**Цель: Научиться работать с Git.**

**Задачи: Клонировать репозиторий, научиться фиксировать изменения и пушить в репозиторий.**

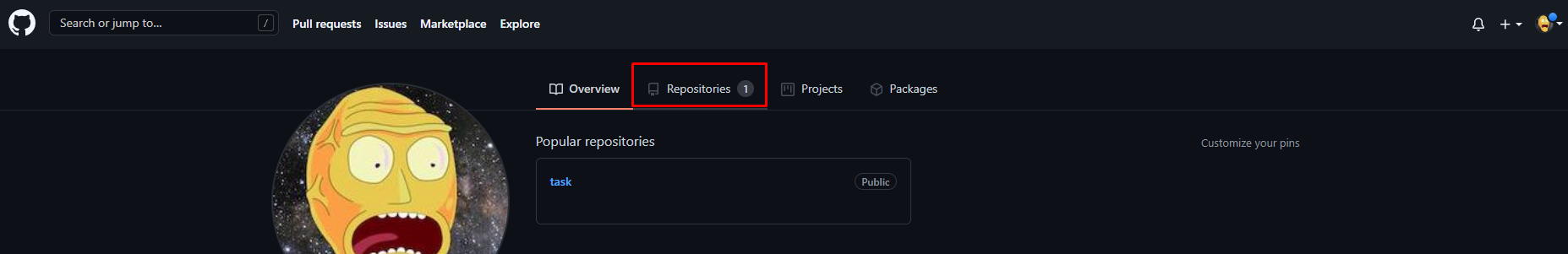
Чтобы клонировать репозиторий нужно зайти на githab во кладку репозитории (Рисунок 1).  


Рисунок 1 - Переход во вкладку репозиториев.

Выбираем нужный репозиторий, тут важно подметить, что репозитории бывают публичными и приватными. Если репозиторий приватный, то доступ к нему нужно попросить у его владельца.  
 Выбираем репозиторий, переходим по ссылке. В открывшемся репозитории выбираем “code” и копируем ссылку (Рисунок 2).

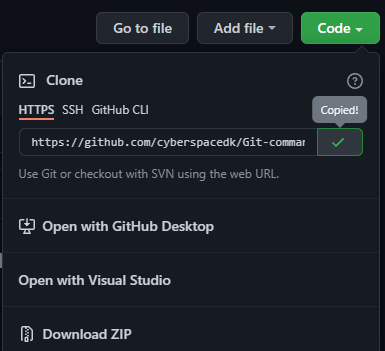


Рисунок 2 - копирование ссылки на репозиторий.

Создаём папку в которую мы будем клонировать наш репозиторий (Рисунок 3).

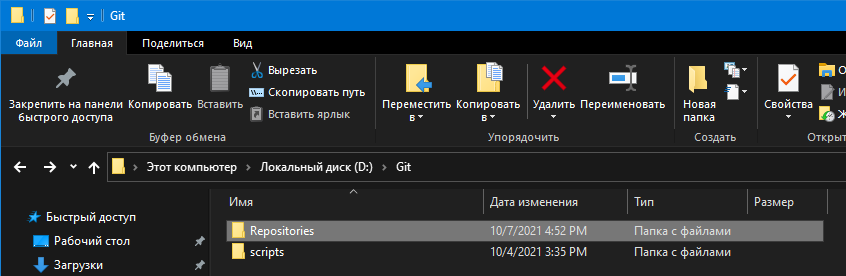


Рисунок 3 - создаем папку.

В путь файла, вместо всего, что там есть пишем cmd и нажимаем enter (Рисунок 3 - 4).

