**Лабораторная работа 6**

Системы контроля версий. Основы работы в Git

4. Для проверки установленной версии я использовал команду **$ git version**

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

5. Для получения справки я воспользовался командой **$ git help**

Изображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Автоматически созданное описание

6. Сначала я создал рабочую папку «lab6» в папке я создал файл «Hello.txt»





Затем в папке запустил Git Bash (с помощью контекстного меню: «Git Bash Here»).

Инициализировал репозиторий с помощью команды **$ git init**

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

В папке появилась новая скрытая папка .git (локальный репозиторий)



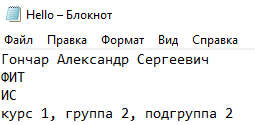
Для того, чтобы отобразить состояние (status) репозитория я использовал команду **$ git status.** Также была создана ветка **master**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Красным цветом отмечены новые и модифицированные файлы и папки (Hello.txt).

Содержимое Hello.txt



Для того, чтобы добавить файл Hello.txt в репозиторий индексированных файлов я использовал команду **$ git add Hello.txt.**



Далее я отобразил обновленное состояние репозитория при помощи команды **$ git status.**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание (Теперь файл Hello.txt проиндексирован)

Изменения я зафиксировал в репозитории при помощи команды **$ git commit -m "added Hello.txt to the repo".** (Ключ –m позволяет добавить комментарий, описывающий, что именно было изменено в коммите "added Hello.txt to the repo").

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Для того, чтобы посмотреть журнал я использовал команду **$ git log**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

7. Настройка конфигурационного файла

Изображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Автоматически созданное описание

15. Я изменил файл «Hello.txt»

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

После этого я создай новый файл «new.txt»



Добавил его в репозиторий



Затем исследовал состояние репозитория.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Здесь видно, что измененный файл «Hello.txt» модифицирован.

17. Создаем файл **.gitignore.** Помещаем туда файл, который хотим исключить.

Изображение выглядит как текст, Шрифт, линия, снимок экрана

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

test1 не добавился

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

18. Создаю новую ветку test при помощи команды **$ git branch <название ветки>**



Затем при помощи команды **$ git branch** получаю список всех веток и текущую

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание (Cейчас checkout стоит в ветке master)

Переключаю ветку при помощи команды **$ git checkout <название ветки>**



19. Создал и зафиксировал test.html

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

20. Текущая ветка test

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

И HEAD указывает на ветку test

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

21. Переходим в ветку в которой будем выполнять слияние



Затем при помощи команды **& git merge <название ветки>** соединяем две ветки в одну

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

22. В ветке master редактируем файл test.html

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Переходим в ветку test и видим, что в файле test.html изменений нет

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

(Добавим какое-то новое содержимое)

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

При переходе в ветку master мы получаем сообщение о том, что у нас есть измененный файл и мы не можем просто так изменить состояние с одной ветки на другую. Для этого мы будем вынуждены выполнить добавление данного файла на индексацию и после чего выполнить commit

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

Затем переходим в master, выполняем слияние и на выходе получаем конфликт

Изображение выглядит как текст, Шрифт, снимок экрана, Графика

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Переходим в test.html

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, снимок экрана

Автоматически созданное описание

Здесь мы должны отредактировать файл так, как мы хотим видеть его в финальном результате. До ===== показывается, что было в главной ветке, а после ==== что поменялось в ветке test. Затем редактируем файл как нам нужно и сохраняем

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Выполнив **$ git status** мы увидим подсказку (use «git merge –abort» для того, чтобы отменить слияние или добавить уже отредактированный файл на индексацию):

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

При commit мы не пишем комментарий так как был конфликт слияний и нам предложит комментарий, описывающий ситуацию (принято использовать именного его).

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Для выхода используем wq (записать и выйти).