ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

Московский технический университет связи и информатики

(МТУСИ)

Дисциплина «Технология разработки программного обеспечения»

Лабораторная работа № 1

на тему: «Основы работы с системой контроля версий»

Выполнила:

Студентка гр. М091901(75)

Алексеева Елизавета

Москва 2019

**Цель работы**: Изучение базовых функций системы контроля версий на примере Git

**Задание:** Определить проект (состоящий из нескольких файлов), который будет добавлен под версионный контроль, и произвести ряд операций в СКВ, включающих:

1. Создание репозитория или клонирование существующего;
2. Внесение ряда изменений в файл(ы) и выполнение коммитов (кол-во не меньше 2);
3. Создание веток и выполнение ряда изменений в проекте в созданных ветках;
4. Слияние ветки с веткой мастер без конфликтов;
5. Слияние ветки с веткой мастер с решением возникающих конфликтов;
6. Откат изменений до одного из предыдущих коммитов.

Возможна совместная работа студентов над одним проектом для изучения взаимодействия группы разработчиков и возможностей работы с СКВ.

**Выполнение:**

**Команда *git init*** – создает в текущем каталоге новый подкаталог с именем .git содержащий все необходимые файлы репозитория — основу репозитория Git. Далее, необходимо добавить в папку репозитория любые файлы. В данном случае это будет PracticeFile.txt

**Команда *git add*** – индексирует файлы для включения их в следующий коммит. После коммита добавленные файлы попадают под версионный контроль.

**Команда *git status*** – основной инструмент, используемый для определения, какие файлы в каком состоянии находятся.

На рисунке 1 представлено выполнение этих команд.

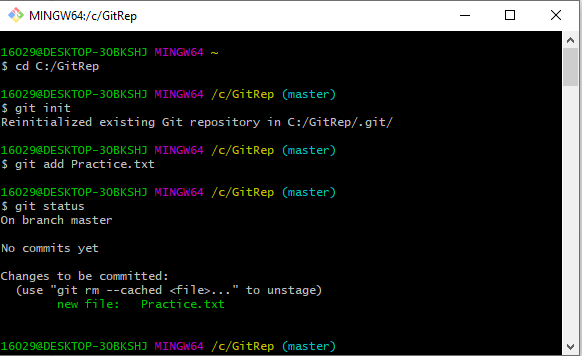


Рисунок 1. Выполнение команд git init, git add и git status

Изменим файл, добавив в него текст. Результат работы git status после изменения файла и его индексирования (рисунок 2):

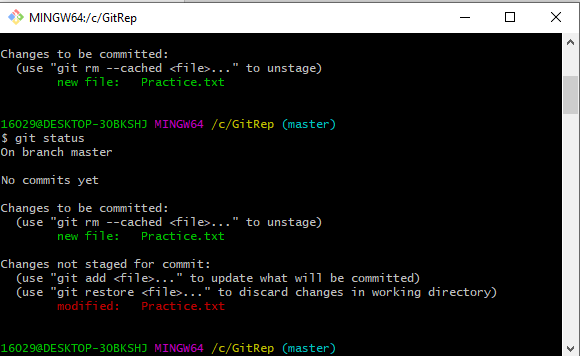


Рисунок 2. Выполнение команды git status после индексирования файла

**Команда *git commit*** – фиксирует проиндексированные изменения (рисунки 3). При коммите необходимо в кавычках указывать описание проиндексированных изменений через аргумент -m

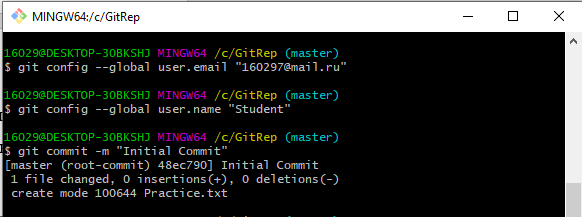


Рисунок 3. Выполнение команды git commit - результат

Если результат работы команды git status недостаточно информативен и необходимо знать, что конкретно поменялось, а не только какие файлы были изменены, можно использовать команду ***git diff***(рисунок 4).

