменения и объединять их с другими версиями. Git также обладает высокой производительностью и имеет широкий набор команд и инструментов для управления версиями.

4. Файлы в Git могут находиться в трех основных состояниях:

- Измененные (modified) - файлы, которые были изменены, но еще не были проиндексированы для фиксации.

- Проиндексированные (staged) - файлы, которые были отмечены для фиксации с помощью команды git add, но еще не были зафиксированы.

- Зафиксированные (committed) - файлы, которые были сохранены в локальной базе данных Git. Эти файлы представляют собой определенную версию проекта и являются сохраненными точками в истории.

5. Индексация файла в Git - это процесс добавления изменений файла в область индекса, перед их фиксацией. Команда для индексации файла в Git - git add <filename>. Она добавляет изменения файла в стейджинговую область, где Git сохраняет информацию о состоянии файла, чтобы в дальнейшем зафиксировать эти изменения.

6. Фиксация файла в Git - это процесс создания коммита, который сохраняет изменения файла в репозитории. Команда для фиксации файла в Git - git commit -m "commit message". Она создает коммит с сообщением, описывающим сделанные изменения.

7. Команда для проверки выбранных настроек в Git - git config --list. Она выводит список всех выбранных настроек, таких как имя пользователя, электронная почта и другие.

8. Для инициализации репозитория в существующей папке нужно открыть командную строку или терминал, перейти в папку проекта с помощью команды cd <путь к папке проекта> и выполнить команду git init.

9. Чтобы указать файлы, за которыми должна следить система, нужно добавить их в стейджинговую область с помощью команды git add <filename> для каждого файла, или git add . для добавления всех измененных файлов в текущем каталоге.

10. Для выполнения фиксации изменений нужно использовать команду git commit -m "commit message". Она создаст коммит с выбранным сообщением, фиксируя все изменения, которые были проиндексированы предварительно.

11. Командой git status можно определить состояние файлов. Она покажет список измененных, проиндексированных и неотслеживаемых файлов, а также информацию о текущей ветке.

12. Файл .gitignore создается для игнорирования определенных файлов или папок системой Git. В нем можно указать маски файлов, папок или расширений, которые не должны быть добавлены в репозиторий. Структура файла. gitignore состоит из паттернов, каждая строка в файле представляет собой отдельный паттерн, который описывает файлы или папки, которые нужно игнорировать.