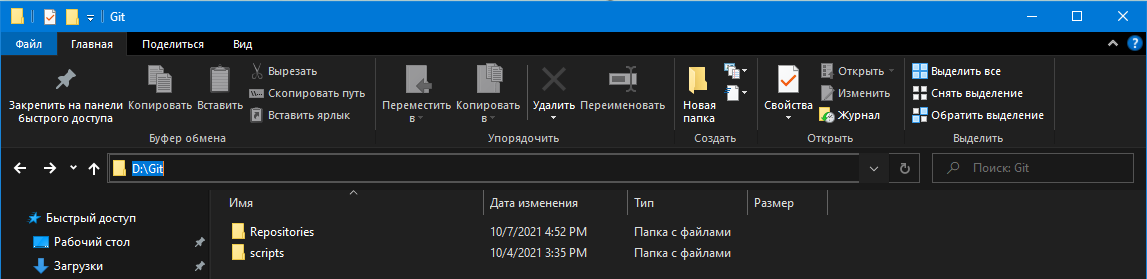


Рисунок 3 - путь к файлу.

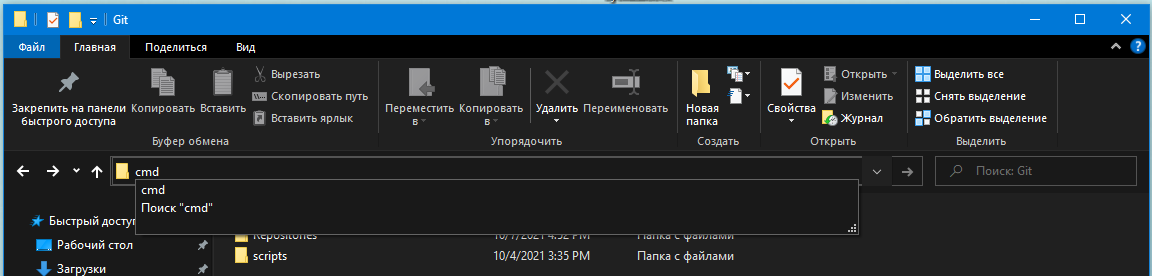


Рисунок 4 - пишем cmd.

В открывшейся консоли нужно перейти в нужную директорию с помощью команды cd и написания интересующей нас папки, в которую мы хотим клонировать то, что находиться в репозитории (Рисунок 5).

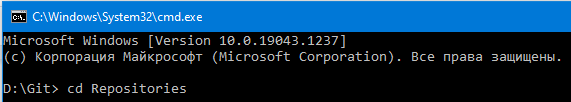


Рисунок 5 - переход в нужную директорию.

После того, как мы находимся в нужной папке - вставляем нашу ссылку из гита (Рисунок 2 и 6)

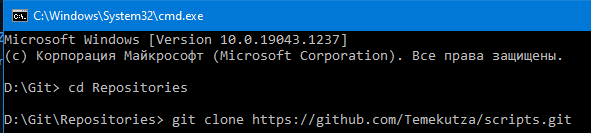
****

Рисунок 6 - Клонирование из репозитория.

Если всё прошло успешно, то в выбранной папке появится то, что мы копировали с репозитория (Рисунок 7, 8).

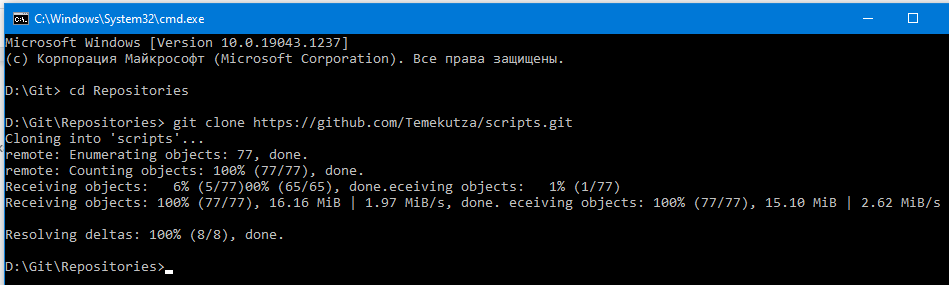
****

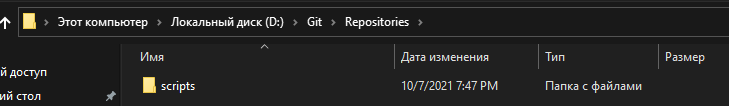
Рисунок 7 - Процесс копирования.

Рисунок 8 - то что скопировалось.