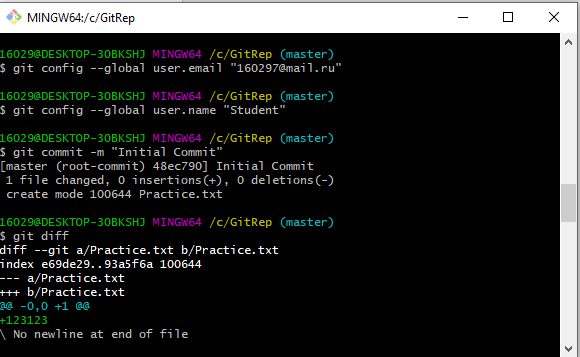


Рисунок 4. Выполнение команды git diff

**Команда *git reset*** выполняет отмены индексации файла (рисунок 5).

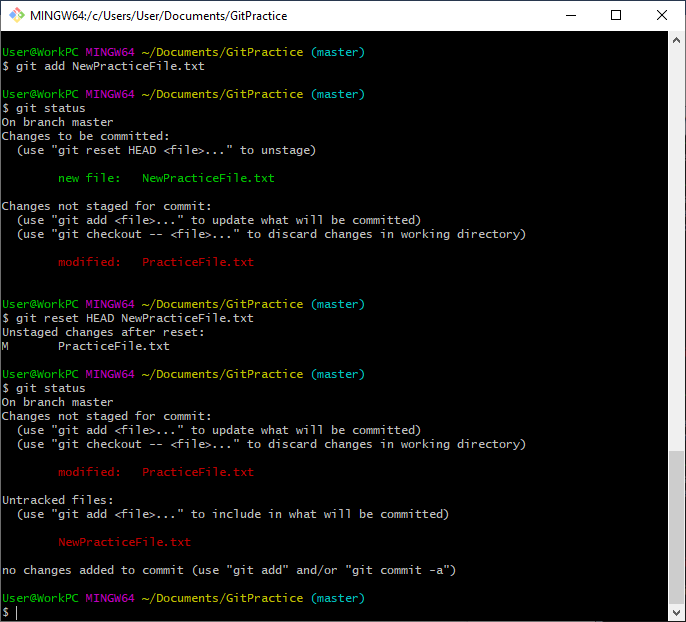


Рисунок 5. Выполнение команды git reset

Если необходимо скопировать уже существующий репозиторий в каталог, необходимо использовать команду ***git clone*** (рисунки 6-8).

