**Команды командной строки:**

ls – позволяет получить список файлов и папок данной директории (*в CMD команда dir*).

cd (change dir) – смена директории.

Пример: cd~ - тильда приведёт нас обратно в наш каталог.

cd/ - слэш приведёт прямо на диск C: , с которого открыт терминал.

cd c: - обращение к диску C:\ .

cd .. – возвращает на каталог выше.

cd ../.. – возвращает на два каталога выше.

cd ../../.. – возвращает на три каталога выше.

cd . – текущий каталог (например, нужно перейти в какую-то папку относительно текущей, тогда пишем cd ./Desktop/школа ).

clear - очищает консоль (когда много чего понаписывал) (в CMD команда CLS , как Clear Screen).

В Git все команды начинаются со слова git. Например, git init, git commit, git status.

Дальше может идти опция, которая относится к конкретной команде (git config --local). У одной команды может быть несколько опций.

У Git есть встроенный справочник команд, который доступен из консоли. Данный справочник вызывается git help -a . Для пролистывания справочника нажимаем Пробел. Когда дошли до конца и хотим выйти из справочника, жмём клавишу Q (в английской раскладке).

Если нужна справка по какой-либо команде, пишем эту команду и в конце добавляем опцию

--help.

Например, git status --help.

**Команды Git**

git init – инициализация системы контроля версий Git в текущем проекте, для того чтобы можно было начать делать снимки своего проекта в Git (делаем один раз, если уже работали с данным проектом и применяли эту команду, то второй раз делать не нужно).

**Репозиторий** – специальный каталог, в котором хранятся все изменения, которые когда либо были сделаны во время всей жизни вашего проекта, с момента инициализации git-репозитория и до полного завершения проекта.

После инициализации появляется скрытая папка .git , которую нельзя изменять, удалять, переносить и т.д.

git commit – специальная команда для сохранения файла в .git-репозиторий (файл копируется в .git-репозиторий и получает специальный идентификатор хэш).

**Хэш** – специальный набор символов (40 16-тиричных символов). Он хранит в себе инфо о том кто сделал изменения, дату изменения и ссылку на предыдущую версию проекта, если она была. Если ничего не менялось, то будет хранить ссылку на предыдущую версию (коммит).

Все изменения попадают в .git-репозиторий навсегда и хранятся там.

Последняя версия проекта получает специальный указатель HEAD и весь локальный репозиторий ссылается на HEAD.

HEAD – указатель на последнее изменение.

Конфигурационный файл:

* системный
* глобальный
* локальный

git config --system --list - отображает параметры системного конфига (сюда не лезем).

git config --global --list - отображает параметры глобального конфига.

\* значения user.name и user.email .

Глобальный конфиг – настраивается на уровне вашего компьютера. Ты можешь работать над своим проектом за чужим компьютером, за своим.

git config --global user.name “Alex” – добавление для user.name значения “Alex”

git config --local –list

git status – даёт инфо в каком состоянии находятся файлы текущего проекта.

git add – добавление файлов в staging area (отслеживаемые) для отслеживания.

git add index.html - добавление в отслеживаемые файла index.html

git add ./ - добавляет всю текущую директорию и все файлы в ней в отслеживаемые.

git diff - даёт возможность посмотреть разницу между версиями до изменения в отслеживаемых файлах и текущей версией (рассматривает разницу между рабочим каталогом и индексом, т.е. staging area и именно до сохранения).