Прежде чем продолжить, для начала нужно разобраться что такое коммит.

Коммит — «сохранение» (хранит набор изменений, сделанный в рабочей директории с момента предыдущего коммита). Коммит неизменен, его нельзя отредактировать.

У всех коммитов (кроме самого первого) есть один или более родительских коммитов, поскольку коммиты хранят изменения от предыдущих состояний.

Начнём с того, что нужно проинициализировать репозиторий. Для этого в консоли нужно прописать команду “git init” (рисунок 9).

“git init” - Эта команда создаёт в текущем каталоге новый подкаталог с именем “.git”, содержащий все необходимые файлы репозитория — структуру Git репозитория. На этом этапе проект ещё не находится под версионным контролем.

После выполнения команды получаем следующие

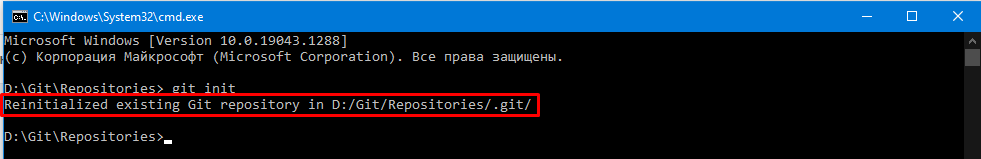


Рисунок 9 - инициализация проекта.

Теперь с помощью команды “git status” вы сможете посмотреть все изменений в проекте на вашем ПК (рисунок 10). Чтобы их сохранить, потребуется выполнить команду . Также тут отображается в какой ветке мы сейчас находимся (рисунок 10.1).

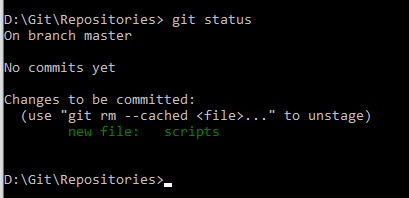


Рисунок 10 - просмотр изменений.

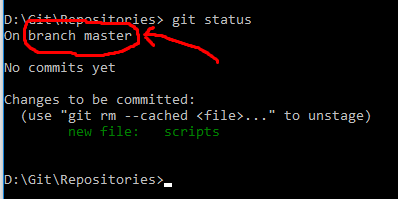


Рисунок 10.1 - ветка мастер.

Ветка в Git — это простой перемещаемый указатель на один из таких коммитов. По умолчанию, имя основной ветки в Git — master. Как только вы начнёте создавать коммиты, ветка master будет всегда указывать на последний коммит. Каждый раз при создании коммита указатель ветки master будет передвигаться на следующий коммит автоматически. Всё это работает следующим образом (рисунок 11). Важно понимать, что когда мы создаем новую ветку, она копирует всё, что есть в текущей ветке, и в дальнейшем будет ссылаться на ту ветку из которой её создают (рисунок 12).

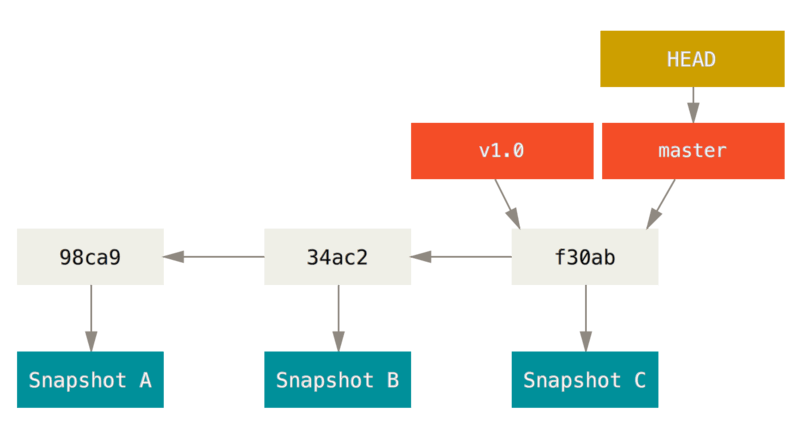


Рисунок 11 - работа с ветками и история коммитов.