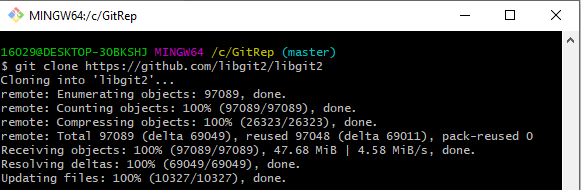


Рисунок 6. Выполнение команды git clone

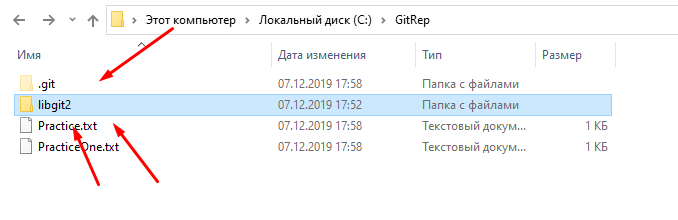


Рисунок 7. Выполнение команды git clone – загруженный каталог

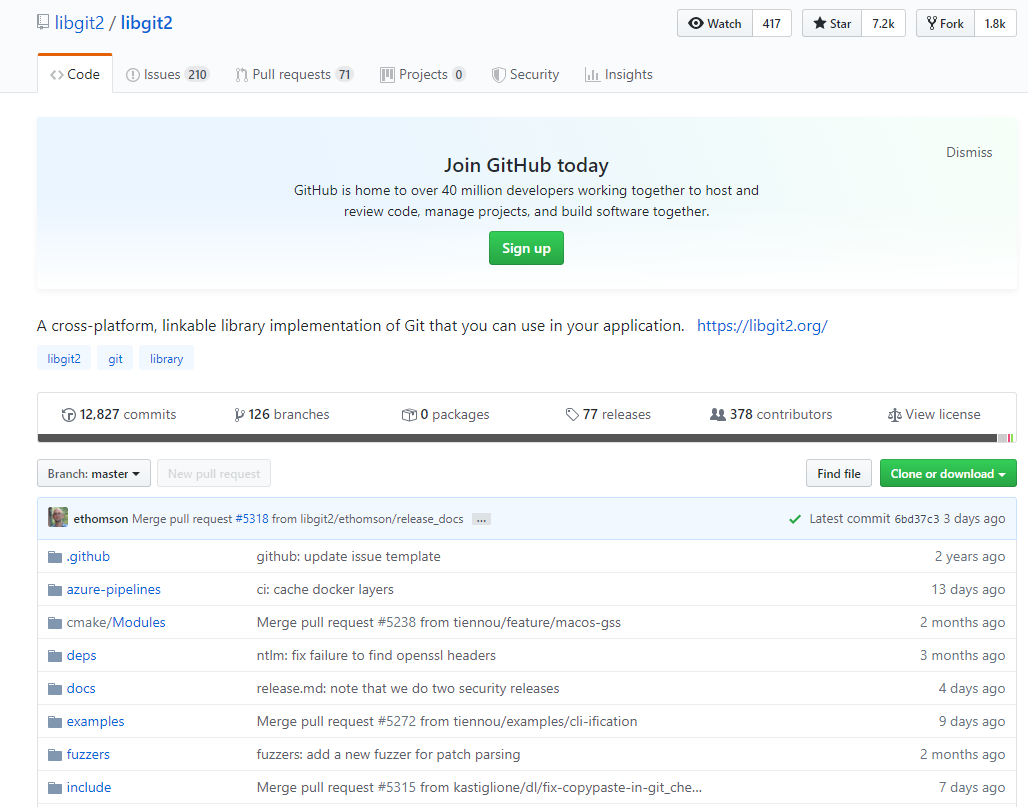


Рисунок 8. Выполнение команды git clone – содержимое репозитория

**Команда *git log*** – отображает историю репозитория (рисунок 9).

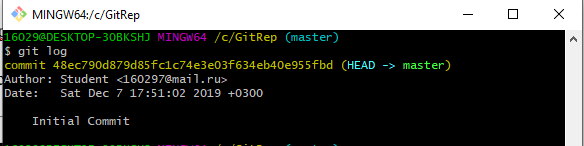


Рисунок 9. Выполнение команды git log

Ветки в git существуют для того, чтобы каждый программист мог отдельно работать с кодом, реализуя свой функционал. Команда *git branch* – создание новой ветки. Команда *git checkout* – осуществляет переход на существующую ветку. Команда *git merge* осуществляет слияние двух веток.

Результат выполнения команд представлен на рисунке 10.

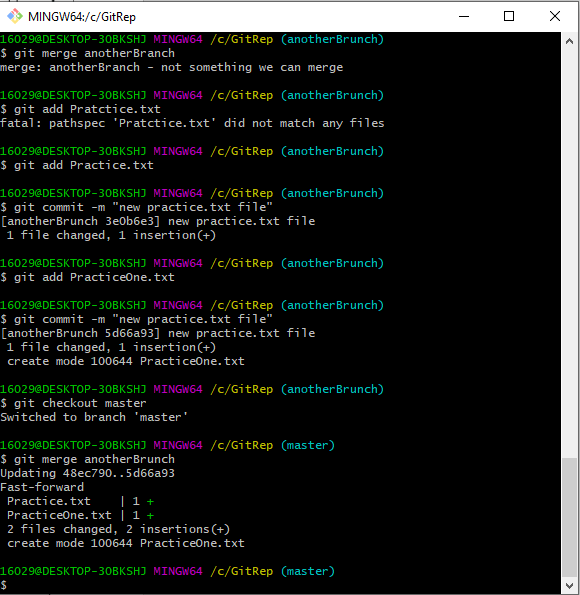


Рисунок 10. Выполнение команд git branch, git merge и git checkout без конфликта слияния

**Выводы**

Система управления версиями (СУВ) — это система, сохраняющая изменения в одном или нескольких файлах так, чтобы потом можно было восстановить определённые старые версии. Она позволяет вернуть файлы к прежнему виду, вернуть к прежнему состоянию весь проект, сравнить изменения с какого-то времени, увидеть, кто последним изменял модуль, который дал сбой, кто создал проблему, и так далее. Если, пользуясь СУВ, было сделано неудачное изменение или были потеряны файлы, всё можно легко восстановить. Кроме того, издержки на всё это будут очень маленькими.